­

**MALIGNANT COMMENTS CLASSIFIER USING NLP**

Submitted by:

YASH BHARDWAJ

# **ACKNOWLEDGMENT**

I whole heartedly thank our SME Sapna Verma, flip robo technologies for their support towards me to complete this project.

I also thank my family for supporting me.

Yash Bhardwaj

Contents

[**ACKNOWLEDGMENT** 2](#_Toc114782440)

[**INTRODUCTION** 4](#_Toc114782441)

[Business Problem Framing 4](#_Toc114782442)

[Conceptual Background of the Domain Problem 4](#_Toc114782443)

[Review of Literature 5](#_Toc114782444)

[Motivation for the Problem Undertaken 6](#_Toc114782445)

[**Analytical Problem Framing** 7](#_Toc114782446)

[Mathematical/ Analytical Modeling of the Problem 7](#_Toc114782447)

[Data description 8](#_Toc114782448)

[Data Preprocessing Done 9](#_Toc114782449)

[**Model/s Development and Evaluation** 11](#_Toc114782450)

[Identification of possible problem-solving approaches (methods) 11](#_Toc114782451)

[Testing of Identified Approaches (Algorithms) 12](#_Toc114782452)

[Best Model? 13](#_Toc114782453)

[Key Metrics for success in solving problem under consideration 14](#_Toc114782454)

[Visualizations 15](#_Toc114782455)

[**CONCLUSION** 20](#_Toc114782456)

[Key Findings and Conclusions of the Study 20](#_Toc114782457)

[Learning Outcomes of the Study in respect of Data Science 22](#_Toc114782458)

[Limitations of this work and Scope for Future Work 22](#_Toc114782459)

[**References** 23](#_Toc114782460)

# **INTRODUCTION**

## Business Problem Framing

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

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 un-offensive, but “u are an idiot” is clearly offensive.

## Review of Literature

Several studies have formerly investigated hate speech using neural network techniques; Badjatiya et al., used extensive experiments with multiple deep learning architectures to learn semantic word embedding to handle toxic comments identification. In another study, sentiment analysis model of YouTube video comments, using a deep neural network was proposed that leaded to 70-80% accuracy. Also, general use of different types of neural network methods for comment classification have been extensively used in recently published literature ; however, these approaches only addressed some of the task’s challenges while others still remain unsolved. Furthermore, Farag, El-Seoud reported that extensive numbers of literature have shown that supervised learning techniques have been the most frequently used methods for cyber-bullying detection. Nevertheless, other non-supervised techniques and methods have recognized to be operative on cyber-bullying recognition. Also, Karlekar and Bansal reported an increased number of personal sexual harassment and abuse that are shared and posted online. In this study, authors presented the task of automatically categorizing and analyzing various forms of sexual harassment, based on stories shared on the online forum SafeCity and used labeling levels of groping, ogling, and commenting; their results indicated that single-label CNN-RNN model achieves an accuracy of 86.5. One of the main undiscovered issues is how to identify algorithms that are able to implement high sensitivity in detection of toxic comments. Of course, identifying comment that is not toxic as a toxic can be frustrating for the users and there should be a lot of effort to form an algorithm with highest degree of sensitivities. In this paper, we implemented deep learning networks trained 3 3 Zaheri et al.: Toxic Comment Classification Published by SMU Scholar, 2020 and tuned with the full capability of current Elastic Cloud Computing (EC2) infrastructure, supporting the MXNet framework deployment model. We applied LSTM/RNN (Long-short term memory/Recurrent neural networks), optimized with MXNet to harness the capability of multiple GPUs, as one of the very recent advances method in this field in order to learn the sequential relationship between choice of vocabulary; that causes the current paper to be categorized as a novel and practical work in the area of toxic comments.

