PRML MAJOR PROJECT

TOXIC COMMENT CLASSIFICATION

Gagan Gandi, Hima Varshitha Nandi, Bheemesh Pujari
B21ES010, B21CS049, B21EE053

Abstract - *This paper reports our experience with building a model for classifying toxic behavior in Wikipedia comments. The data includes labeled comments that have been rated by human raters for different types of toxicity, such as toxic, severe_toxic, obscene, threat, insult, and identity_hate. The goal is to develop a model that can predict the probability of each type of toxicity for a given comment.*

I. INTRODUCTION

Toxic comments are a growing problem in online communities, particularly in social media and news websites. These comments can contain offensive, harmful or threatening language and can create a negative environment for other users. Toxic comments can also lead to cyberbullying and harassment which can have a serious impact on the mental health and well-being of the individuals targeted. To address this issue, machine learning algorithms can be used to automatically identify toxic comments and classify them into different categories of toxicity. In this report, we will discuss the process of building a machine learning model for toxic comment classification. We will start with data preprocessing, followed by model selection and evaluation. We will also explore different techniques to improve the performance of the model, such as hyperparameter tuning and ensemble methods.

II. METHODOLOGY

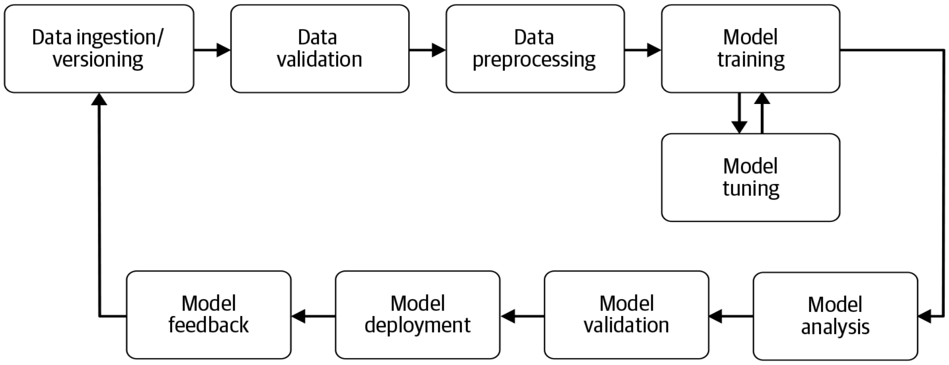


Fig. The machine learning pipeline

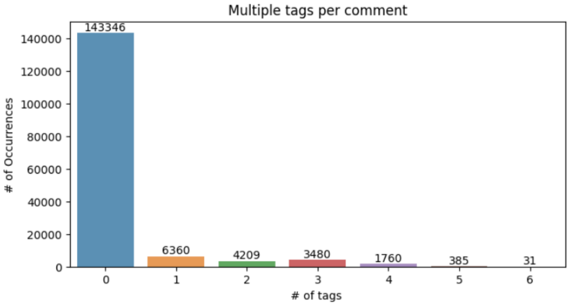
There are various classification algorithms present out of which we shall implement the following

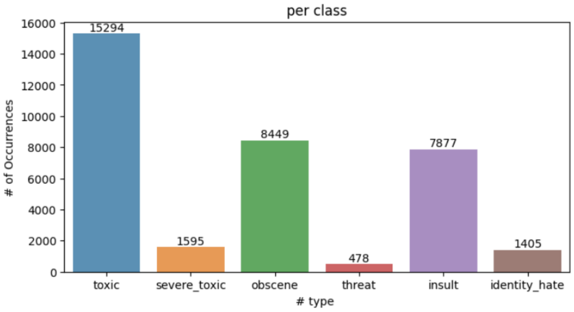
- Logistic regression
- Multinomial Naive Bayes
- Random Forest

For each of these classifiers, evaluation metrics will be calculated followed by plotting ROC curves and finding the required probabilities.

