Introduction to Data Mining Project Report

Cpt\_S 315

Washington State University

Toxic Comment Classification

Created by

Ben Kaufmann

Daniel Semenko

April 2022

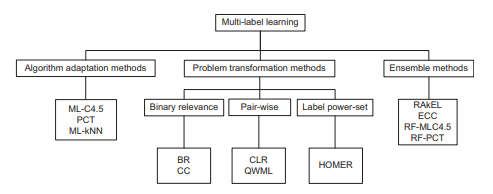
**1.0 Introduction**

DISCLAIMER FROM THE DATA SOURCE: The dataset contains text that may be considered profane, vulgar, or offensive.

**1.1 Motivations and Questions**

Discussing topics in any sort of online format can be challenging, for many different reasons. The threat of conversational toxicity is an issue that can lead people to no longer feel comfortable, out of the fear of aggravation and abuse online. This is a common problem in online video games and forums, where the ability to “hide behind your screen” allows malicious users to leave these comments, without risk of repercussion. The goal of this project is to use Machine Learning algorithms to train a model to identify toxic behavior in comments. This could be used to discourage those hostile users from posting potentially toxic or hurtful messages, allowing positive, civil discourse to occur.

The goal of this project is to implement a few different algorithms – specifically – to find the best approach to classifying toxic comments. Through the project, we will evaluate and analyze the effectiveness of the different algorithms. These different algorithms all stemmed from the type of problem presented in this project – the comments having multiple “labels”. Also known as Multi-Label Classification, it allows the authors to attempt many different methods (see **Fig. 1**), as Multi-Label Classification has many different approaches to solve it. These can include Binary Classification, Multi-Class Classification, Ensemble methods, in addition to classic algorithms being adjusted to the multi-label task – including kNN, Decision Trees, Neural Networks, and more (Madjarov, Kocev, Gjorgjevikj, & Džeroski, 2012). These were the starting point of many of our questions for this project. What is the best approach to classifying toxic behavior? Are there correlations between labels, and why do those correlations exist? How would the authors’ approaches change if just classifying as toxic vs non-toxic – single-label classification?



**Fig. 1.** Multi-label learning methods

The decision to choose this topic, and therefore this project, stems from both authors’ familiarity with toxic comments and hostile online environments. Both play video games and engage in online forums, where these types of hostile interactions are all too common. In some situations, these online platforms struggle to facilitate online conversations, leading to the removal of user comments and text chats. The authors hope that with the implementation of algorithms such as the one we explore, their usage can lead to improvements in online conversation with more productive and respectful discussion.

**1.2 The Task**

[Briefly describe challenges faced during task] The authors ran into many challenges throughout the duration of the project, the first coming on just the first day! This project takes data from the *Jigsaw Toxic Comment Classification Challenge*, hosted by Kaggle, in which the data provided is a dataset of comments from Wikipedia’s talk page edits. The very first problem was the formatting of the provided csv files, as during the download, the delimiting (occurring on commas) didn’t work as intended. [Therefore, we had to change the format blah blah blah…] [more problems…]

The main aspect of the project revolved around testing different multi-label classification methods against each other. To get there, there were a few steps before. First, the authors determined the definition of the problem, and how they would go about solving and evaluating them. The problem being the challenge found [here](https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge) on Kaggle – build a model capable of detecting different types of toxicity, which predicts the probability of each type of toxicity for each comment, trained using a highly-imbalanced training data set. We could then evaluate the performances of the different methods by comparing the accuracies of each, and the CPU computation time. The next step was to import and clean up the data – the preprocessing step. Here, the authors make the data as nice and clean as possible to make the next steps easier. At this point, the data is prepared for the different multi-label classification methods. Once those methods have been successfully implemented and run, the authors can evaluate the effectiveness of the algorithms, by comparing their accuracies and compute times.

**1.3 Brief Result Overview**

[Briefly summarize results]

**2.0 Data Mining Task**

**2.1 Task Details**

**2.2 Authors’ Questions**

**2.3 Key Challenges**

**3.0 Technical Approach**

**3.1 Algorithm**

[include pseudo-code or figure to illustrate approach]

**3.2 Addressing Challenges**

**4.0 Evaluation Methodology**

**4.1 Input Data**

**4.2 Evaluation Metrics**

**5.0 Results and Discussion**

**5.1 Results**

**5.2 Discussion**

**6.0 Lessons Learned**

**7.0 Acknowledgements**

[bibliography goes here]

Appendix 1 Python Code Modules

[python code goes here]