MALIGNANT COMMENTS CLASSIFIER

Submitted by:

PRAKHAR PRAKASH

Email: [mr.prakhar@gmail.com](mailto:mr.prakhar@gmail.com)

Github: prakhar3146

**ACKNOWLEDGMENT**

**In this project,I have used the study material,vlogs and sample projects provided(as a part of my data science with nlp course) by the DataTrained institute(based in Noida) for references.**

**Apart from that websites** [**www.towardsdatascience.com,www.wikipedia.org,sklearn,www.dataaspirant.com**](http://www.towardsdatascience.com,www.wikipedia.org,sklearn,www.dataaspirant.com) **were also used.**

Pages :27

Wordcount: 5026

INDEXING

|  |  |
| --- | --- |
| Page-3 | Heading-  Introduction |
| 5 | Review of literature |
| 6 | Motivation for problem undertaken |
| 6 | Analytical problem framing |
| 8 | Data sources and their formats |
| 10 | Data preprocessing done |
| 13 | Hardware and Software Requirements and Tools Used |
| 15 | Model development and evaluation |
| 23 | Conclusion |
|  |  |

**INTRODUCTION**

**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.**

**An article published by the guardian claimed a rapid surge in online hatred during covid lockdowns and labelled it “NASTIER THAN EVER”**

**Here’s what it said:**

**The coronavirus pandemic has been punctuated by outbursts of online hate that have had real-world consequences. The most alarming examples are the 6 January riot in Washington, which was spurred by rightwing groups organising online, and the racist abuse of England footballers during Euro 2020 that culminated in a mural of Marcus Rashford being defaced in Manchester.**

**Following the killing of Sir David Amess, there have been renewed concerns about whether lockdowns have created the conditions for a surge in hate, as frustrated extremists or people vulnerable to radicalisation hunkered down over their laptops and mobile phones.**

**The Report Harmful Content platform, run by the UK Safer Internet Centre, reported a 225% increase in online hate speech incidents in the UK last year.**

**According to Imran Ahmed, chief executive of the Center for Countering Digital Hate, a US- and UK-based campaign group, the pandemic has given an edge to the existing culture of online abuse and hate associated with social media and video-sharing platforms. “It’s just a little bit nastier than ever before,” says Ahmed.**

**He says coronavirus has enhanced a link between “disgust sensitivity” and xenophobia, whereby a sense of disgust – triggered by disease, for instance – makes people find their own group more attractive and those outside it less so. “The pandemic has driven all types of movements which are based on protection of identity groups, those groups people feel kinship to.” Pointing to the US, he adds: “If you look at the surge in rightwing, authoritarian identity movements, that has been really, really stark.”**

**According to Ofcom, the UK communications regulator, British adults were online for longer than their counterparts in other European countries last year, spending on average three hours and 37 minutes online each day, a rise of 8% on 2019.**

**Even without exact platform-by-platform data on instances of hate crime, the political and regulatory environment for social media companies after lockdown has hardened. In the US the competition regulator is trying to break up Facebook, in the EU the competition commissioner also says big tech firms should be broken up, while in the UK the draft online safety bill will impose a duty of care on social media companies to protect users from harmful content.**

**There have been renewed calls to tackle abuse from anonymous social media accounts since the killing of Amess on Friday. The UK’s home secretary, Priti Patel, hinted at a crackdown on online anonymity on Sunday, although attempting to ban it entirely would raise objections on freedom of speech grounds. Companies under the scope of the online safety bill, such as Facebook and Twitter, will be expected to tackle anonymous abuse as part of their duty of care.**

**Damian Collins, the Conservative MP chairing the joint committee scrutinising the online safety bill, said anonymity should be retained, but social media companies must have a means of identifying abusers. ​​**

**“I don’t think anonymity should be taken away,” said Collins. “But if users exploit it to break hate speech laws or those against incitement of violence, I think social media companies should have enough information on who they really are so that they are able to clearly identify them to the police.”**

**Moderation of harmful and radicalising content will remain a point of focus. The Wall Street Journal published another exposé of Facebook’s safety systems on Sunday when it cited internal documents indicating that the platform was struggling to deal with hateful content. According to one internal document, Facebook’s automated systems removed posts that generated 2% of the views of hate speech on the platform.**

**In response, Facebook said the statistics did not reflect its full range of measures to tackle hate and that over the past year it had halved the amount of views of content that contained hate speech, to five out of every 10,000 content views. Nonetheless, the regulatory and political pressure is going to increase against online hate and the companies that host it.**

**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

**Before deep learning (NLP), companies resorted to ineffective methods of identifying hate speech, such as simple keyword searches (bag of word). This method has “high recall but leads to high rates of false positives” , mistakenly removing normal conversation. Recently, research has already been conducted in the deep learning field to identify hate speech. A paper published in August 2019 used multiple-view stacked Support Vector Machine (mSVM) to achieve approximately 80% accuracy with data from various social media companies . Another paper published in 2018 utilizes various word embeddings to train a CNN\_GRU model, achieving 90% accuracy on 3 different classes In addition, many social media companies have invested in methods to eliminate online hate speech. In July 2020, Facebook Canada announced that it is "teaming up with Ontario Tech University's Centre on Hate, Bias and Extremism to create what it calls the Global Network Against Hate" , for which Facebook will invest $500,000 to spot online extremism and countering methods.**

* Motivation for the Problem Undertaken

The project was given as an assignment during my internship at the Flip-robotechnologies,which is a banglore based software company.

