1. Toxic Content Filtering

Author: Tammy Metz
CSPB 4830: Natural Language Processing
May 5, 2025

2. Abstract

This paper explores the implementation of machine learning-backed identification of toxic comments to enable content moderation at scale. Using the NLTK English stopwords corpus, GloVe (Global Vectors) pre-trained word embeddings, and Tensorflow with keras we implemented an LSTM model.

3. Introduction

Some people view the internet as their own personal playground where they are not actually interacting with other human beings. Because of this, many comments are toxic in some way (racist, sexist, foul language, etc.). Companies may have Trust & Safety teams to help moderate content on the platform, but their efforts are needed for the most serious violations of terms of service such as Child Sexual Abuse Material and terrorist activity.

As platforms scale their user bases, they cannot realistically scale their human capital enough to keep up with all of the content. When they are able to use machine learning or other automated methods to identify objectionable content, they can free their manual review team for the most egregious cases.

This project will classify comments to identify and filter toxic comments, a tool which can then be implemented to reduce the workload of manual reviewers.

4. Related Work

Toxic comment classification has been researched extensively, with simpler solutions such as dictionary-based text matching and Naive Bayes being replaced with various neural networks that include a word embedding layer.

Wang, et al. compared the Naive Bayes approach with Convolutional Neural Networks, and Long Short Term Memory networks. Unsurprisingly, the CNN and LSTM each had higher accuracy than Naive Bayes. They also found that using GloVe for word embeddings resulted in slightly higher accuracy than their other choice, FastText [1].

Schmidy and Wiegand demonstrated toxic comment classification using Support Vector Networks. They compiled their own dataset from a variety of sources, as according to them they did not have access to a standard corpus [2]. I believe their research took place shortly before the Jigsaw comment toxicity dataset was published in 2017.

Duchêne, et al. compared several transformer-based models and found all performed strongly [3]. It seems likely that one of these would yield the best results. However, for the purposes of this project we will be using an LSTM.

For any of the above modern approaches, we need a method for creating word embeddings. One popular framework for learning word embeddings is *word2vec*, which includes two model architectures: Skip-Gram and Continuous Bag of Words[4].

Global Vectors for Word Representation (*GloVe*) is a next generation model that uses the global context to build a word co-occurance matrix. *GloVe* outperforms both *word2vec*'s CBOW and skip-gram with a corpus less than half the size [5].

Our work will follow the *GloVe* embeddings with LSTM approach that Wang, Yang, and Wu [1] took, but using more than one hidden dense layer.

5. Data

The source of the data used was the Jigsaw comment toxicity dataset. This dataset consists of:

- Training data containing 159,571 rows of labeled comments ('comment_text') with the following boolean features (using 0 or 1): ['toxic', 'severe_toxic', 'obscene', 'threat', 'insult', 'identity_hate']
- Test data containing 153,164 rows of comments in one file and labels in a separate file, of which 63978 were correctly labeled and kept

The only preprocessing done was to remove comments from the testing set if any toxicity label was -1, as those were inserted as part of the Kaggle competition and not viable data.

6. Methodology

After preprocessing the data, we removed stopwords by using the NLTK English stopwords corpus. We also tokenized words with the keras Tokenizer, and padded shorter comments so all comments would be of length 30 characters.

The next step was to use Tensorflow keras to create a model with a neural network containing an embedding layer, 3 dense hidden layers, and an output layer using a sigmoid function. The embedding layer used a pretrained GloVe embedding.

