Toxic Comment Classifier:
Identifying Toxic Comments on Online Platforms using Text Classification Models

Irsa Ashraf Yifu Hou Ken Kliesner

University of Chicago – Masters of Science in Computational Analysis & Public Policy

Introduction

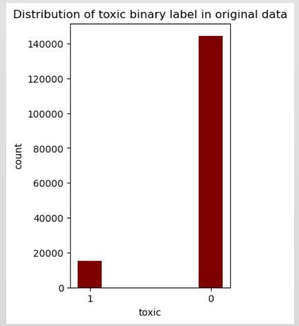
The online environment has become increasingly hostile and abusive in recent years, with hate speech and other forms of abusive behavior being prevalent on social media platforms.

This project aims to build a classifier to identify toxic comments on social media, specifically, Wikipedia webpage. A pre-labeled dataset of toxic and non-toxic online comments is be used to train the model, enabling it to discern various levels of toxicity on media platforms.

By applying multiple NLP approaches including Bag-of-Words, Convolution Neural Network and Transformers, the project strives to improve the performance of binary classification tasks on toxic/non-toxic comments. It also seeks to identify diverse toxicity categories such as *identity-based hate*, *threats*, and *insults*. Ultimately, the project endeavors to develop a versatile and scalable model capable of classifying various forms of toxic comments across multiple platforms.

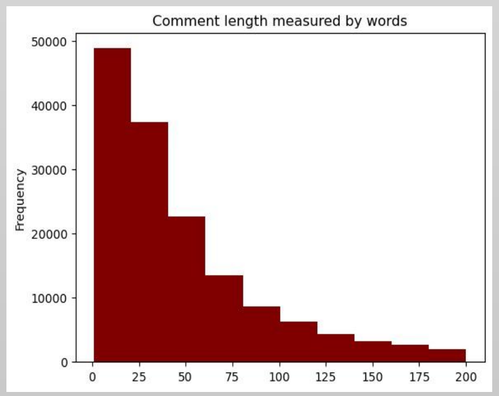
Dataset

The dataset we used for this project is the *Jigsaw Toxic Comment Classification Challenge* dataset on Kaggle, which consists of a large number of comments from *Wikipedia talk page edits*, along with binary labels. The dataset contains around **310,000** comments, and each comment is labeled on 6 different types of toxicity: **toxic**, **severe toxic**, **obscene**, **threat**, **insult**, and **identity hate**. One comment can be categorized as more than one type of toxicity.



For the data preprocessing, this project:

- performed resampling to achieve balanced data
- tokenized natural language with package `word_tokenize`
- used GloVe pre-trained word embedding to vectorize
- (for some models) standardized comment length with padding and truncating



Models

BASELINE MODEL:

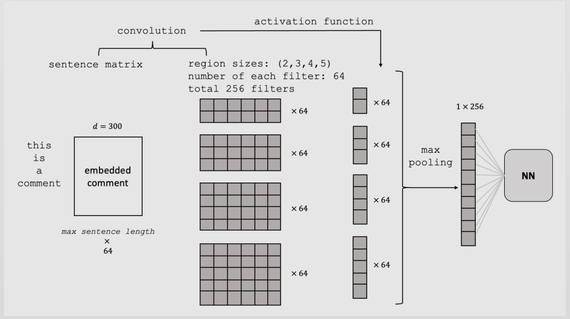
A simple BOW model is applied as the baseline model for this project. The Single-Layer BOW approach encountered issues with *NaN* losses when no words were identified. Alternatively, a BOW model with pre-trained GloVe word embeddings was used. The model achieved around 90% accuracy on imbalanced training data.

CNN BINARY-LABEL CLASSIFICATION MODEL:

To improve performance, a Convolutional Neural Network (CNN) was implemented for binary label classification of toxic and non-toxic comments. The initial version of Binary-label CNN model has following attributes:

- GloVe word embeddings of size `300`
- 64 CNN filters of each size `(2, 3, 4, 5)`
- ReLU activation, `0.5` Dropout rate, and Max Pooling

The changes in later versions include increasing the number of filters, modifying the Dropout rate, and increase number of epochs in training.



CNN MULTI-LABEL CLASSIFICATION MODEL:

For the CNN multi-label classification model, the focus was on accurately predicting the reasons for comment toxicity using additional columns such as 'severe_toxic', 'obscene', 'threat', 'insult', and 'identity_hate'. The 'severe_toxic' column was dropped due to the vague distinction from 'toxic'. Although 'identity_hate' and 'threat' were imbalanced classes, the training dataset was reduced to only toxic comments, resulting in 15,294 instances. The model's simplicity may have limited its ability to learn patterns effectively, potentially impacting its generalization on unseen data.