Company address: Floor, Suite No.1759, #39, 2nd, NGEF Ln, Indiranagar, Bengaluru, Karnataka 560038

Also,as a social media user and a responsible member of the society,the increase in the cyberbulling aswell as online hate is a serious cause of concern for me and I would really like to play a part in its prevention/detection or at least contribute to the cause.

**Analytical Problem Framing**

A number of tools/libraries were used for the mathematical/analytical modelling and visualization of the dataset.

The libraries used were:

NUMPY

**NumPy is a Python library used for working with arrays.**

**It also has functions for working in domain of linear algebra, fourier transform, and matrices.**

**NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.**

**NumPy stands for Numerical Python.**

**Why Use NumPy?**

**In Python we have lists that serve the purpose of arrays, but they are slow to process.**

**NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.**

**The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.**

**Arrays are very frequently used in data science, where speed and resources are very important.**

**PANDAS**

**pandas is a**[**software library**](https://en.wikipedia.org/wiki/Software_library)**written for the**[**Python programming language**](https://en.wikipedia.org/wiki/Python_(programming_language))**for data manipulation and**[**analysis**](https://en.wikipedia.org/wiki/Data_analysis)**. In particular, it offers**[**data structures**](https://en.wikipedia.org/wiki/Data_structure)**and operations for manipulating numerical tables and**[**time series**](https://en.wikipedia.org/wiki/Time_series)**. It is**[**free software**](https://en.wikipedia.org/wiki/Free_software)**released under the**[**three-clause BSD license**](https://en.wikipedia.org/wiki/3-clause_BSD_license)**. The name is derived from the term "**[**panel data**](https://en.wikipedia.org/wiki/Panel_data)**", an**[**econometrics**](https://en.wikipedia.org/wiki/Econometrics)**term for**[**data sets**](https://en.wikipedia.org/wiki/Data_set)**that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself.**[**Wes McKinney**](https://en.wikipedia.org/wiki/Wes_McKinney)**started building what would become pandas at**[**AQR Capital**](https://en.wikipedia.org/wiki/AQR_Capital)**while he was a researcher there from 2007 to 2010.**

## **Library features[**[**edit**](https://en.wikipedia.org/w/index.php?title=Pandas_(software)&action=edit&section=1)**]**

* **DataFrame**[**object**](https://en.wikipedia.org/wiki/Object-oriented_programming)**for data manipulation with integrated indexing.**
* **Tools for reading and writing data between in-memory**[**data structures**](https://en.wikipedia.org/wiki/Data_structure)**and different**[**file formats**](https://en.wikipedia.org/wiki/File_format)**.**
* **Data alignment and integrated handling of missing data.**
* **Reshaping and pivoting of data sets.**
* **Label-based slicing, fancy indexing, and subsetting of large data sets.**
* **Data structure column insertion and deletion.**
* **Group by engine allowing split-apply-combine operations on data sets.**
* **Data set merging and joining.**
* **Hierarchical axis indexing to work with high-dimensional data in a lower-dimensional data structure.**
* **Time series-functionality: Date range generation**[**[6]**](https://en.wikipedia.org/wiki/Pandas_(software)#cite_note-6)**and frequency conversions, moving window**[**statistics**](https://en.wikipedia.org/wiki/Statistics)**, moving window**[**linear regressions**](https://en.wikipedia.org/wiki/Linear_regression)**, date shifting and lagging.**
* **Provides data filtration.**

Use in this project:

**Most of the work in this project was done with the help of the Pandas library.**

**Right from the first line of code after importing the required libraries was reading the dataset by reading it in the form of a dataframe(two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and** **columns)**

df**=**pd**.**read\_csv('malignant\_comments\_classifier.csv',sep**=**'\t')

**After the dataset was present in a tabular format,methods such as ‘df.dtypes-used to check the datatypes of the features) and ‘df.isnull().sum()-used to check for any missing values present in the dataset’.**

**Pandas methods used in this project are:**

df**.**drop()

df.dtypes

pd.read\_csv()

df.isnull().sum()

df.head()

df['feature']**.**unique()

df['feature']**.n**unique()

df.columns

df.shape

df.groupby()

df.describe()

df.value\_counts()

pd.DataFrame()

pd.crosstab()

* Data Sources and their formats

**The project was given as an assignment during my internship at the Flip-robotechnologies,which is a banglore based software company.**

