CLASSIFICATION OF ONLINE TOXIC COMMENTS USING MACHINE LEARNING ALGORITHMS

Veernala Sireesha[1]., Devvela Sai Priya[2].,Ettamaina Sravani[3].,Mutha Shirisha[4].,Patel Indhu[5]

1 Assistant Professor, Department Of CSE., Malla Reddy College Of Engineering For Women., Maisammaguda.,

Medchal.,Ts, India (✉veernalasireesha@gmail.com)

2, 3, 4, 5 B.Tech CSE, (19RG1A05D4, 19RG1A05D9, 19RG1A05F8, 19RG1A05G4),

Malla Reddy College Of Engineering For Women., Maisammaguda., Medchal., Ts, India

Abstract

The current pandemic has led to a meteoric rise in internet usage over the past four months, enabling a huge number of active new and old customers to use the web for a wide variety of services, including those in the fields of education, entertainment, industry, monitoring, and the emergence of a new trend in the corporate world: working from home. The dramatic growth in internet users has also led to an increase in the population of pranksters. Now, it's the responsibility of every online platform to foster positive, welcoming dialogue. Twitter, a web-based media platform where individuals may voice their opinions, serves as the ideal example. Many internet service providers find it difficult to police platforms like this one because of the ease with which hate speech, insults, threats, and libellous activities may propagate on them. Thus, the field of Toxic comment categorization is currently active with research. Here, we offer a model that dominates all others in head-to-head comparisons using the dataset and a collection of non-identical machine learning and other simple methodologies. Due to its popularity and importance as a resource for researchers, the Kaggle dataset has inspired us to take on the problem of toxic comment classification. The findings would aid in the development of an online interface that would allow us to determine the relative amount of toxicity in a specific phrase or sentence.

Keywords:

Classification, Defamatory, Multinomial Naive Bayes, Support Vector Machine, Toxic Comment, and Binary Relevance.

Introduction

People are more likely to voice their opinions and provide constructive criticism in online discussion groups now that the internet has permeated every aspect of society. Most of the time, comments like this assist the artist extemporize the content they're providing, but sometimes they may be abusive and spark hostility. Therefore, as these are accessible to the public, which includes people of varying ages, ethnicities, and socioeconomic backgrounds, it is the primary responsibility of the content-creator (the host) to remove such comments to prevent the further spread of negativity or hatred. Many governments around the world have noticed an increase in cases related to cyber bullying, which has contributed to the spread of hatred and violence, as a result of the prevalence of negative and threatening content on online platforms, especially social media. Every one of us has become a content creator, creating and distributing our own material, thanks to the democratization of content production that followed the launch of web-based media platforms, creating a framework where the nature of dispersed content cannot, at this moment, be regulated. The innovation upheaval of the last twenty years has had far-reaching effects on institutions, political systems, families, communities, and people today [1]. All researchers and scientists have struggled mightily with the problem of how to identify toxic

comments. Many people are curious about this topic not just because of the prevalence of hate speech on the internet, but also because they fear for their safety if they take part in online forums. This has far-reaching consequences for the ability of all content creators and providers to offer a safe space for public discourse. There have been certain advances in this field, such as a few models provided through API. However, these models continue to have shortcomings and cannot provide a reliable answer. In this study, we have extensively examined a family of models used for this purpose. Numerous disciplines, including economics, medicine, and ecology, have made extensive use of these models and techniques. In this work, we have taken a three-pronged strategy. We started by comparing the algorithms' performance (by adjusting the pre-processing parameters to get more desirable outcomes). Second, we've contrasted the two side by side to highlight their unique qualities. We have organized the results into groups and highlighted the relative expected values by placing them in a sequential order.

Connected Tasks

After delving into the many pieces of writing, it became clear that the early works had been the

subject of several research. In 2018, for instance, Revathi Sharma and Meet Kumar Patel used complex Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to perform the task of classifying toxic comments. They did so by employing word embedding techniques and comparing their results with those of simpler neural network algorithms. The investigation of the results of using Long-Short Term Memory (LSTM) instead of Convolutional Neural Networks (CNNs) with word-level embeddings [4] shows that the LSTM performs better than the CNNs in terms of accuracy and time execution given the same number of epochs. Logistic regression and RNN, LSTM among these 2 LSTM layers and 4 conv layers, has gotten a score of 0.9645 demonstrates greatest accuracy [7] if we look at Mujtahid A. Asif's work from 2018. Taking into account all of this work, these articles categorize the harmful remarks using Neural Network methods. In this study, we used the Binary relevance approach with the Multinomial Naive Bayes and the Support vector classifier to categorize harmful comments. We used BR Methods to try to categorize remarks that were harmful, offensive, insulting, very poisonous, based on the target's identity, or included a threat.

