Multi Label Toxic Comment Classification using Machine Learning Algorithms

Abhishek Aggarwal, Atul Tiwari

Abstract*: *Toxic comments are the comments found in the online forums that are rude, offensive, or unfair and usually cause many users to exit the conversation. The threat of bullying and abuse on the internet obstructs the free exchange of ideas by limiting people's opposing viewpoints. Most of the Websites fail to successfully facilitate healthy conversations, leading them to either restrict or disable user comments entirely. This paper would explore the scope of online abuse and categorize them into different labels to assess the toxicity as accurately as possible using machine learning algorithms.

Keywords*: *Accuracy, Multilabel Classification, Machine Learning Algorithms, Toxic Comments

I. INTRODUCTION

One of the best inventions of the twenty-first century is that one person can connect with another person anywhere in the world using only a smartphone and the internet, thanks to the rapid growth of computer science and technology.

People only used email to communicate with one another in the early days of the internet, and it was flooded with spam. It was difficult to distinguish between authentic and spam emails back then. However, communication and data flow across the internet have evolved dramatically over time, especially with the introduction of social networking websites like Facebook and Reddit. Hence, it is becoming critical to categorize posts as positive or negative to avoid societal damage and save individuals from engaging in antisocial behavior.

Authorities have recently made several arrests as a result of people's toxic and dangerous online posts. Last Year, the Vadodara police arrested a popular YouTube figure, Shubham Mishra, after he uploaded a video threatening the stand-up comedian Agrima Joshua to his thousands of YouTube subscribers. Moreover, in January of 2021, after allegedly inciting the Washington riots, Donald J. Trump was banned from nearly all social networking sites. As a result, there is a troubling situation, and it is critical to spot certain material before it is uploaded. And these harmful contents are making the internet a dangerous environment for users.

Manuscript received on May 07, 2021.
Revised Manuscript received on May 15, 2021.
Manuscript published on May 30, 2021.
* Correspondence Author
Abhishek Aggarwal, Bachelor of Technology in Electrical Engineering, Delhi Technological University, Delhi, India.
Atul Tiwari*, Bachelor of Technology in Electrical Engineering, Delhi Technological University, Delhi, India.

Let's say somebody makes a statement on the internet," Bullshit? Back up wanker. I'll have your account terminated". The derogatory connotation of terms like 'Bullshit' and 'Back up wanker' is clearly toxic. However, this statement will first undergo a specific process known as pre-processing and after which a classification algorithm will be used in order to extract the toxicity.

We will use various classification techniques and machine learning algorithms on the dataset to solve the toxic comment classification problem and compare them based on hamming loss, log-loss, and accuracy

II. RELATED WORK

Toxic comment classification has been extensively studied in recent years, especially in the context of social media, where researchers have used various machine learning algorithms to classify toxic comments found on social media forums into different toxic classes.

In [1], authors have used supervised learning for identifying harassment. To train a model for detecting toxic posts in chat rooms and discussion forums, authors combined local features, contextual features, and sentiment features.

In [2], Ravi used machine learning algorithms to find the toxicity of comments found on social media and achieved an accuracy of 82 percent using the WEKA machine learning toolkit.

In [3], authors used a semi-supervised approach to detect profanity-related offensive content on Twitter. They achieved a 75.1 percent TP rate with Logistic Regression and a 69.7 percent TP rate with popular keyword matching baseline. The false-positive rate was identical for both at about 3.77 percent.

In [4] authors used an automatic flame detection system that applies multilevel classification and extracts features at various conceptual levels.

In [5], the authors used SVM and Naïve Bayes classifiers for detecting abusive text messages and images on social networking websites. However, they were not able to detect offensive videos or audios on social networking websites.

In paper [6], For the classification, authors have used the neural network-based approach and the non-neural

network-based approach. They used the Naïve Bayes algorithm combined with logistic regression for the non-neural based approach. They achieved good accuracy, but the F1 score for this technique is very low. However, the neural networkbased model (RNN stacked and bidirectional) outperformed the non-neural based model in terms of F1 score and accuracy.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication

In paper [7], authors used Convolutional Neural Networks (CNN) over the traditional Bag of words (BoW) text

classification technique. The result showed that the use of CNN with word embedding outperformed the standard BoW text classification model, which employs SVM, KNN, NB, and LDA methods. Furthermore, CNN achieved an accuracy of more than 90 percent.