**Company address: Floor, Suite No.1759, #39, 2nd, NGEF Ln, Indiranagar, Bengaluru, Karnataka 560038**

A snapshot of the dataset:

| **id** | **comment\_text** | **malignant** | **highly\_malignant** | **rude** | **threat** | **abuse** | **loathe** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0000997932d777bf | Explanation\nWhy the edits made under my usern... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **1** | 000103f0d9cfb60f | D'aww! He matches this background colour I'm s... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **2** | 000113f07ec002fd | Hey man, I'm really not trying to edit war. It... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **3** | 0001b41b1c6bb37e | "\nMore\nI can't make any real suggestions on ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **4** | 0001d958c54c6e35 | You, sir, are my hero. Any chance you remember... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **159616** | ffe987279560d7ff | ":::::And for the second time of asking, when ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **159617** | ffea4adeee384e90 | You should be ashamed of yourself \n\nThat is ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **159618** | ffee36eab5c267c9 | Spitzer \n\nUmm, theres no actual article for ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **159619** | fff125370e4aaaf3 | And it looks like it was actually you who put ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **159620** | fff46fc426af1f9a | "\nAnd ... I really don't think you understand... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

159621 rows × 8 columns

The features along with their description:

|  |  |
| --- | --- |
| **VARIABLE** | **DESCRIPTION** |
| id | A unique id aligned with each comment text. |
| comment\_text | It includes the comment text. |
| malignant | It is a column with binary values depicting which comments are malignant in nature. |
| highly\_malignant | Binary column with labels for highly malignant text. |
| rude | Binary column with labels for comments that are rude in nature. |
| threat | Binary column with labels for threatening context in the comments. |
| abuse | Binary column with labels with abusive behaviour. |
| loathe | Label to comments that are full of loathe and hatred. |

* Data Preprocessing Done

**The independent variable comment text was used in classifying the comment categories.**

**The text data was very unclean(full of noise,punctuations and stopwords) which were a major roadblock in classification of the text.**

**To overcome the problem stated above,nltk library and its members were used.**

NLTK(Natural Language Toolkit)

[**Natural language processing**](https://en.wikipedia.org/wiki/Natural_language_processing)**(NLP) is a field that focuses on making natural human language usable by computer programs. NLTK, or**[**Natural Language Toolkit**](https://www.nltk.org/)**, is a Python package that you can use for NLP.**

**A lot of the data that you could be analyzing is**[**unstructured data**](https://en.wikipedia.org/wiki/Unstructured_data)**and contains human-readable text. Before you can analyze that data programmatically, you first need to preprocess it. In this tutorial, you’ll take your first look at the kinds of text preprocessing tasks you can do with NLTK so that you’ll be ready to apply them in future projects. You’ll also see how to do some basic text analysis and create visualizations.**

The code for the relevant pakage imports:

import nltk

import numpy as np

import re

import pandas as pd

import string

from nltk.corpus import stopwords # Remove useless words

from nltk.stem.lancaster import LancasterStemmer # Convert words to base form; aggressive

# Import packages that help us to create document-term matrix

from sklearn.feature\_extraction.text import CountVectorizer, TfidfVectorizer

## TfidfVectorizer:

**TF-IDF is technique in Natural Language Processing for converting words in Vectors and with some semantic information and it gives weighted to uncommon words , used in various NLP applications.**

CountVectorizer

**CountVectorizer is a great tool provided by the scikit-learn library in Python. It is used to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text. This is helpful when we have multiple such texts, and we wish to convert each word in each text into vectors (for using in further text analysis).**

* Data Inputs- Logic- Output Relationships

**The relationship within and among the various features were analysed using a countplot from the seaborn library and also with the help of the crosstab() method from pandas.**

The observations were:

A)Within the features:

The target variable(malignant) is highly imbalanced with the 'NO'(0.0) class having 1,44,239(90% approx.) instances and the 'YES'(1.0) having only 15,289 instances.

The 'highly\_malignant' feature also has an imbalanced distribution with 0.0('NO') class constituting almost 99%(1,57,932) of the feature's data whereas 1.0('YES') category has only 1,596.

The rude feature is also imbalanced with 0.0('NO') category constituting 1,51,084(95% approx.) and "YES" category constituting 8,444(5% approx).

The threat feature was highly imbalanced with 0.0('NO') category constituting 1,59,050(99.5% approx.) of the data and 1.0('YES') having only 478 instances.

The abuse feature was also highly imbalanced with 0.0('NO') category constituting 1,51,653(95% approx.) and 1.0('YES') constituting 7875(5% approx) of the data.

The loathe feature is highly imbalanced with 0.0('NO') category having 1,58,104(99% approx.) and 1.0('YES') has only 1,405 instances.

B) Relationships among the variables:

Malignant vs highly malignant:

1)Out of 15,289 malignant comments,1594 comments(10.4%) were highly malignant. 2)All the comments that were not malignant were also not highly\_malignant.

3)13,695 comments out of 15,289 malignant(1.0) comments were not highly malignant.

