Toxic Comment Classification Using ML

[1]Ms. J Janisha, [2]E Umesh Chandhra, [3]Gadhi Rajesh, [4]G T V Satyanarayana, [5]G Naveen Sai Manikanta

[1]Assistant Professor, [2,3,4,5]Student
Department of Computer Science and Engineering
Bharath Institute of Higher Education And Research
Chennai, India-600073

***Abstract*- Toxic users are those that leave a debate because of insulting, violent, aggressive, or unreasonable remarks. In the modern age, social media has permeated every area of people's lives. There are several reasons why people are bullied. Some people use the internet as a means of venting their resentment, anxieties, and biases, while others would like to engage in civilized conversation. This kind of antisocial behaviour is often displayed in online debates, discussions, and skirmishes when insulting and rude remarks—also known as poisonous remarks—are exchanged. Comments containing explicit language might fall into a wide range of categories, such as Identity Hate, Obscene, Threat, Severe Toxic, and Toxic. Many give up trying to find solutions and stop expressing themselves since they fear being mistreated and harassed.**

Keywords: **Natural Language Processing, Word Embeddings, Linear Regression , XG boost, MNBM.**

I. INTRODUCTION

The People are using the internet more and more to express their thoughts, worries, and feedback in various online forums, which has increased people's active participation in these forums. Even while these remarks are often beneficial, occasionally they can be hurtful and incite animosity in others. We must filter these comments above all else in order to prevent the spread of negativity or hatred among people, since they are publicly accessible and viewed by individuals from a diverse range of social groups, age groups, communities, and socioeconomic backgrounds. Given the variety of data gathered, pre-processing techniques, and models, identifying toxic comments has proven to be extremely difficult.

II. Related Work

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” [1], 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 (m-SVM) to achieve approximately 80% accuracy with data from various social media companies [2]. Another paper published in 2018 utilizes various word embeddings to train a CNN_GRU model, achieving 90% accuracy on 3 different classes [3] 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 3 Hate, Bias and Extremism to create what it calls the Global Network Against Hate" [4], for which Facebook will invest $500,000 to spot online extremism and countering methods.

III. DATA DESCRIPTION AND PREPROCESSING

The dataset contains text comments collected from social media posts and the target toxicity and additional toxicity subtype attributes which are rated as continuous values between 0 and 1. We used data from the “Toxic Comment Classification Challenge” on Kaggle, which contains comments from Wikipedia editors labelled by human volunteers individually [5]. Comments are labelled in 6 categories: toxic, severe toxic, obscene, threat, insult, and identity hate. For example, if a comment is labelled as 100100, it is toxic and threat. However, we found that the labels are not accurate for some of the comments, which we manually relabelled.

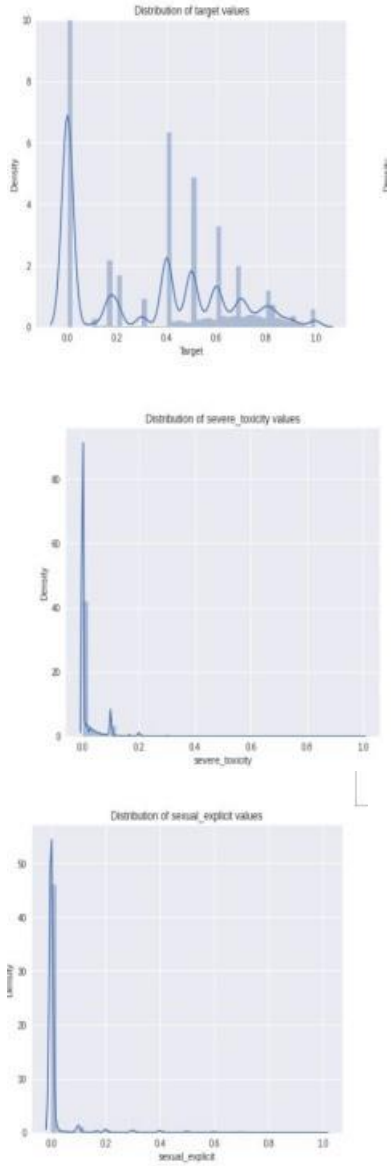


Fig. 1. a. Distribution of Target Toxcity, b. Distribution of severe toxicity, c. Distribution of Sexual Explicit

2. Data Preprocessing

A. Performing NLP

Converted the whole comments into lower case.
Replaced punctuations and other unnecessary elements with spaces.
Dropped links (like https, html, etc) and emails. stop words were removed using the NLTK library and lemmatized the text using the same NLTK library.