In paper [8], To protect teenagers from online harassment on YouTube, they used lexical and parser feature together to detect the toxicity in the comments section of YouTube.

Unfortunately, the prevalence of toxicity on the internet is having a negative impact on people's lives [9]. As a result, we need to find a method for detecting the toxicity of comments efficiently. In our paper, we will use techniques to break the multi-label problem into several single-label problems, allowing us to use existing single-label machine learning algorithms

III. PROPOSED METHODOLOGY

A. Type of Classification

In this paper, we will classify the given dataset (comments written by a user in an online forum) provided by Kaggle in six labels, i.e., toxic, obscene, identity hate, severe toxic, threat, or insult.

The next step is to determine if the given data (comment) belongs to one or more than one or none of the mentioned six labels. For example, the given comment can be toxic and insulting, hence falling into more than one label, but the comment can also be non-toxic and not fall into any of the six labels.

Before we begin, we should first understand the difference between multi-class and multi-label classification.

The classes in multi-class problems are mutually exclusive, meaning that each input is assigned to only one label. So, for example, your phone operating system could be android or iOS, but it cannot run both simultaneously.

However, Multi-label classification assigns each input a set of target labels, i.e., the input may be assigned to several labels at the same time.

So, we may conclude that toxic comment classification is a multi-label classification problem.

B. Exploratory Data Analysis

Exploratory data analysis is a crucial step in the data analysis process. The main aim of EDA is to gain a better understanding of the given data and to analyze their key characteristics. This is achieved by using data visualization techniques.

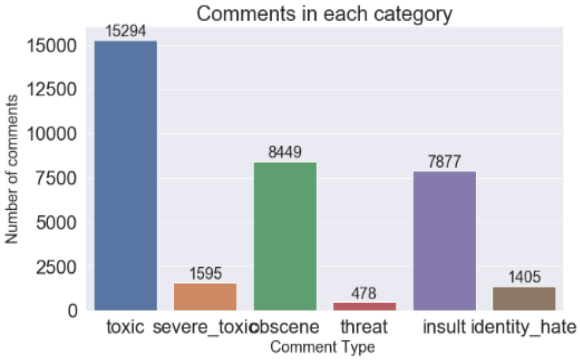

Fig. 1. Plot 1

Plot 1 depicts the number of comments that fall under each label. It can be observed that the bulk of the comments fall into the toxic category, and the threat category has the least number of comments.

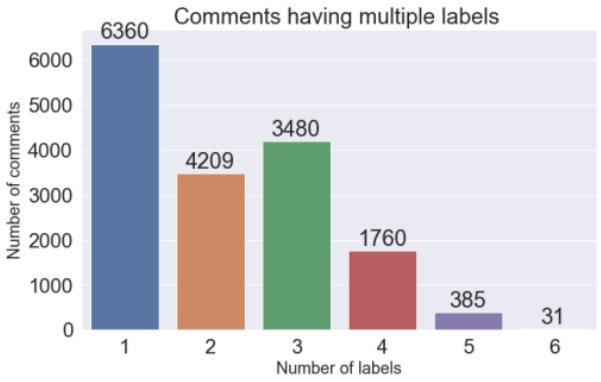

Fig. 2. Plot 2

Plot 2 shows the number of comments having multiple labels.

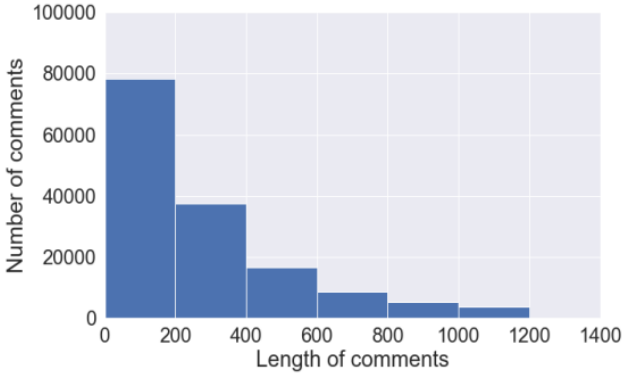

Fig. 3. Plot 3

Plot 3 depicts the number of comments of different lengths. It can be observed that the length of the comments ranged from less than 100 characters to more than 1200 characters. However, the length of most comments is less than 200 characters.

After conducting the exploratory data analysis, we have concluded that for pre-processing, we select comments with less than 400 words.