Malignant vs rude:

1)7,922 comments were rude and malignant

2)7637 comments were malignant but not rude

3)522 comments were rude but not malignant

4)1,43,698 comments were neither rude nor malignant.

Malignant vs threat:

 1)14,840 comments were malignant but not threatning.

2)449 comments were malignant aswell as threatning.

3)29 comments were threatning but not malignant.

4)1,44,191 comments were neither malignant nor threatning.

Malignant vs Abuse:

1)7,342 comments were malignant as well as abuse.

2)7,947 comments were malignant but not abuse.

3)533 comments were abuse but not malignant.

4)1,43,687 comments were neither malignant nor abuse.

Malignant vs loathe:

 1)1,302 comments were malignant and loathe

2)13,987 comments were malignant but not loathe

3)103 comments were loathe but not malignant

4)1,44,117 comments were neither malignant nor loathe.

There was some data missing in the dataset.

* id 0
* comment\_text 74
* malignant 93
* highly\_malignant 93
* rude 93
* threat 93
* abuse 93
* loathe 112

**After giving the figures proper thought it is observed that the number of rows having missing values are negligible when compared to the dataset,hence not performing any replace operations as it may mislead the model.Instead,the rows with missing values were dropped.**

* Hardware and Software Requirements and Tools Used

Jupyter Notebook

**Jupyter Notebook (formerly IPython Notebooks) is a**[**web-based interactive**](https://en.wikipedia.org/wiki/Web_application)**computational environment for creating**[**notebook**](https://en.wikipedia.org/wiki/Notebook_interface)**documents.**

**A Jupyter Notebook document is a browser-based**[**REPL**](https://en.wikipedia.org/wiki/Read%E2%80%93eval%E2%80%93print_loop)**containing an ordered list of input/output cells which can contain code, text (using**[**Markdown**](https://en.wikipedia.org/wiki/Markdown)**), mathematics,**[**plots**](https://en.wikipedia.org/wiki/Plot_(graphics))**and**[**rich media**](https://en.wikipedia.org/wiki/Interactive_media)**. Underneath the interface, a notebook is a**[**JSON**](https://en.wikipedia.org/wiki/JSON)**document, following a versioned schema, usually ending with the ".ipynb" extension.**

**Jupyter notebooks are built upon a number of popular**[**open-source**](https://en.wikipedia.org/wiki/Open-source_software)**libraries:**

* [**IPython**](https://en.wikipedia.org/wiki/IPython)
* [**ZeroMQ**](https://en.wikipedia.org/wiki/ZeroMQ)
* [**Tornado**](https://en.wikipedia.org/wiki/Tornado_(web_server))
* [**jQuery**](https://en.wikipedia.org/wiki/JQuery)
* [**Bootstrap (front-end framework)**](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))
* [**MathJax**](https://en.wikipedia.org/wiki/MathJax)

**Jupyter Notebook can connect to many kernels to allow programming in different languages. A Jupyter kernel is a program responsible for handling various types of requests (**[**code execution**](https://en.wikipedia.org/wiki/Execution_(computing))**,**[**code completions**](https://en.wikipedia.org/wiki/Autocomplete)**, inspection), and providing a reply. Kernels talk to the other components of Jupyter using**[**ZeroMQ**](https://en.wikipedia.org/wiki/ZeroMQ)**, and thus can be on the same or**[**remote machines**](https://en.wikipedia.org/wiki/Remote_computer)**. Unlike many other Notebook-like interfaces, in Jupyter, kernels are not aware that they are attached to a specific document, and can be connected to many clients at once. Usually kernels allow execution of only a single language, but there are a couple of exceptions.[**[**citation needed**](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)**] By default Jupyter Notebook ships with the IPython kernel. As of the 2.3 release**[**[12]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-releasenote23-12)[**[13]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-releasenote20-13)**(October 2014), there are 49 Jupyter-compatible kernels for many programming languages, including Python, R, Julia and Haskell.**[**[14]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-14)

**A Jupyter Notebook can be converted to a number of**[**open standard**](https://en.wikipedia.org/wiki/Open_standard)**output formats (**[**HTML**](https://en.wikipedia.org/wiki/HTML)**,**[**presentation slides**](https://en.wikipedia.org/wiki/Presentation_slide)**,**[**LaTeX**](https://en.wikipedia.org/wiki/LaTeX)**,**[**PDF**](https://en.wikipedia.org/wiki/PDF)**,**[**ReStructuredText**](https://en.wikipedia.org/wiki/ReStructuredText)**,**[**Markdown**](https://en.wikipedia.org/wiki/Markdown)**, Python) through "Download As" in the web interface, via the nbconvert library**[**[15]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-15)**or "jupyter nbconvert" command line interface in a shell. To simplify visualisation of Jupyter notebook documents on the web, the nbconvert library**[**[16]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-16)**is provided as a service through NbViewer**[**[17]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-17)**which can take a URL to any publicly available notebook document, convert it to HTML on the fly and display it to the user.**