Arrangement Advised

The proposed Toxic comment classification system relies on a number of algorithms and stages, including logistic regression, the BR method with multinomial naive bayes classifiers, and the BR method with support vector machine (SVM) classifiers, all of which are discussed in this section.

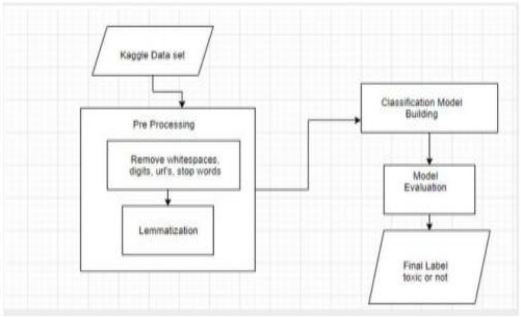

Figure 1: Proposed system.

Since the goal was to determine which of the six categories the data belonged to, the first step in approaching the problem was to establish an audible distinction between multi-label and multi-class classification. We assume that our data fits neatly into exactly one of the available labels when doing multi-class categorization. Let's imagine that you have a picture of a vegetable, and you know it could only be a potato, a cabbage, or an onion. Data in multi-label categorization, on the other hand, may share characteristics with more than one label at the same time. In this project, for instance, a comment may belong to more than one category at once; for instance, it may be toxic, hateful, obscene, and abusive, and it may also concomitantly belong to the non-toxic category, showing no affinity to any of the six labels which are used for classification. At that time, we dealt with the total number of comments across all categories (as shown by a clear visual). The most common kind of remark was poisonous, followed by vulgar, insulting, very toxic, identity-hating, and threatening comments. The amount of text in the comments is substantial, so we used a few different visualization techniques to break it down. First, we determined how many comments of varying degrees of poisonousness fell into each category.

	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	0	0	0	0	0	0
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	1	1	1	0	1	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0

Figure 2: Count of different toxicity of comments in each of the bins.

This analysisgives indepth insights about the distribution of the data in the database. The succeeding step was to perform pre-processing of the data, as the volume of the data was good as per our requirements and end goals for this project, we have discussed further on this about the techniques and procedures that we have adopted in order to curate the data and use it further in the project.

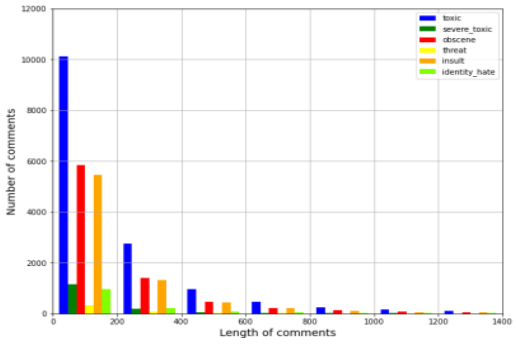

Figure3: Length of comments

Pre-processing

As a first step in pre-processing, we stripped the comments of any punctuation and other non-printing characters. At that time, it dawns on us that we have to clean up the dataset by getting rid of the worthless stop words as well. The words were also subjected to lemmatization and stemming. Finally, we used a count vectorizer and then divided the

data into training and testing sets. We've experimented with a number of different approaches to data visualization in order to extract useful insights that will improve our understanding of both the dataset and our ultimate aims. We made an effort to divide the study of the dataset into several sections based on criteria such as word length, the presence of poisonous terms, and the level of toxicity present. Our main aim was to categorize the harmful comments effectively, and we were able to narrow the field down to two algorithms thanks to the insights we gained from the visualizations.

Algorithm

Because they were developed to predict a single label, traditional algorithms struggle when presented with a set of multi-label instances. Therefore, numerous techniques were implemented using the scikit-multialarm library. A basic classifier is developed for each label and then integrated in an unconventional fashion in each approach. Multinomial Naive Bayes and SVC classifiers were utilized here. We are utilizing a binary relevance strategy to deal with multi-labels. All of the dataset's labels are broken down into individual labels, and then each label is treated as a separate classification issue in the Binary Relevance Approach. And it's shown in figures 4 and 5.