C. Data Pre-Processing

Data pre-processing is a technique used to transform the raw data into an understandable and readable format to make it suitable for building and training Machine Learning models. For our dataset, this can be achieved in 2 stages: (1) Data Cleaning (Removal of unnecessary elements from our text); (2) Feature Engineering (extracting features from data and transforming them into formats that are suitable for Machine Learning algorithms).

Retrieval Number: 100.1/ijrte.A58140510121
DOI: 10.35940/ijrte.A5814.0510121

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication

Steps for Data (Text) Cleaning:

- Removing Punctuations and other special/ non-ASCII characters.
- Splitting the comments into individual words.
- Removing Stop Words.
- Stemming and Lemmatising.
- Stemming and Lemmatising.

Hence, we get the comments as lists of clean tokens, and now we need to convert each of those comments into a vector through feature engineering to make them suitable for the SciKit Learn's algorithms.

The next step is to extract features using two techniques: **Count Vectorization and TF-IDF Transformation.**

	Word 1 Count	Word 2 Count	...	Word N Count
Message 1	0	1	...	0
Message 2	0	0	...	0
...	1	2	...	0
Message N	0	1	...	1

Fig. 4. Bag of words model using Count Vectorizer

Now our dataset is ready for train-test split and we can run it in any suitable Machine Learning model.

D. Finalizing Evaluation Metrics

Now that our data is pre-processed, the next step is to apply machine learning algorithms to them. But before applying any algorithms, we must first decide the proper evaluation metrics. Because machine learning algorithms' effectiveness is calculated using evaluation metrics. There are two main types of metrics for multi-label classification:

Label-based metrics: These are evaluated separately for each of the labels and then averaged for all of them without considering the labels' relationships. E.g., one-error, average precision, etc.

Example-based metrics: These are calculated for each example and then averaged across the test set. E.g., accuracy, log-loss, hamming-loss, etc.

An important observation is that our data is skewed; that is, the majority of the comments in our dataset are non-toxic. So, we cannot use accuracy as our only measure. For example, 92 percent of the comments in our dataset are non-toxic, which means even if we apply a basic machine learning algorithm that predicts the non-toxic value for all the comments would also result in 92 percent accuracy. As a result, selecting the metric that will determine the loss will be a safer alternative. Hence, we will use Hamming-Loss and Log-Loss along with accuracy as our evaluation metrics to compare the performance of various models in our machine learning algorithms.

E. Applying Multi Label Classification Techniques

The majority of conventional machine learning algorithms are designed for classification problems with single-label. Hence, we'll use techniques to break the multi-label problem into several single-label problems, allowing us to use the existing conventional machine learning algorithms.

1. **Binary Relevance Method:** The interdependence of labels is not taken into account in this process. Each label is solved separately, like a single-label classification problem.

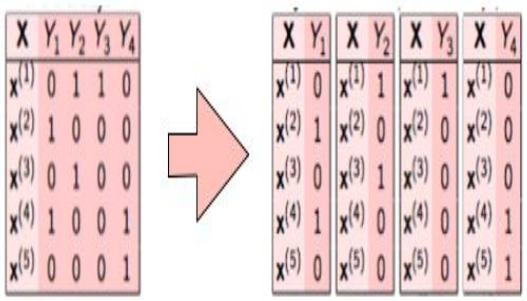

Fig. 5. Binary Relevance Method

2. **Classifier Chain Method:** We train the first classifier on the given data in this method, followed by each subsequent classifier being trained on the previous classifier and the input space, and so on. Hence, this approach considers the interdependence of labels and input data. Some classifiers may show dependence, such as toxic and severely toxic.

Fig. 6. Classifier Chain Method

3. **Label Power Set Method:** This approach takes all possible label combinations into account. As a result, any specific combination can be used as a label, breaking our multi-label problem into a multi-class classification problem.

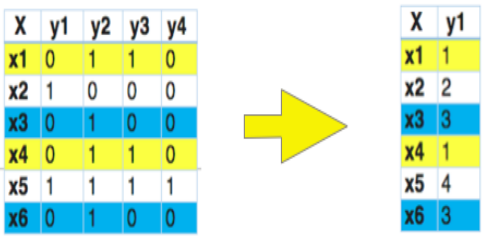

Fig. 7. Label Power Set Method

With each of these methods, we will use five machine algorithms to get optimal results.