**The notebook interface was added to IPython in the 0.12 release**[**[18]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-releasenote012-18)**(December 2011), renamed to Jupyter notebook in 2015 (IPython 4.0 is Jupyter 1.0). Jupyter Notebook is similar to the notebook interface of other programs such as**[**Maple**](https://en.wikipedia.org/wiki/Maple_(software))**,**[**Mathematica**](https://en.wikipedia.org/wiki/Mathematica)**, and**[**SageMath**](https://en.wikipedia.org/wiki/SageMath)**, a computational interface style that originated with Mathematica in the 1980s.**[**[19]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-Somers2018-19)**Jupyter interest overtook the popularity of the Mathematica notebook interface in early 2018.**[**[19]**](https://en.wikipedia.org/wiki/Project_Jupyter#cite_note-Somers2018-19)

**ANACONDA**

**Anaconda is a**[**distribution**](https://en.wikipedia.org/wiki/Software_distribution)**of the**[**Python**](https://en.wikipedia.org/wiki/Python_(programming_language))**and**[**R**](https://en.wikipedia.org/wiki/R_(programming_language))[**programming languages**](https://en.wikipedia.org/wiki/Programming_language)**for**[**scientific computing**](https://en.wikipedia.org/wiki/Scientific_computing)**(**[**data science**](https://en.wikipedia.org/wiki/Data_science)**,**[**machine learning**](https://en.wikipedia.org/wiki/Machine_learning)**applications, large-scale**[**data processing**](https://en.wikipedia.org/wiki/Data_processing)**,**[**predictive analytics**](https://en.wikipedia.org/wiki/Predictive_analytics)**, etc.), that aims to simplify**[**package management**](https://en.wikipedia.org/wiki/Package_management)**and**[**deployment**](https://en.wikipedia.org/wiki/Deployment_environment)**. The distribution includes data-science packages suitable for**[**Windows**](https://en.wikipedia.org/wiki/Microsoft_Windows)**,**[**Linux**](https://en.wikipedia.org/wiki/Linux)**, and**[**macOS**](https://en.wikipedia.org/wiki/MacOS)**. It is developed and maintained by Anaconda, Inc., which was founded by Peter Wang and**[**Travis Oliphant**](https://en.wikipedia.org/wiki/Travis_Oliphant)**in 2012.**[**[8]**](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)#cite_note-8)**As an Anaconda, Inc. product, it is also known as Anaconda Distribution or Anaconda Individual Edition, while other products from the company are Anaconda Team Edition and Anaconda Enterprise Edition, both of which are not free.**[**[6]**](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)#cite_note-l1-6)[**[7]**](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)#cite_note-l2-7)

**Package versions in Anaconda are managed by the package management system**[**conda**](https://en.wikipedia.org/wiki/Conda_(package_manager))**.**[**[9]**](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)#cite_note-9)**This package manager was spun out as a separate**[**open-source**](https://en.wikipedia.org/wiki/Open_source)**package as it ended up being useful on its own and for things other than Python.**[**[10]**](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)#cite_note-10)**There is also a small,**[**bootstrap**](https://en.wikipedia.org/wiki/Bootstrapping)**version of Anaconda called Miniconda, which includes only conda, Python, the packages they depend on, and a small number of other packages.**[**[11]**](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)#cite_note-11)

The libraries used were:

1)nltk

2)pickle

3)sklearn

4)pandas

5)seaborn

6)matplotlib

7)Scipy

8)string

9)numpy

**Model/s Development and Evaluation**

**The comment\_text feature was the only independent feature used to predict the target features.The TfidfVectorizer was used to convert the texts into relevant vectors after removing the noise from the texts(punctuations,stopwords,digits etc.)**

**Code:**

# Initiating a Tfidf vectorizer

tfv = TfidfVectorizer(ngram\_range=(1,1), stop\_words='english')

x\_train\_tfv=tfv.fit\_transform(x\_train)

x\_test\_tfv=tfv.transform(x\_test)

**The dataset had multiple target variables,all of which were largely imbalanced.As a result of which the f-1 scores for the 0 and 1 classes had a significant difference between them as the algorithms had been trained mostly on one kind of class.**

**To overcome this problem,samples in equal propotions(1:1 or 1:2) of both 0 and 1 classes were extracted into another dataframe(unique to each target feature) which was then used for training the algorithms.**

**The code:**

bal1=df\_new[df\_new['malignant']==1]

bal2=df\_new[df\_new['malignant']==0]

balanced\_mal=pd.concat([bal1,bal2[:15289]],axis=0)

balanced\_mal['malignant'].value\_counts()

**The models were trained and tested for LogisticRegression and RandomForestClassifier algorithms using these dataframes only.**

**In the second stage of testing and training,all the balanced dataframes unique to different target features were merged into a new dataframe and the LogisticRegression aswell as RandomForestClassifier algorithms were trained and tested over this dataframe.**

Testing of Identified Approaches (Algorithms)

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

**The algorithms used were LogisticRegression,RandomForestClassifier,LinearSVC and SVC(kernel=’linear) was also used although the LinearSVC and SVC(kernel=’linear’) had to be removed from the code due to the following reasons:**