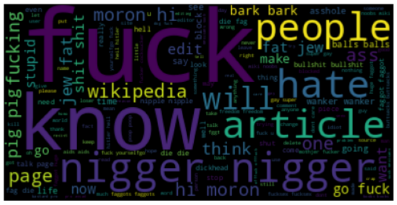
III. Data Description and Preprocessing

The dataset contains 1,59,571 rows of text comments from wikipedia with the observed toxicity level for various categories like toxic, threat, insult, identity hate, severe toxic and obscene. The bar plots are plotted to find the number of multiple tags per comment and also per class.





Cloud words are formed in toxic comment classification to identify the most frequently used words or phrases in a set of comments. The cloud word for toxic comments might include words or phrases that are derogatory, offensive, or contain profanity, while the cloud word for non-toxic comments might include more neutral or positive language. The first one is the cloud word for overall data. The following cloud words are for each of the given six categories.



shit shit
fuck
ass
go fuck
faggot faggot
die die
fuck yourselfgo
fucksex fucksex
nigger nigger
asshole

nigger nigger
fuck
bullshit bullshit
fucking
asshole
go fuck
page
ass
nipple nipple
article
know
die die
fucksex fucksex
dickhead

kill
ass
die die
going
must
die will
jim wale
wales must
fucking
will block

fat jew
hi moron
fuck
know
hate
nigger nigger
moron hi
jew fat
fucking
wikipedia
shit
go fuck

mexicans suck
suck mexicans
huge faggot
fuck
jew fat
fat jew
faggot huge
nigger nigger
die die
fucking
gay
cunt cunt

The first step of the preprocessing was to remove null elements from the dataset. Using isnull().sum() it is observed that there are no null values. The next step was the cleaning of the dataset in which punctuations, hyperlinks, numbers, hashes, non-alphanumeric, mentions and multiple white spaces were removed. Also stopwords were removed by downloading pre-created stopwords from nltk. Then the comment texts were lemmatized so as to group together inflected or variant forms of the same word using pre-trained WordNetLemmatizer model from nltk library. The final preprocessing step was to vectorize the sentences using TfindfVectorizer from Scikit-Learn library with max-features set to 5000.