- Finally, cleaned posts were stored in a separate column to compare the original and pre-processed texts.

B. Labels transformation

The given values of the labels were continuous as the "target toxicity" and other toxicity subtype attributes were rated in range of 0 and 1 as continuous values. So, the labels are converted into binary classes by considering the threshold value of 0.5, as most of the toxic comments have target toxicity value greater than 0.5. So, the labels with continuous value greater than equal to 0.5 are replaced with 1 and less than 0.5 are replaced with 0. ***C. Data Splitting***
Splitted the cleaned data into train and validation data in the ratio 75:25 so that the model can be trained on train data and validated using validation data.

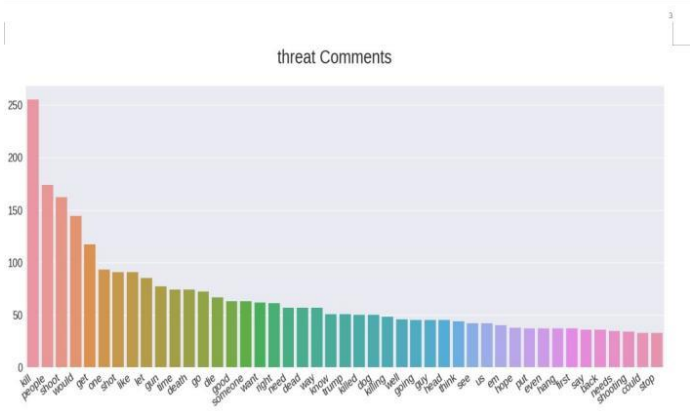


Fig. 2. Visualization of data

1. ***MACHINE LEARNING MODELS***

A. Multinomial Naive Bayes

In Natural Language Processing (NLP), the Multinomial Naive Bayes method is a common Bayesian learning approach. Using the Bayes theorem, the programme estimates the tag of a text, such as an email or a newspaper piece. It assesses the likelihood of each tag for a given sample and returns the tag with the highest probability.

Nave Bayesian classification [20] is a probabilistic approach to machine learning. It is based in the Bayes Theorem (Equation 1). P(A|B) = P(B/A) P(B)/P(A) (1)

The probability of A happening knowing that B has occurred could be calculate. Event B represents the evidence and A the hypothesis to be approved. The theorem runs on the assumption that all predictors/features are independent and the presence of one would not affect the other. This is the Bayes approach naive simplification. The probability of one event, B, is independent from another B event occurring. The approach to classify an Internet comment as offensive or toxic would begin by study our collection of training data labelled as toxic and non-toxic. From the knowledge gained in the initial assessment of used data, the probability of whether that new comment is toxic or non-toxic was calculated. As an example,

P (message is toxic message content: Hi! I am back again!)

(2)

Applying the Bayes Theorem to calculate the probability that a new message is toxic according to the content of the message was then estimated by:

P (message is toxic/message content) = P(message content/toxic)P(Toxic) /P(message content) (3)

The probability that a message is toxic, P(Toxic), was then calculated based on the proportion of toxic messages occurred in training data set. As described in Data Science in R [6], there is no requirement to calculate the probability of the message content, P(message content), the probability that the new messages content is found in words pool that are toxic is required, based on the naive simplification of the Bayes Theorem. The naive assumption leads to calculate the likelihood of the new message being toxic as the product of the word appearing or not appearing in the toxic words pool. Likewise, the likelihood of the new message being non-toxic can be calculated, and the classification of the new message would depend on the likelihood ratio (Equation 4).