**1)LinearSVC had the best f1-score among all the aforementioned algorithms but it was removed because it cannot be used to predict the probability of the comments for different categories.**

**2) SVC(kernel=’linear’) was removed because it took forever to process.**

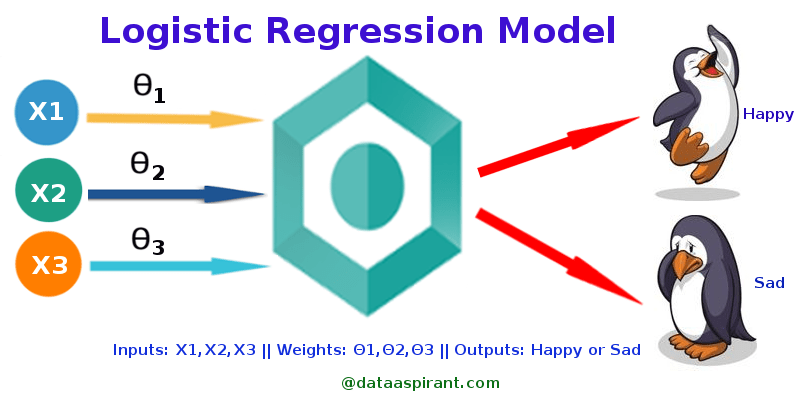
**3)The file was too big to be uploaded on github.Hence,most of the code regarding algorithms that were rejected had to be removed to overcome this problem.**

**The LogisticRegression and RandomForestClassifier did good in terms of f-1 score.**

* Run and Evaluate selected models

**The RandomForestClassifier and LogisticRegression was used as our key algorithms.**

LogisticRegression



**Logistic Regression was used in the biological sciences in early twentieth century. It was then used in many social science applications. Logistic Regression is used when the dependent variable(target) is categorical.**

**Types of Logistic Regression**

**1. Binary Logistic Regression**

**The categorical response has only two 2 possible outcomes. Example: Spam or Not**

**2. Multinomial Logistic Regression**

**Three or more categories without ordering. Example: Predicting which food is preferred more (Veg, Non-Veg, Vegan)**

**3. Ordinal Logistic Regression**

**Three or more categories with ordering. Example: Movie rating from 1 to 5**

RandomForestClassifier

**A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is controlled with the max\_samples parameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree.**

The Algorithms used Along with the snapshot of their performance on different metrics and target features:

LOGISTIC REGRESSION

Feature-malignant

0.9329965336676747

precision recall f1-score support

0.0 0.92 0.97 0.94 15279

1.0 0.96 0.89 0.92 11839

accuracy 0.93 27118

macro avg 0.94 0.93 0.93 27118

weighted avg 0.93 0.93 0.93 27118

feature- threat

0.8504098360655737

precision recall f1-score support

0.0 0.82 1.00 0.90 331

1.0 0.99 0.54 0.70 157

accuracy 0.85 488

macro avg 0.90 0.77 0.80 488

weighted avg 0.87 0.85 0.84 488

feature- abuse

accuracy: 0.9017088356073653

0.9017088356073653

precision recall f1-score support

0.0 0.89 0.98 0.93 4966

1.0 0.94 0.76 0.84 2583

accuracy 0.90 7549

macro avg 0.91 0.87 0.88 7549

weighted avg 0.91 0.90 0.90 7549

feature- loathe

accuracy: 0.8589951377633711

0.8589951377633711

precision recall f1-score support

0.0 0.84 0.98 0.90 1231

1.0 0.93 0.63 0.75 620

accuracy 0.86 1851

macro avg 0.88 0.80 0.83 1851

weighted avg 0.87 0.86 0.85 1851

feature- highly malignant

accuracy: 0.886822318955283

0.886822318955283

precision recall f1-score support

0.0 0.86 0.98 0.92 1625

1.0 0.95 0.72 0.82 902

accuracy 0.89 2527

macro avg 0.91 0.85 0.87 2527

weighted avg 0.89 0.89 0.88 2527

RANDOM FOREST CLASSIFIER

Feature- malignant

0.9550483073973007

precision recall f1-score support

0.0 0.94 0.98 0.96 15279

1.0 0.98 0.92 0.95 11839

accuracy 0.96 27118

macro avg 0.96 0.95 0.95 27118

weighted avg 0.96 0.96 0.95 27118

Feature- threat

0.8790983606557377

precision recall f1-score support

0.8975409836065574

precision recall f1-score support

0.0 0.88 0.99 0.93 331

1.0 0.97 0.71 0.82 157

accuracy 0.90 488

macro avg 0.92 0.85 0.87 488

weighted avg 0.91 0.90 0.89 488

Feature- abuse

0.9014438998542853

precision recall f1-score support

0.0 0.89 0.97 0.93 4966

1.0 0.93 0.77 0.84 2583

accuracy 0.90 7549

macro avg 0.91 0.87 0.89 7549

weighted avg 0.90 0.90 0.90 7549

Feature- loathe

0.8665586169638033

precision recall f1-score support

0.0 0.86 0.96 0.91 1231

1.0 0.90 0.68 0.77 620

accuracy 0.87 1851

macro avg 0.88 0.82 0.84 1851

weighted avg 0.87 0.87 0.86 1851

Feature- highly malignant

0.9137316976652157

precision recall f1-score support

0.0 0.90 0.97 0.94 1625

1.0 0.94 0.81 0.87 902

accuracy 0.91 2527

macro avg 0.92 0.89 0.90 2527

weighted avg 0.92 0.91 0.91 2527

* Key Metrics for success in solving problem under consideration

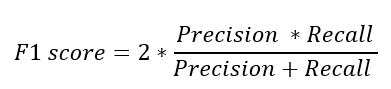
The key metric for the model selection was the F1-score.

**Justification:**

**Precision and Recall are the two building blocks of the F1 score. The goal of the F1 score is to combine the precision and recall metrics into a single metric. At the same time, the F1 score has been designed to work well on imbalanced data.**

**The F1 score is defined as the harmonic mean of precision and recall.**

**Formula:**

****

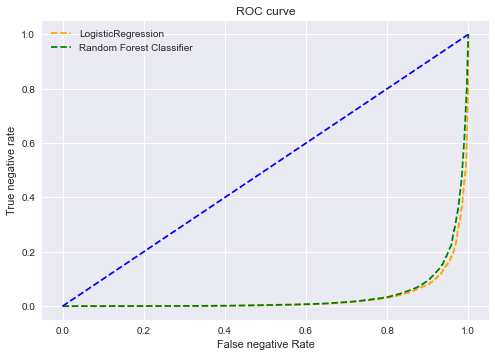
**Other metric used was roc\_curve and accuracy score,although they were not significant in selecting the best model.**

* Visualizations

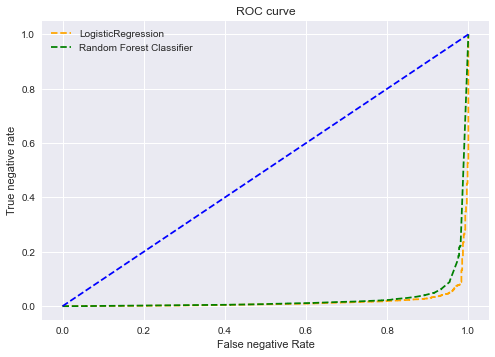
The roc\_curve was compared for RandomForestClassifier against LogisticRegression algorithm for every target variable.

The visualizations :

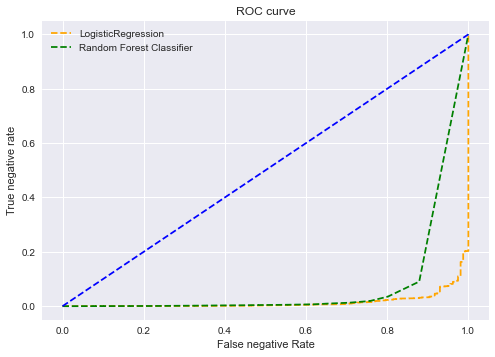
Malignant comments



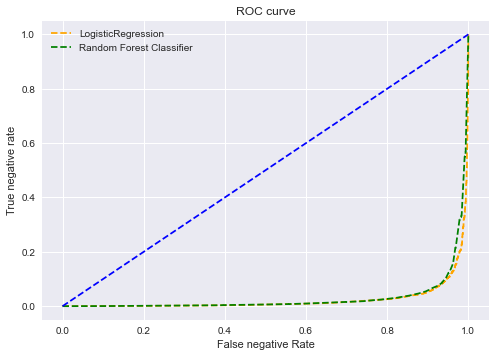
Highly malignant comments



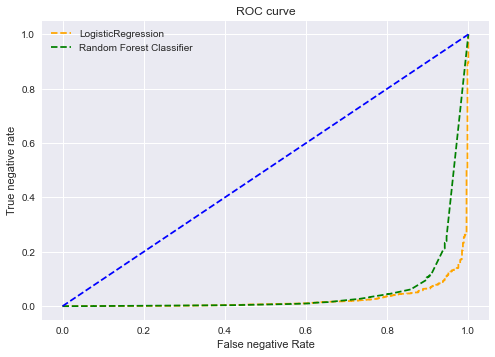
Threatening comments



Abusive comments



Loathe comments



* Interpretation of the Results

**The roc curve plotted for the different target variables depicted that Logistic Regression had a better discriminating power than the RandomForestClassifier.**

**For some features like abuse,highly malignant and malignant,the difference between the area under curve was not high indicating that both the algorithms had similar potential in discriminating between the 0 and 1 classes.However, the roc curve for the threat and loathe features depicted that logistic regression performed significantly better in identifying the 0 and 1 classes.**

**The f1-scores were considered as the deciding metrics for selecting our best model in which RandomForestClassifier with criterion=’entropy’ had performed better.**