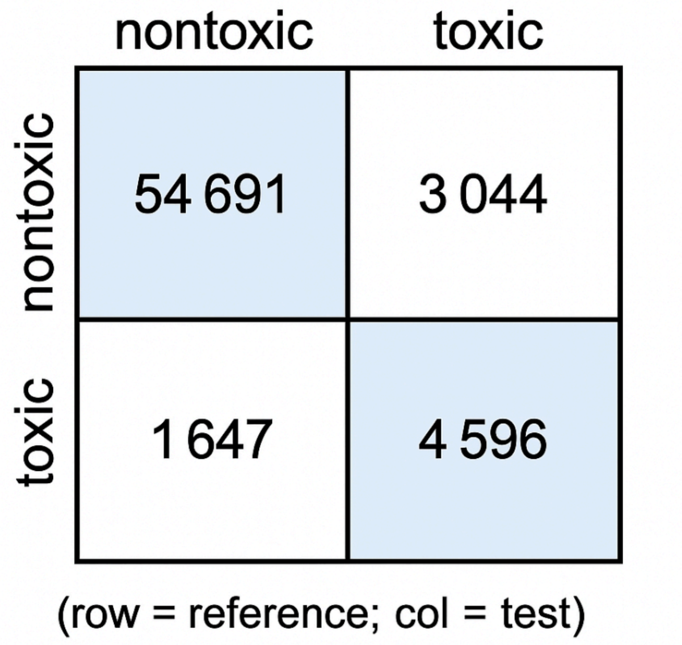
We trained the model on training data provided by the Jigsaw comment toxicity dataset, and ran the model on its test data to evaluate the model performance.

7. Results

The overall accuracy of the test data was 0.9267. This was encouraging, but not nearly as good as Wang, Yang, and Wu's 0.996 with a similar setup. The model had some success in classifying non-toxic comments. With precision of 0.9708, recall of 0.9473, and an F-measure of 0.9589, the results were reasonably correct.

However, toxic comments fared much worse. They had a precision of 0.6016, recall of 0.7362, and an F-measure of 0.6621.

A confusion matrix demonstrates 54,691 true negatives (nontoxic comments) and 4,596 true positives (toxic comment identified). However, the amount of false positives (comments identified as toxic that were actually nontoxic) is quite high at 3,044.



	nontoxic	toxic
nontoxic	54 691	3 044
toxic	1 647	4 596

(row = reference; col = test)

8. Discussion

Using a short max_sequence_length of 30 characters sped up the model training time considerably. Additionally, after experimenting with various batch sizes, a batch size of 128 worked well.

One area the confusion matrix might suggest examining is the imbalance of nontoxic to toxic comments in the test set. It's possible that a more balanced test set would yield better precision and recall for the identification of toxic comments.

9. Conclusion and Future Work

The LSTM with GloVe embeddings could be improved upon. Future work could include trying a larger GloVe embedding, experimenting with pre-trained models instead of training our own, trying other embeddings, or using a transformer.

10. Bibliography

[1] K. Wang, J. Yang, and H. Wu, "A survey of toxic comment classification methods," *arXiv preprint arXiv:2112.06412*, Dec. 2021. [Online]. Available: https://arxiv.org/abs/2112.06412.

[2] A. Schmidt and M. Wiegand, "A survey on hate speech detection using natural language processing," in *Proc. 5th Int. Workshop Natural Lang. Process. Social Media*, Valencia, Spain, 2017, pp. 1–10.

[3] C. Duchêne, H. Jamet, P. Guillaume, and R. Dehak, "A benchmark for toxic comment classification on Civil Comments dataset," *EGC 2023*, vol. RNTI-E-39, pp. 19–30, Jan. 2023. [Online]. Available: https://arxiv.org/abs/2301.11125.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," presented at the International Conference on Learning Representations (ICLR), Scottsdale, Arizona, USA, May 2-4, 2013. [Online]. Available: https://arxiv.org/abs/1301.3781.

[5] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. 2014 Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1532–1543. [Online]. Available: https://aclanthology.org/D14-1162.pdf.

[6] N. Van Otten, "How to Use GloVe Embeddings in TensorFlow," *Spot Intelligence*, Nov. 27, 2023. [Online]. Available: https://spotintelligence.com/2023/11/27/glove-embedding/#How_to_use_GloVe_Embeddings_in_TensorFlow