X	Y_1	Y_2	Y_3	Y_4
$x^{(1)}$	0	1	1	0
$x^{(2)}$	1	0	0	0
$x^{(3)}$	0	1	0	0
$x^{(4)}$	1	0	0	1
$x^{(5)}$	0	0	0	1

X	Y_1
$x^{(1)}$	0
$x^{(2)}$	1
$x^{(3)}$	0
$x^{(4)}$	1
$x^{(5)}$	0

X	Y_2
$x^{(1)}$	1
$x^{(2)}$	0
$x^{(3)}$	1
$x^{(4)}$	0
$x^{(5)}$	0

X	Y_3
$x^{(1)}$	1
$x^{(2)}$	0
$x^{(3)}$	0
$x^{(4)}$	0
$x^{(5)}$	0

X	Y_4
$x^{(1)}$	0
$x^{(2)}$	0
$x^{(3)}$	0
$x^{(4)}$	1
$x^{(5)}$	1

Figure 4 Figure 5

In addition, there is a little bias in the data, meaning that only a small percentage of the comments are really harmful, which means that the accuracy statistic provides misleading findings. Therefore, hamming loss and Log loss are optimal measures of performance for this algorithm.

Conclusions and Remarks

When comparing the results of the two algorithms, we find that Naive Bayes has a hamming loss of 3.6 and an accuracy of 87.6, while SVM has a hamming loss of 4.36 and an accuracy of 88.16. This provides a quick overview of the best method for ranking abusive comments.

```
Hamming_loss : 3.6503650365036506
Accuracy : 87.5937593759376
Log_loss : 1.9625004170858478
```

Figure 6: BR Method with Multinomial Naive Bayes classifiers

```
Hamming_loss : 4.367936793679368
Accuracy : 88.16381638163816
Log loss : 0.4695775266638782
```

Figure 7: BR Method with SVM classifier (from scikit-multialarm)

Conclusion

Therefore, we can conclude that the Binary Relevance method with Multinomial Naive Bayes is an effective algorithm for our purpose, with a hamming loss of 3.6 compared to the hamming loss of SVM, which was 4.36, using hamming loss as a measure of identifying the optimal algorithm to classify toxic comments.

References:

[1] NayanBanik, Md. Hasan Hafizur Rahman," Toxicity Detection on Bengali Social Media Comments using Supervised Models", (ICIET) 23-24 December, 2019

[2] Salvatore Carta, Andrea Corriga, Riccardo Mulas, Diego ReforgiatoRecupero and Roberto Saia," A Supervised Multi-class Multi-label Word Embeddings Approach for Toxic Comment Classification",2019

[3] Hind Almerekhi, Haewoon Kwak, Bernard J. Jansen, Joni Salminen," Detecting Toxicity Triggers in Online Discussions", HT '19, September 17–20, 2019, Hof, Germany, pg no: 291 – 292.

[4] Revati Sharma , Meetkumar Patel, "Toxic Comment Classification Using Neural Networks and Machine Learning", Vol. 5, Issue 9, September 2018, DOI 10.17148/IARJSET.2018.597,pg no:47- 52

[5]Mai Ibrahim, Marwan Torki and Nagwa El-Makky. (2018), "Imbalanced Toxic Comments Classification using Data Augmentation and Deep Learning ",2018

[6] Spiros V. Georgakopoulos, Sotiris K. Tasoulis, Aristidis G. Vrahatis, Vassilis P. Plagianakos," Convolutional Neural Networks for Toxic Comment Classification" ,arXiv:1802.09957v1 [cs.CL] 27 Feb 2018.

[7] Mujahed A. Saif, Alexander N. Medvedev, Maxim A. Medvedev, TodorkaAtanasova, "Classification of Online Toxic Comments Using the Logistic Regression and Neural Networks Models",2018

[8] Fahim Mohammad, "Is preprocessing of text really worth your time for toxic comment classification", Int'l Conf. Artificial Intelligence | ICAI'18,2018, pg no: 447-480.

[9] Pooja Parekh, Hetal Patel," Toxic Comment Tools: A Case Study", Volume 8, No. 5, May-June 2017, pg no: 964 – 967

[10] NavoneelChakrabarty ," A Machine Learning Approach to Comment Toxicity Classification", 2016

[11]Pallam Ravi, Hari Narayana Batta, Greeshma S, Shaik Yaseen," Toxic Comment Classification",Volume: 3, Issue: 4, 2019.