**Hence,the RandomForestClassifier with criterion=’entropy’ was selected as the best model although hyperparameter tuning was done without using GridSearchCV as it took forever to execute.**

**The different RandomForestClassifier instances were used to save algorithms trained for different target variables and were saved by the names:**

|  |  |
| --- | --- |
| **Feature name** | **filename** |
| **malignant** | **malignant\_comments\_classifier.sav** |
| **highly malignant** | **highly\_malignant\_comments\_classifier.sav** |
| **abuse** | **abuse\_comments\_classifier.sav** |
| **threat** | **threatning\_comments\_classifier.sav** |
| **loathe** | **loathe\_comments\_classifier.sav** |

**CONCLUSION**

Finally,the saved models were used for making predictions on the test data after performing the same cleaning operations that were performed on the training dataset.The test dataset looked like:

|  | **id** | **comment\_text** |
| --- | --- | --- |
| **0** | 00001cee341fdb12 | Yo bitch Ja Rule is more succesful then you'll... |
| **1** | 0000247867823ef7 | == From RfC == \n\n The title is fine as it is... |
| **2** | 00013b17ad220c46 | " \n\n == Sources == \n\n \* Zawe Ashton on Lap... |
| **3** | 00017563c3f7919a | :If you have a look back at the source, the in... |
| **4** | 00017695ad8997eb | I don't anonymously edit articles at all. |
| **...** | ... | ... |
| **153181** | fffcd0960ee309b5 | . \n i totally agree, this stuff is nothing bu... |
| **153182** | fffd7a9a6eb32c16 | == Throw from out field to home plate. == \n\n... |
| **153183** | fffda9e8d6fafa9e | " \n\n == Okinotorishima categories == \n\n I ... |
| **153184** | fffe8f1340a79fc2 | " \n\n == ""One of the founding nations of the... |
| **153185** | ffffce3fb183ee80 | " \n :::Stop already. Your bullshit is not wel... |

A snapshot of the predicted dataset:

|  | **comment\_text** | **malignant(in%)** | **highly\_malignant(in%)** | **abuse(in%)** | **threat(in%)** | **loathe(in%)** |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| **0** | yo bitch ja rule is more succesful then youll ... | 0.940000 | 0.71 | 0.870000 | 0.54 | 0.570000 |
| **1** | from rfc the title is fine as it is imo | 0.054508 | 0.05 | 0.061597 | 0.24 | 0.155024 |
| **2** | sources zawe ashton on lapland â€” | 0.097367 | 0.02 | 0.039435 | 0.08 | 0.112984 |
| **3** | if you have a look back at the source the info... | 0.140000 | 0.10 | 0.100000 | 0.13 | 0.070000 |
| **4** | i dont anonymously edit articles at all | 0.290000 | 0.18 | 0.223305 | 0.26 | 0.048889 |
| **...** | ... | ... | ... | ... | ... | ... |
| **153181** | i totally agree this stuff is nothing but too... | 0.102466 | 0.04 | 0.170741 | 0.15 | 0.145187 |
| **153182** | throw from out field to home plate does it g... | 0.358852 | 0.16 | 0.374822 | 0.21 | 0.244000 |
| **153183** | okinotorishima categories i see your change... | 0.130000 | 0.05 | 0.041875 | 0.05 | 0.112369 |
| **153184** | one of the founding nations of the eu germa... | 0.282025 | 0.09 | 0.052222 | 0.11 | 0.402833 |
| **153185** | stop already your bullshit is not welcome her... | 0.780000 | 0.17 | 0.501333 | 0.36 | 0.317833 |

* Learning Outcomes of the Study in respect of Data Science

As I went on with the project, I learned a few interesting aswell as important facts related to the algorithms and data cleaning.

They were:

1)The inbuilt stopwords in the nltk.corpus library do not cover all kinds of words that are repetative and hold little significance in the context.Hence,customizing our own stopwords according to the given statement can be more fruitful.

2)The SVC() and LinearSVC() from the support vector machine works very well with the vectors of the text data obtained using TfidfVectorizer.However,the LinearSVC() algorithm cannot be used to predict the probability of the test data and the SVC() takes a really long time to process especially if the dataset is quite big.

3)The hyperparametertuning the randomforestclassifier using GridSearchCV can take forever to run especially if the dataset is too big.

* Limitations of this work and Scope for Future Work

The algorithms predict the categories of comments by previous examples that they were trained on similar to humans.If the algorithm encounters a problem/text different from the ones it was trained on,it is more likely to fail in classifying the type.

Eg:It is not necessary that a comment is threatning if it uses the word kill more than once or is not threatning if it does not use those words i.e there is more than one forms of threat and more than one way to express them.

“I will kill you” is threatning but “I will kill time until you get here” is not.

Therefore,it is important that the model be trained with a large varieties of malignant/abuse/threat/loathe comments so that it can identify/differentiate between the categories better and provide better results.

\*\*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*\*