P (message is Toxic/ message content)/P(message is non- Toxic /message content) (4)

Additionally, further simplifying for this ratio could be achieved by a series of summations taking the log scale:

Log (P (message is toxic /message content) /P(message is non-toxic/ message content)) (5)

To avoid an obvious division by zero, a 0.5 scalar was added to the numerator and denominator.

Log (P(message is toxic/ message content) +0.5/ P(message is non-toxic/ message content) + 0.5) (6)

B. Linear SVC

The Linear Support Vector Classifier (SVC) approach uses a linear kernel function to classify data and works well with huge datasets. When compared to the SVC model, the Linear SVC adds additional parameters such as penalty

1(i.e, C=1) and normalization (L1 or L2), and loss function. It is designed to fit to the data you provide and provide a ”best fit” hyperplane that divides or categorises your data. Following that, you may input some features to your classifier to check what the ”predicted” class is after you’ve obtained the hyperplane. we got an accuracy of 94.82 by using this model

C. XGBoost Classifier

It is an ensemble learning technique which builds a strong classifier by combining different weak classifiers. At first a model is built using training data and the second model is built in a way that it tries to correct the misclassification done in first and this keeps on repeating till the training data is correctly classified or till maximum number of models is reached. XG Boost classifier was used to train the model and the model was evaluated. we got an accuracy of 93.72 using this model

D. Logistic Regression

Logistic Regression is a classification system based on Machine Learning. It’s a method for predicting a categorical dependent variable from a set of independent variables. Solver used was “sag” as it performs well when there is a large dataset and the cost function used was “sigmoid function” which converts the predicted values into the probabilities between the range 0 and 1
, we got accuracy of 94.41 using this model

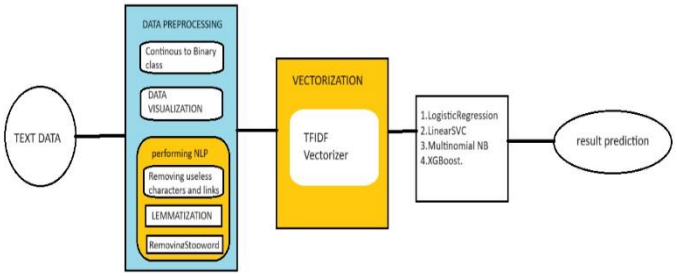


Fig. 3. whole architecture of the project.

3. MODEL BUILDING

A. Pipeline

A pipeline was created which takes the comment texts and performs pre-processing using a class which uses sklearn base library and performs NLP on the text comments and sends the output of the pre-processed text to the vectorizer which was created using TFIDF vectorizer which vectorizes the comments text and sends the vectorized text comments to the Machine learning model.

B. Model training

The selected machine learning models (i.e., Logist Regression, Linear SVC, Multinomial Naive Bayes, XGBoost) were trained using training data by creating separate pipeline for each model and reported the performance for each model.

During the process of training the models along with the “target toxicity” other additional toxicity subtype attributes like obscene, identity attack, insult, threat and others were also predicted to classify the toxic comment more precisely based on subtype attributes. The accuracies reported finally were the values of “target toxicity” and different models were compared based on the same label.

4. CLASSIFIER COMPARISION

We have compared the four different Machine Learning models (Random Forest, Linear SVC, MULTINOMIAL Naive Bayes, XG Boost). For these models, along with accuracy, measures like precision, recall, F1-score were compared. The comparision is shown below:

Classifier	Accuracy%	Precision	Recall	F1-Score
Logistic Regression	94.41	0.9389	0.9441	0.9339
XG Boost	93.72	0.9337	0.9372	0.9203
Linear SVM	94.82	0.9428	0.9482	0.9428
Multinational Navies Bayes	92.49	0.9233	0.9249	0.892