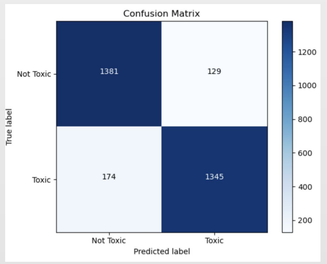
TRANSFORMER MODEL:

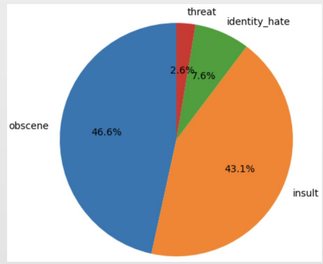
We also used a basic Transformer model, adapted from the Hugging Face library, for sequence classification tasks. The methodology we used covers fine-tuning a pre-trained Transformer model using the Transformers library. It also includes data pre-processing, model configuration, tokenization, model training, and evaluation. We used popular Transformer architectures like BERT and RoBERTa to achieve results in sequence classification tasks.

Results

CNN MULTI-LABEL CLASSIFICATION MODEL

The CNN multi-label model achieved satisfying performance. Because the unbalanced training data, the model achieved high accuracy prediction toxic label, but have issues identifying other minor label. To complement this, multiple binary-label CNN models was trained to predict each specific label.





CNN BINARY-LABEL CLASSIFICATION MODEL:

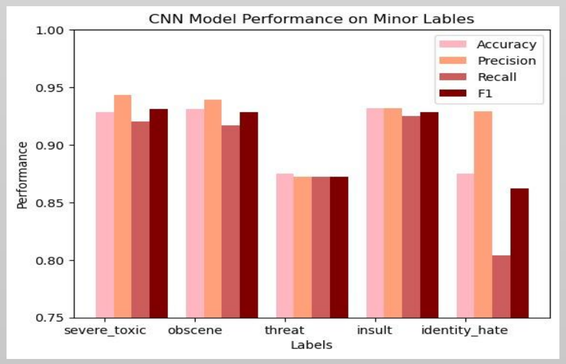
The CNN binary-label model achieved good performance on most of the labels.

For major label `toxic`, the best performance was seen from `CNN Model V1.1`. In comparison with the initial model, it increased number of filters to `128` and reduced Dropout rate to `0.2`

V1.1	Accuracy	Precision	Recall	F1
`toxic`	0.921	0.919	0.926	0.922

For minor labels, the performance of the original model `CNN Model V0` is below:

V0	Accuracy	Precision	Recall	F1
`severe_toxic`	0.928	0.943	0.920	0.93
`obscene`	0.930	0.939	0.917	0.928
`threats`	0.875	0.872	0.872	0.872
`insults`	0.932	0.932	0.925	0.928
`identity_hate`	0.875	0.929	0.804	0.862



Findings

PATTERNS IN FAILED PREDICTIONS:

The binary-label CNN model performed well on identifying toxic comments. Below are some examples of failed predictions:

False Negatives

- **misspell or uncommon toxic language**
  - ID: e22a2557c33d5df3
  - *"tno thanks mate p i s s offe"*
  - ID: f16ec7cafd4ff73c
  - *"you obviously know shit-nothing about physics, if the buildings were ..."*
- **mitigated or oddly-worded insults**
  - ID: 06a44c69b4c3fb43
  - *"In response to your recent comment on my talk page. I suggest you contract cancer."*
- **ambiguous connotations for language models**
  - ID: e8d66a843390f637
  - *"Do it and I will cut you"*

False Positives

- **triggering words used in non-toxic context**
  - ID: *289b9ebd8ee46b91*
  - *"This article is useless without pics"*

Ambiguous Labels (up for debate of interpretation):

- It is worth mentioning that the labeling of original dataset is not perfect - there are a few comments that might be incorrectly labelled.
  - ID: *583c3800a5b3b464*
  - *"Do what you want, but you'll never get rid of me, that's a promise. Give your sister a kiss for me."*
    - Labeled as `not toxic`, we identified as `toxic`
  - ID: *d4090f8db8939d73*
  - *"Yo who the heeck wrote this and how the heck do they even know what happends."*
    - Labeled as `toxic`, we identified as `not toxic`

Future Work

- This project should look into additional language models, including more complicated Transformer models that can have more specified hyperparameter optimization, to improve prediction performance.
- Most of the training process of this project was done on CPU. By using scalable solutions such as GPU and AWS, the project is expected to significantly reduce runtime for each model.
- The project hopes to expand dataset to other sources, such as Instagram comments, Twitter comments, etc.