## Motivation for the Problem Undertaken

This is a huge concern as in this world, there are 7.7 billion people, and, out of these 7.7 billion, more than 3.5 billion people use some or the other form of online social media. Which means that every one-in-three people uses social media platform. This problem thus can be eliminated as it falls under the category of Natural Language Processing. In this, we try to recognize the intention of the speaker by building a model that’s capable of detecting different types of toxicity like threats, obscenity, insults, and identity-based hate. Moreover, it is crucial to handle any such kind of nuisance, to make a more user-friendly experience, only after which people can actually enjoy in participating in discussions with regard to online conversation.

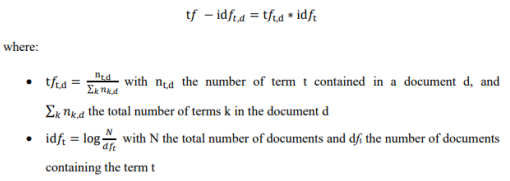
# **Analytical Problem Framing**

## Mathematical/ Analytical Modelling of the Problem

In order to apply text classification, the unstructured format of text has to be converted into a structured format for the simple reason that it is much easier for computer to deal with numbers than text. This is mainly achieved by projecting the textual contents into Vector Space Model, where text data is converted into vectors of numbers.

In the field of text classification, documents are commonly treated like a Bag-of-Words (BoW), meaning that each word is independent from the others that are present in the document. They are examined without regard to grammar neither to the word order. In such a model, the term frequency (occurrence of each word) is used as a feature in order to train the classifier.

However, using the term frequency implies that all terms are considered equally important. As its name suggests, the term frequency simply weights each term based on their occurrence frequency and does not take the discriminatory power of terms into account. To address this problem and penalize words that are too frequent, each word is given a term frequency inverse document frequency (tf-idf) score which is defined as follow:



## Data 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’.

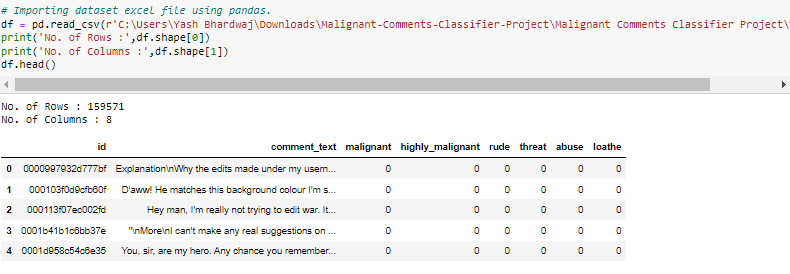
The label can be either 0 or 1, where 0 denotes a NO while 1 denotes a YES. There are various comments which have multiple labels. The first attribute is a unique ID associated with each comment.

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.

The training dataset looks like:



## Data Preprocessing Done

**Now the data is pre-processed using the following techniques:**

* Removing inverted commas and other special characters
* Removing punctuations
* Removing stop-words
* Lemmatizing
* Converting into vectors



* Hardware and Software Requirements and Tools Used

**Hardware used:**

1. Processor — AMD A9-9425 RADEON R5, 5 COMPUTE CORES 2C+3G 3.10 GHz

2. RAM — 4 GB

3. GPU — AMD Radeon(TM) R5 Graphics

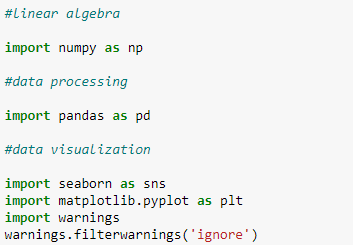
**Software utilised –**

1. Anaconda – Jupyter Notebook

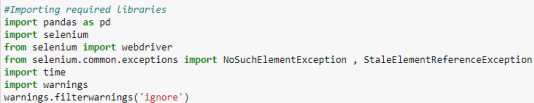
2. Selenium – Web scraping

**Libraries Used –**

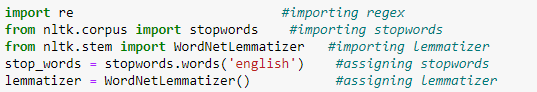
General library for data wrangling & visualisation:



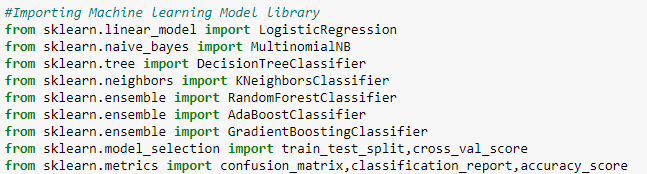
Libraries used for web scraping data from e-commerce website:



Libraries used for text pre-processing:



Libraries used for model building and its metrics:



# **Model/s Development and Evaluation**

## Identification of possible problem-solving approaches (methods)

Firstly, loading the dataset and performing data analysis like data integrity check, missing values analysis etc. Afterwards performing text mining operation to convert textual review in ML algorithm useable form. Third part of problem is building machine learning model to classify the comments with the appropriate label. This problem can be solve by as the task was to figure out whether the data belongs to zero, one, or more than one categories out of the six listed above in the data description part, the first step of the model building part is to distinguish between multi-label and multi-class classification.

In multi-class classification, we have one basic assumption that our data can belong to only one label out of all the labels we have. For example, a given picture of a fruit may be an apple, orange or guava only and not a combination of these.

In multi-label classification, data can belong to more than one label simultaneously. For example, in our case a comment may be malignant, rude and threat at the same time. It may also happen that the comment is non-toxic and hence does not belong to any of the six label.

Hence, I had a multi-label classification problem to solve.

Further, hyperparameter tuning is performed to build a more accurate model out of best model.

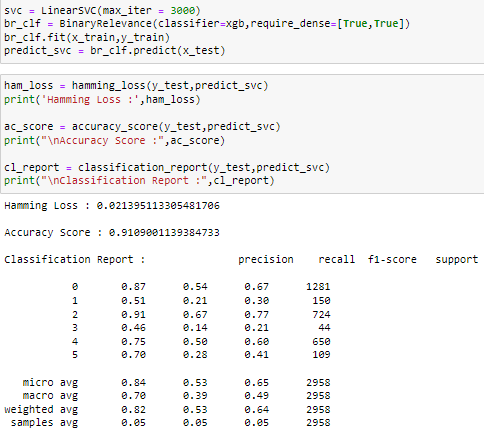
## Testing of Identified Approaches (Algorithms)

The different ML algorithms used are:

* Logistic Regression
* Ada boost Classifier
* Linear Support Vector Classifier
* XGB Classifier

## Best Model?

* In this section model was trained with above listed machine learning algorithms after splitting the data into training and testing purpose.
* All the models were compared on the basis of hamming loss, accuracy score, confusion matrix and classification report.
* We are going to select support vector classifier model as our best model because it is maximum accuracy score and least hamming loss.
* The linear support vector classifier model is giving an accuracy score of 91.09 and hamming loss of 0.02139 which fairly low.
* The model with the training and metrics looks like:



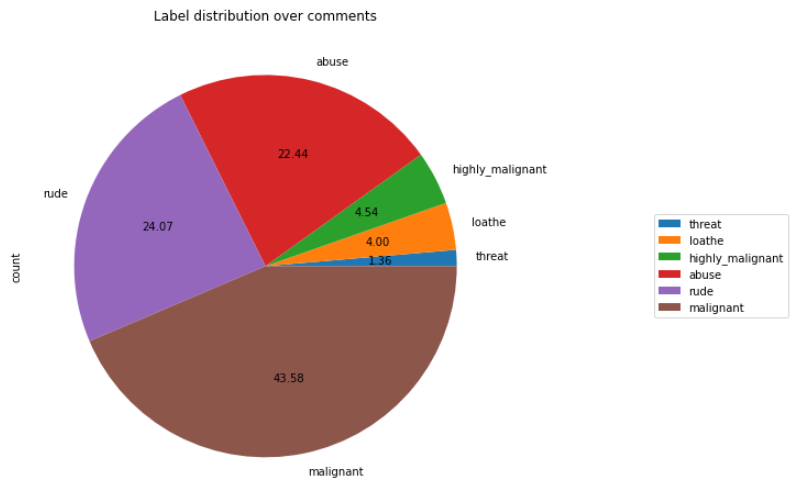
* Afterwards, hyperparameter tuning was done on the best model to optimize its performance. Below is the output for the hyperparameter tuning of the support vector classifier model:



## Key Metrics for success in solving problem under consideration

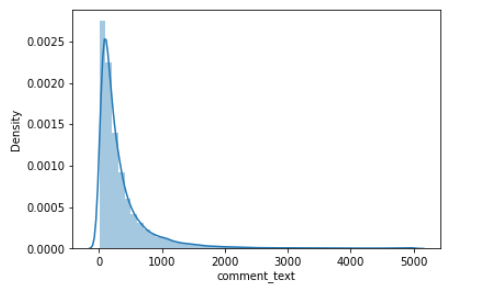
* Precision can be seen as a measure of quality; higher precision means that an algorithm returns more relevant results than irrelevant ones.
* Recall is used as a measure of quantity and high recall means that an algorithm returns most of the relevant results.
* Accuracy score is used when the True Positives and True negatives are more important. Accuracy can be used when the class distribution is similar.
* F1-score is used when the False Negatives and False Positives are crucial. While F1-score is a better metric when there are imbalanced classes.
* In multiclass classification, the Hamming loss corresponds to the Hamming distance between y\_true and y\_pred which is equivalent to the subset zero\_one\_loss function, when normalize parameter is set to True. In multilabel classification, the Hamming loss is different from the subset zero-one loss. The zero-one loss considers the entire set of labels for a given sample incorrect if it does not entirely match the true set of labels. Hamming loss is more forgiving in that it penalizes only the individual labels. The Hamming loss is upperbounded by the subset zero-one loss, when normalize parameter is set to True. It is always between 0 and 1, lower being better.

## Visualizations



Observations:

* + Out of total negative comments around 43.58% are malignant in nature followed by 24.07% are rude comments.



Observations:

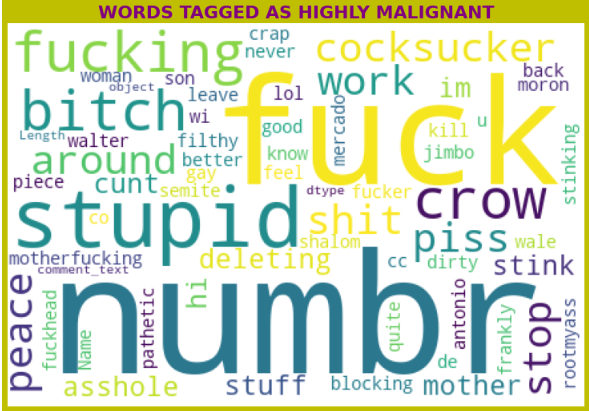
* + Above is a plot showing the comment length frequency. As noticed, most of the comments are short with only a few comments longer than 1000 words.
  + Majority of the comments are of length 500, where maximum length is 5000 and minimum length is 5. Median length being 250.

**Word Cloud:**

* Word Cloud is a visualization technique for text data wherein each word is picturized with its importance in the context or its frequency.
* The more commonly the term appears within the text being analysed, the larger the word appears in the image generated.
* The enlarged texts are the greatest number of words used there and small texts are the smaller number of words used.
* For malignant comments:



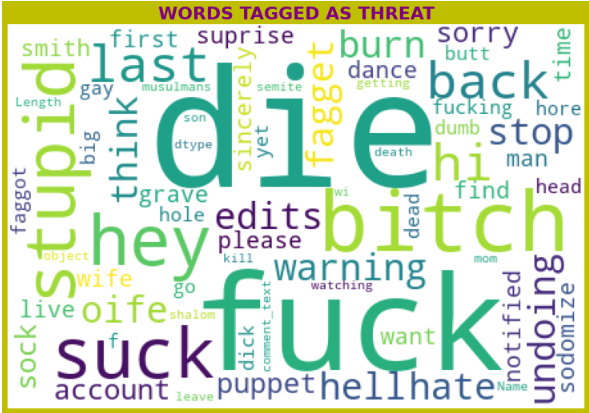
* For highly malignant comments:



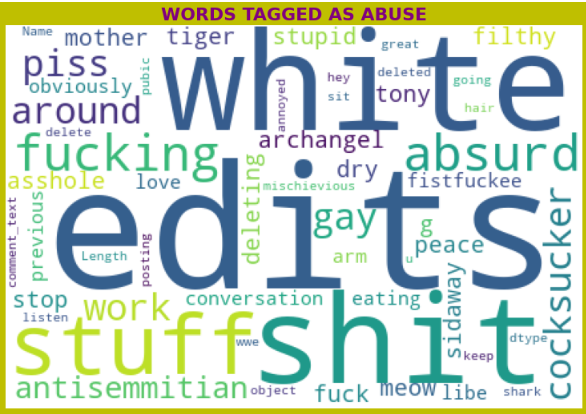
* For rude comments:



* For threat comments:



* For abuse comments:



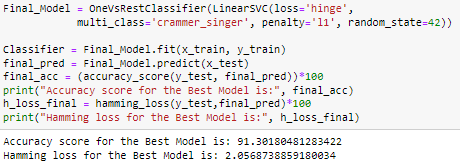
* For loathe comments:



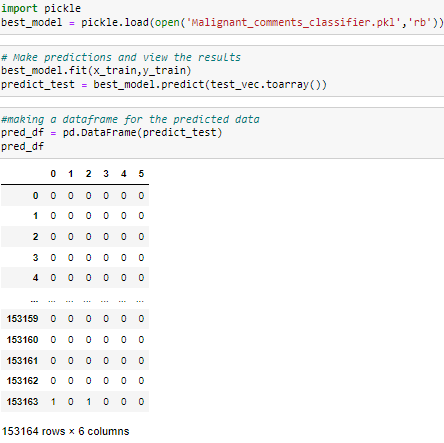
# **CONCLUSION**

## Key Findings and Conclusions of the Study

* Below is the final model i.e. after hyperparameter tuning:



* Before hyperparameter tuning the accuracy was 91.09 and hamming loss of 0.02139 but after hyperparameter the accuracy has been improved slightly.
* The Linear Support Vector Machine classifier is working extremely well and has been optimized after hyperparameter tuning giving an increased accuracy score of 91.30 and decreased hamming loss of 0.0205.
* SVM classifier is fastest algorithm compare to others.
* The final model was saved using pickle dump.
* After the saving of the final model the saved model was used to classify the comments of the test dataset using pickle load and predicted values then saved into csv format.



## Learning Outcomes of the Study in respect of Data Science

* In this project we were able to learn various Natural language processing techniques like lemmatization, stemming, removal of Stop words.
* This project has demonstrated the importance of sampling effectively, modelling and predicting data.

## Limitations of this work and Scope for Future Work

* The Maximum feature used while vectorization is 2000. Employing more feature in vectorization lead to more accurate model which is not employed due to limited computational resources.
* Data is imbalanced in nature but due to computational limitation we have not employed balancing techniques here.
* Deep learning CNN, ANN can be employed to create more accurate model.
* There is scope for application of advanced deep learning NLP tool to enhanced text mining operation which eventually help in building more accurate model with good cross validation score.
* Extensive hyperparameter tuning can result in better model.

# **References**

* <https://www.google.com/>
* <https://medium.com/@nupurbaghel/toxic-comment-classification-f6e075c3487a>
* <https://levelup.gitconnected.com/how-to-fine-tune-an-nlp-classification-model-with-transformers-and-huggingface-1a2c0ea79c2>
* <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.hamming_loss.html>
* <https://www.quora.com/What-are-the-advantages-of-LinearSVC-Support-Vector-Classifier-algorithms>
* <https://github.com/NikhilGohil/Social-Media-and-Data-Mining-Project>