CIS530: Milestone 1 Report

Matthew Kligerman, Sara Nayak, Aishwarya Ramanath, Laura Tutelman

1 Literature Review

1.1 Introduction

In an era where social media is one of the most prevalent forms of communication, toxic comment analysis is essential to maintaining a positive environment. Due to the massive amount of online discussion that occurs every day, manual moderation of toxic comments is impossible. Toxic comment analysis, prediction, and categorization will be key to building tools that can help to address these issues, from applications like general content moderation to banning toxic users. In order to develop a good model for toxic comment classification, four different related works were used to influence the data preparation and machine learning model that we plan to use.

1.2 Discussion of Relevant Works

Preparing data is one of the earliest steps in any investigation of data and is of vital importance to be able to make accurate predictions. [3](Zhao et Al.) studies the comparison in performance of sentiment analysis of tweets given different preprocessing techniques.

The methodologies for preprocessing explored are replacing negation (e.g, 'haven't' to 'have not'), removing stop words, expanding acronyms (e.g, '2 moro' to 'tomorrow'), removing numbers, and removing URLs and reverting repeated letters (e.g, 'coooool' to 'cool'). Preprocessing is likely to remove noise in data like tweets, but each method performs differently with the models used (SVM, Random Forests, Regression and Naive Bayes).

For the STS-Test, STS-Gold, SS-Twitter, SE-Twitter and SemEval2014 datasets, the experimental results were able to find a pattern in terms of improvement in accuracy when the data was preprocessed in the ways mentioned above. It was evident that the URLs and stop words contained no useful information for sentiment classification, so removing them did not affect the results. Both these methods reduce vocabulary size and can be used as an effective method of preprocessing. Removing numbers didn't improve performance accuracy except with the SVM model, which performed better when numbers were removed from the tweets.

For most models and dataset combinations, removing negation improved accuracy, while both removing letters and random deletion of words were not recommended as preprocessing methods. Removing repeated letters seemed to perform differently on each dataset, possibly influencing the semantic features of words, while deletion of a word that could possibly be a key word may damage the semantic sentence relationship. The recommended preprocessing methods include handling negation, removing stop words, URLs and numbers.

Beyond the preprocessing step, data and methodology must also be considered. One literature review of 31 papers [1](Darko et Al.) on toxic comment analysis found that the most used data set was Jigsaw's Toxic Comment Classification Challenge on Kaggle (used in 22/31 papers) where comments are split into toxic, severe toxic, obscene, threat, insult, and identity hate categories. Other datasets were study-specific and looked at data ranging from Twitter to Instagram to video game data. The primary machine learning models used were Convolutional neural networks (CNNs) (38% of papers), logisitc regression classifiers (29%), bidirectional long short-term memory (Bi-LSTM) (26%), and Bidirectional Gated recurrent unit networks (Bidirectional GRUs) (19%), LSTM (16%), Support vector machines (SVM) (16%), Bidirectional Encoder Representations from Transformers (BERT) (13%). Note some papers evaluated more than one learning model. And the primary method of evaluation was either F1 Score or Accuracy.

The literature review also notes that in recent studies of toxic comment analysis, transformers have shown to be superior in performance (BERT, DLM, DLNet), and should be used for further development of toxic comment classifiers.

These transformers provide better identification of online toxic speech. One paper[2](d'Sa et Al.) showed this using both binary and multi-class classification on a Twitter corpus, proving that between a pre-trained BERT (Bidirectional Encoder Representations from Transformers) model and a DNN classifier, the BERT model had better performance.

The proposed methodology is to consider both binary and multi-class classification by using fastText and BERT embeddings as inputs to DNN classifiers (CNN and Bi-LSTM). Word embeddings aim to project words in some continuous vector space, where semantically or syntactically related words should be located near each other. Due to the computationally demanding nature of training embeddings, it is advantageous to use pre-trained word embeddings (Google's BERT or Facebook's fastText mdoel). fastText embeddings are based on a skip-gram model where each word is a bag of character n-grams (good for misspelled words and rare/missing words). BERT is based on a methodology of transforemres, using an attention mechanism which determines how to look at the relationship of words in a sentence i.e. contextualization.

Using a dataset of over 24,000 Tweets with annotator agreement of 92%, a model with fine-tuning of a pre-trained BERT model (F-Score: 97%) outperformed a feature-based approach (Max F-SCore: 92%).

1.3 Conclusion

[4](Rinal et Al.) compares the performance of RNNs and CNNs for toxic comment classification. While CNNs are predominantly used with image classification problems, RNNs have long been known to work well with text. Instead of learning features of sentences that are local, recurrent neural networks consider long term dependencies. LSTMs are able to retain important information and extract patterns better. Sentiment analysis relies heavily on context, requiring sequential data, on which LSTMs usually perform well. For the dataset we hope to work with, this paper's empirical results show that LSTMs outperformed CNNs in terms of efficiency as well as performance.

For the task of multi-label classification of toxic comments, we will be preprocessing our data using the methodology discussed earlier in the related work section. We will work with Naive Bayes and LSTMs, evaluating their performance using F-score and accuracy. We will also attempt to use BERT, since it's been proven to outperform other feature based models.

2 Data

The dataset contains Wikipedia comments that have been labeled for different types of toxicity. The types of toxic behaviour include - toxic, severe_toxic, obscene, threat, insult, and identity hate. Our objective will be to accurately classify a comment as on of these 7 categories. Below we show the formatting of the data.

id	text	toxic	severe	obscene	threat	insult	identity
1	...	1	0	0	0	0	0

3 References:

1. Andročec, Darko. "Machine Learning Methods for Toxic Comment Classification: A Systematic Review." Acta Universitatis Sapientiae, Informatica, vol. 12, no. 2, 2020, pp. 205–216.

2. Ashwin Geet d'Sa, Irina Illina, Dominique Fohr. BERT and fastText Embeddings for Automatic Detection of Toxic Speech. SIIE 2020 - Information Systems and Economic Intelligence; International Multi-Conference on:"Organization of Knowledge and Advanced Technologies"(OCTA), Feb 2020, Tunis, Tunisia. ffhal-02448197v2f

3. Jianqiang, Zhao, and Gui Xiaolin. "Comparison Research on Text Pre-Processing Methods on Twitter Sentiment Analysis." IEEE Access, vol. 5, 2017, pp. 2870–2879., https://doi.org/10.1109/access.2017.2672677.

4. Patel, Rinal, and Hetal Gaudani. "Toxic Comments Classification Using Neural Network." International Journal of Innovative Technology and Exploring Engineering, https://doi.org/10.35940/ijitee.