ROC CURVES and COMPARISION of ACCURACIES

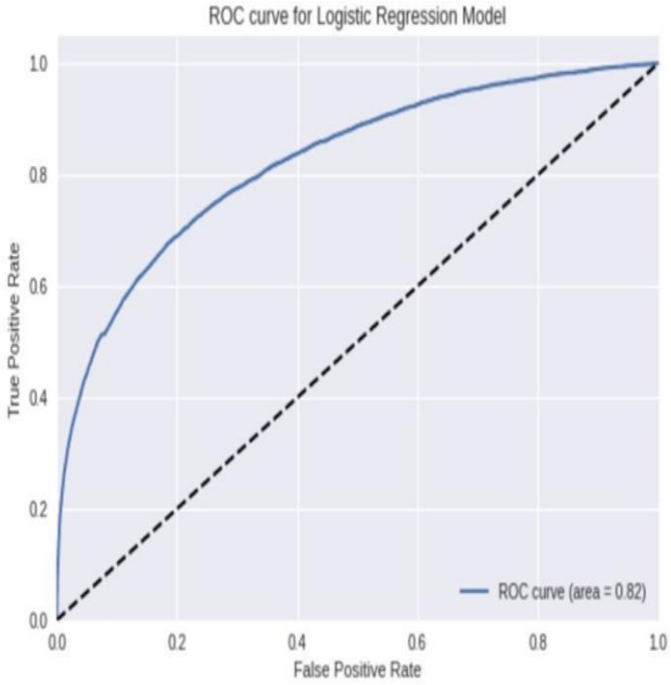


Fig. 4. ROC Curve for Linear Regression Model on cross-validation

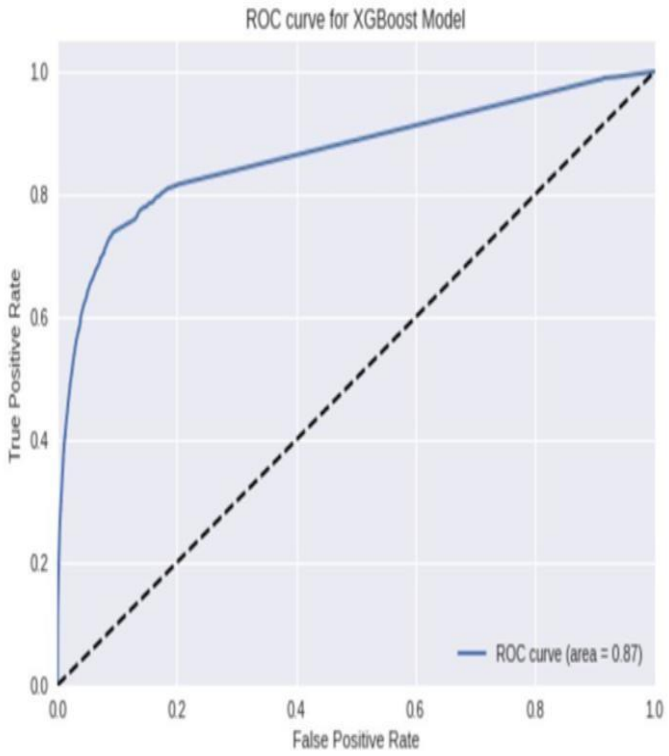
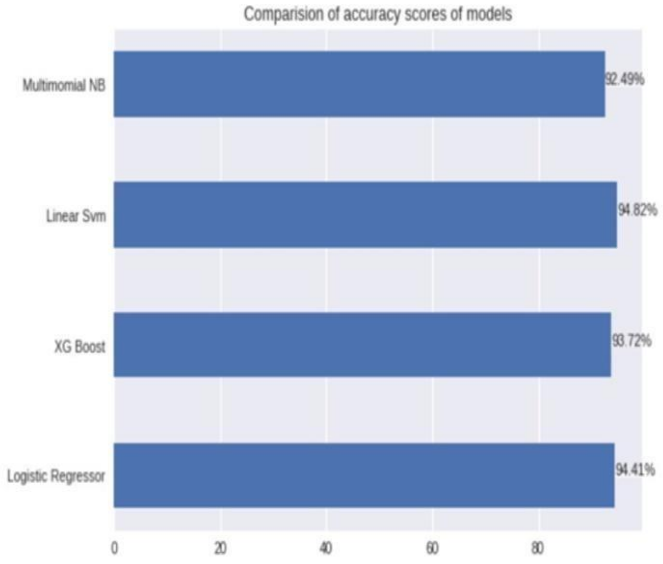