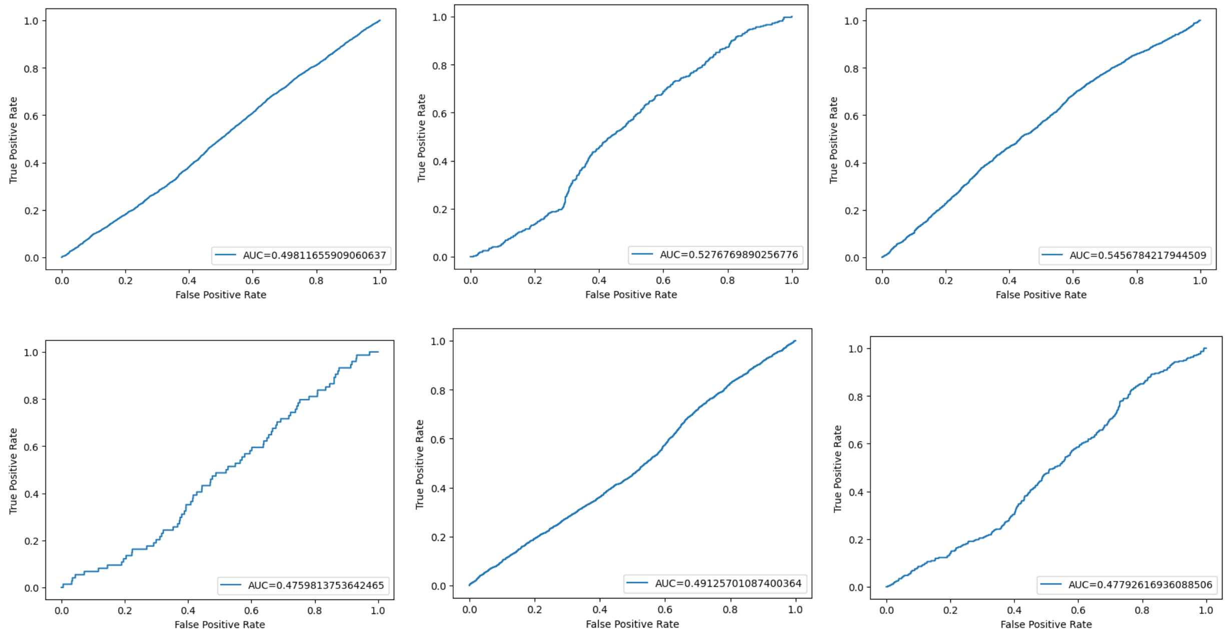
IV. Machine Learning Models

- **Logistic Regression**

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Input values (x) are combined linearly using weights or coefficient values (referred to as Beta) to predict an output value (y). A key difference from linear regression is that the output value being modeled is a binary values (0 or 1) rather than a numeric value.

Features	Accuracy (%)	AUC score
toxic	90	0.4981
severe_toxic	99	0.5276
obscene	95	0.5456
threat	100	0.4759
insult	95	0.4912
identity_hate	99	0.4779

The ROC curves for each feature are as follows:

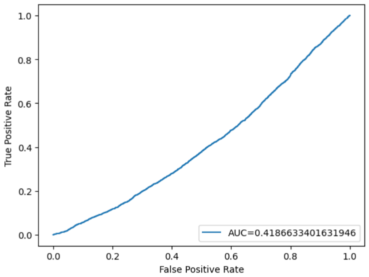


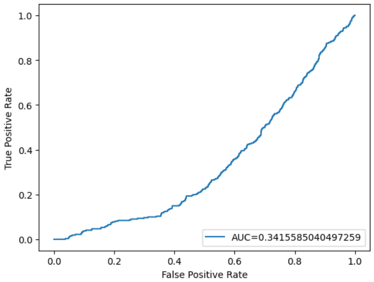
- **Multinomial Naive Bayes**

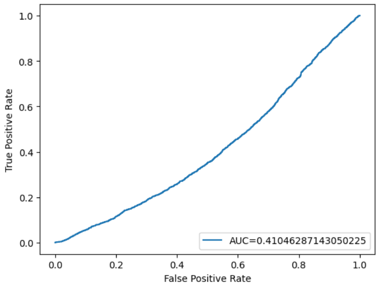
The Multinomial Naive Bayes algorithm is a Bayesian learning approach popular in Natural Language Processing (NLP). It is a collection of many algorithms where all the algorithms share one common principle, and that is each feature being classified is not related to any other feature.

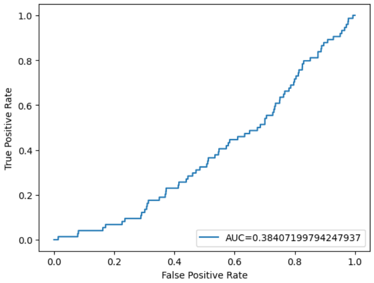
Features	Accuracy (%)	AUC score
toxic	37	0.4186
severe_toxic	65	0.3415
obscene	42	0.4104
threat	80	0.3840
insult	42	0.4019
identity_hate	67	0.3972

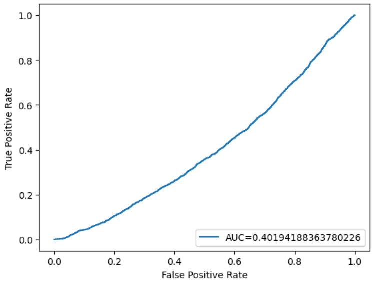
The ROC curves for each feature are as follows:

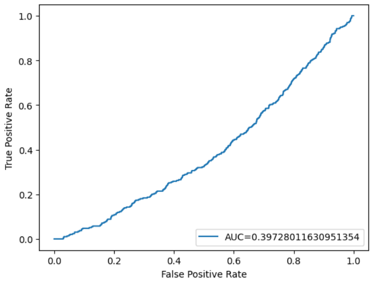










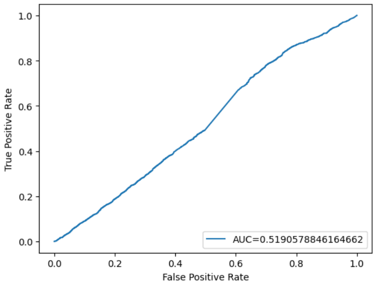


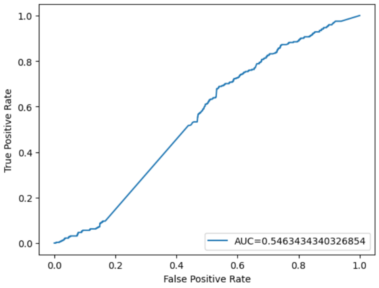
- **Random Forest**

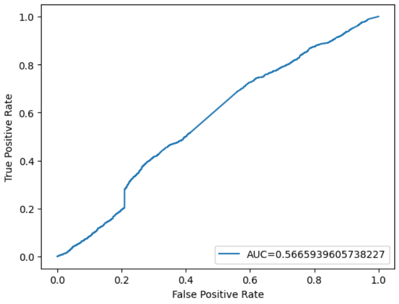
Random Forest is an ensemble machine learning algorithm that constructs multiple decision trees and combines their predictions to make a final prediction. It randomly selects a subset of data and features to prevent overfitting and is well-suited for handling large datasets with many features.

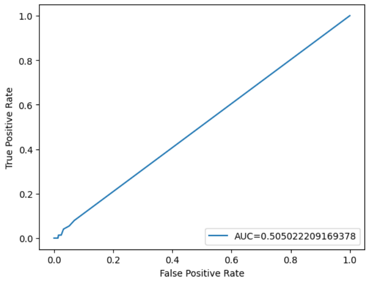
Features	Accuracy (%)	AUC score
toxic	89	0.5190
severe_toxic	99	0.5463
obscene	94	0.5665
threat	100	0.5050
insult	95	0.5089
identity_hate	99	0.5425

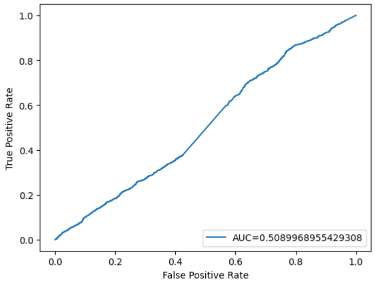
The ROC curves for each feature are as follows:

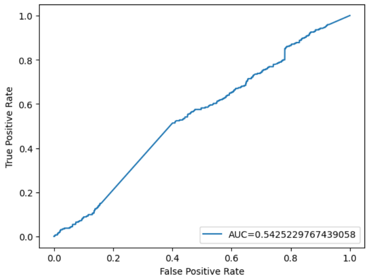












V. Comparisons and Conclusions

The advantages of logistic regressions are their properties that make them easy to interpret ; while the weakness is that they only model linear relationships between dependent and independent variables. The strengths of Multinomial Naive Bayes is that it is simple and very fast – no iterations since the probabilities can be directly computed. So this technique is useful where speed of training is important. The weakness is that Conditional Independence Assumption does not always hold. In most situations, the feature shows some form of dependency. Also the zero probability problem is prevalent. Random Forest is less prone to overfitting than some other machine learning algorithms, making it more robust and better able to generalize to new data but can be computationally expensive, particularly when dealing with large datasets or many features. This can make it less suitable for real-time applications where speed is critical.
The accuracies are almost same for Logistic regression and Random Forest but less for Multinomial NB. The AUC scores are better for Random Forest than Logistic regression and Multinomial NB.

The predicted probabilities of each type of toxicity(toxic, severe_toxic, obscene, threat, insult, and identity_hate) for a given comment are obtained in numpy arrays. These are converted into pandas dataframes and are stored in a folder that is submitted along with the code files.
Link - https://drive.google.com/drive/folders/1XE-qt6hoA5MA1pBl4zpb9HX252PyM7XI?usp=sharing

VI. Contributions

Gagan Gandi (B21ES010) - Worked on iterating different preprocessing aspects and in training various models especially Multinomial NB

Hima Varshitha (B21CS049) - Worked on exploratory data analysis and preprocessing aspects. Worked on logistic regression and making report

Bheemesh Pujari (B21EE053) - Worked on Random Forest classifier. Analyzed the classifiers, plotting ROCs and finding metrics like Precision, Recall and F1-Score.