1) Multinomial Naïve Bayes
2) Random Forest Classifier
3) Bernoulli Naïve Bayes
4) Nearest Centroid
5) Ridge Classifier

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication

IV. RESULT AND ANALYSIS

We used three methods, i.e., Binary Relevance, Classifier chain, and Label power set for each of the machine learning algorithms and the results for each machine learning algorithm are shown below with Log-Loss, Accuracy, and Hamming-Loss as our evaluation metrics.

	Hamming_loss	Accuracy	Log_loss
Binary Relevance	3.861695	88.290936	1.901773
Classifier Chain	3.616800	88.695013	1.374349
Label Powerset	4.167815	88.300119	0.536172

Fig. 8. Multinomial NB

	Hamming_loss	Accuracy	Log_loss
Binary Relevance	3.838736	88.235834	2.174770
Classifier Chain	3.798941	88.281752	2.014295
Label Powerset	4.317813	88.143999	0.538714

Fig. 9. Bernoulli NB

	Hamming_loss	Accuracy	Log_loss
Binary Relevance	2.398445	90.118468	1.982613
Classifier Chain	2.401506	90.146019	1.878685
Label Powerset	2.687728	89.925613	1.241680

Fig. 10. Random Forest Classifier

	Hamming_loss	Accuracy	Log_loss
Binary Relevance	3.453026	87.987878	1.336500
Classifier Chain	6.027490	87.060336	0.465341
Label Powerset	2.687728	89.925613	1.241680

Fig. 11. Nearest Centroid

	Hamming_loss	Accuracy	Log_loss
Binary Relevance	2.546913	90.118468	1.866874
Classifier Chain	2.528546	90.210304	1.536788
Label Powerset	3.093336	89.484801	1.226348

Fig. 12. Ridge Classifier

V. CONCLUSION

This paper has discussed three approaches to implement various machine learning algorithms and compared their Log-Loss, accuracy, and Hamming-Loss. After proper review, we may conclude there is no single best approach to solve the problem. Instead, each algorithm has got its own best approach for optimal results. However, if we look at the time complexity of the algorithms, Random forest is not suitable for this data set as other algorithms give almost the same results in lesser time.

In further research, we can use algorithm adaptation methods that transform the algorithms to perform multi-label classification directly. Furthermore, we can also experiment with more complex deep learning algorithms like CNN (convolutional neural network), MLP (multilayer perceptron), and RNN (Recurrent neural networks) in the near future as we believe our approach could reach the top performance when combined with deep learning models.

REFERENCES

1. Yin, Dawei, Xue, Zhenzhen, Hong, Liangjie, Davison, Brian, Edwards, April, Edwards, Lynne. (2009), "*Detection of harassment on Web 2.0*"
2. Ravi, P. (2012), "*Detecting Insults in Social Commentary*".
3. Guang Xiang, Bin Fan, Ling Wang, Jason Hong, and Carolyn Rose. 2012, "*Detecting offensive tweets via topical feature discovery over a large scale twitter corpus*". In Proceedings of the 21st ACM international conference on Information and knowledge management (CIKM '12). Association for Computing Machinery, New York, NY, USA, 1980–1984. DOI: https://doi.org/10.1145/2396761.2398556.
4. Razavi, A.H., Inkpen, D., Uritsky, S., and Matwin, S. (2010), "*Offensive Language Detection Using Multi-level Classification*". Canadian Conference on AI.
5. Kansara, Krishna B. and N. Shekokar. "*A Framework for Cyberbullying Detection in Social Network*." (2015).
6. Maxime Rivet and Mael Tran, "*Toxic comments classification*", Stanford University journal Year [2016].
7. Spiros V. Georgakopoulos et al. "*Convolutional Neural Networks for Toxic Comment Classification*", Cornell University arXiv:1802.09957 Year 2018.
8. Y. Chen and S. Zhu, "*Detecting Offensive Language in Social Media to Protect Adolescents*," [Online]. Available: http://www.cse.psu.edu/ sxz16/papers/S ocialCom2012.pdf
9. M. Duggan, "*Online harassment 2017*," Pew Res., pp. 1–85, 2017, doi: 202.419.4372

AUTHORS PROFILE

Abhishek Aggarwal, is currently pursuing his Bachelor of Technology in Electrical Engineering from Delhi Technological University. He has worked on multiple machine learning projects on synthetic datasets using python.

Atul Tiwari, is currently pursuing his Bachelor of Technology in Electrical Engineering from Delhi Technological University. He has worked on multiple machine learning projects and has built several websites which are currently live on the internet using the MERN stack.