Fig. 5. ROC Curve for XG Boost model on cross-validation



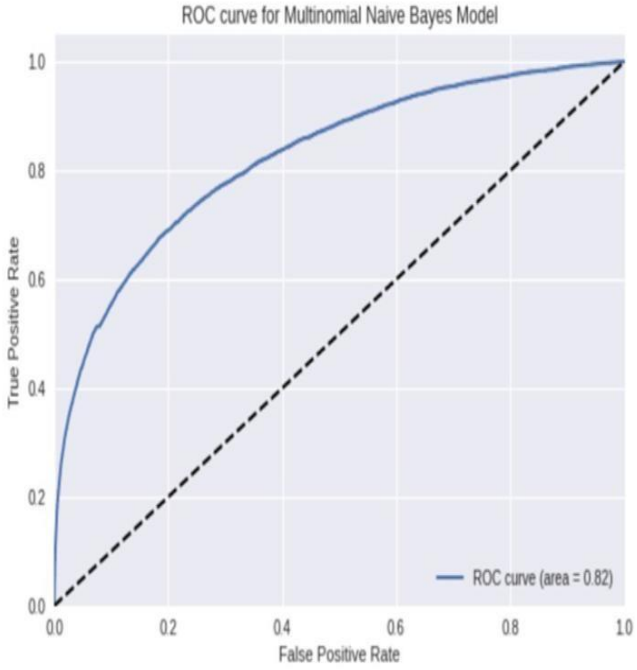


Fig. 6. The comparison of accuracies of various ML
Fig. 7. ROC Curve for Multinomial Naive bayes Model on cross-validation

5. CONCLUSION

In order to classify harmful remarks, this paper evaluates the effectiveness of many machine learning models and suggests an ensemble method known as LSTM-CNN. Extensive experiments are conducted to examine the impact of a balanced dataset and an imbalanced dataset utilizing random Underandoversampling on the model performance. The feature vector for the models' training is obtained using two feature extraction techniques, one of which is TF-IDF a. The balanced dataset tends to improve the classification accuracy, whereas the unbalanced dataset shows poor model performance. The suggested RVVC and RNN deep learning models outperform the machine learning classifiers like SVM, RF, GBM, and classifiers. The comparision table of the models shows that all the classifiers had nearly equally

efficient performance and among all the classifiers used, Linear SVC model was giving result in less time and performance of it was also little better compared to other models. As the dataset contains more data samples with less toxicity compared to the data samples with high toxicity due to which measures like F1 score,precision and recall values were less accurate. We can conclude that Linear SVC model is preferred.

REFERENCES:

1. T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated Hate Speech Detection and the Problem of Offensive Language," dissertation, 2017.
2. S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, "Hate speech detection: Challenges and solutions," PLOS ONE. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371%2Fjo urnal.pone.0221152. [Accessed: 28-Oct-2020].
3. Z. Zhang and L. Luo, "Hate speech detection: A solved problem? The challenging case of long tail on Twitter," Semantic Web, vol. 10, no. 5, pp. 925–945, 2019.
4. Thompson, "Facebook partners with Ontario university on 'global network' to counter rise in online hate | CBC News," CBCnews, 28-Jul-2020. [Online]. Available: https://www.cbc.ca/news/politics/facebook-hate-onlineextremism-1.5664832. [Accessed: 28-Oct-2020].
5. "Toxic Comment Classification Challenge"Kaggle ,2017. [Online]. Available:
6. https://www.kaggle.com/c/jigsaw-
    i. toxic-comment-classificationchallenge/overview. [Accessed: 28-Oct-2020].
    ii. Deborah Nolan and Duncan Temple Lang. Data Science in R: A Case Studies Approach to Computational Reasoning and Problem Solving. CRC Press, 2015.