Classification of Offensive Language on Social Media

Anmol Gupta 2018329

Naman Jain 2018347

Navneet Agarwal 2018348

Sudhir Attri 2018267

1 Introduction

With the exponential rise of the internet and social media, and the anonymity it provides, toxic online content has become a major issue. Toxic content is usually divided into two categories – offensive language and hate speech, with the latter being a serious concern for organisations. Traditionally, a human moderator would review the reports, following the policies the site has put in place against such content. However, every minute hundreds of thousands of comments are made on a single platform, which makes this method expensive and inefficient. The need arises to have a system which can flag toxic content automatically. In our project, we aim to harness the power of natural language processing and machine learning techniques to create such an automated system.

2 Literature Review

Many studies in this domain have worked on the the problem of toxic speech detection as a classification task, by either performing binary classification to determine whether a comment is toxic, or performing multi-class classification to identify the type of toxic comment. Study [1] uses different models to flag toxic content and distinguish between offensive language and hate speech. On a data set of tweets, it uses n-grams for feature extraction which are then weighted using TF-IDF. It compares the performance of Naive Bayes, Logistic Regression (LR) and Support Vector Machines (SVMs). Comparative analysis of different models was done and it is observed that using trigrams and L2 normalisation of TF-IDF weights in LR achieves the best performance. Study [2] also uses a LR model with similar hyperparameters, but includes more features like binary and count indicators for tweet data such as hashtags, mentions, retweets, and URLs, number of characters, words, and syllables in each tweet. Both studies use bag-of-words approach which has a high recall but low precision, i.e. it performs well for offensive content detection, but often misclassifies it as hate speech. It fails to account for the sentence structures, but this can be improved by adding linguistic features.

Study [3] implemented a hierarchical structure (classification at three different levels) to identify the type and target of offensive tweets. Based on this approach OLID (Offensive Language Identification Dataset) was created and three models were tested on it: Linear SVM model trained on unigrams, Bidirectional Long Short-Term-Memory (BiLSTM) model with three layers (an input embedding layer, a bidirectional LSTM layer and an average pooling layer of input features) and a Convolutional Neural Network (CNN) based on same multi-channel inputs as the BiLSTM model. For the first two levels, CNN had the best macro-F1 scores and for the third level both BiLSTM and CNN had equal macro-F1 scores.

Study [5] classified comments from Yahoo Finance and News data set, and tried to improve the methodology for feature extraction. It tested more sophisticated algorithms to learn representations of comments as low-dimensional vectors in contrast to the previous studies that used n-grams with edit distance, manual regex patterns, word sense disambiguation techniques and paragraph2vec. Their methodology consists of computing four classes i.e. n-grams, linguistic, syntactic and distributional semantics of features. They trained models with combinations of these features on Vowpal Wabbit's regression model and obtained best results using word2vec and comment2vec with a high F_1-score.

CNN based approach was used in [6] where the feature embedding was generated using word embedding from word2vec and random vectors techniques as well as character n-grams. The model consists of 2 CNN layers, 2 pooling layers and an output softmax layer. With random vector word model as the baseline, word2vec performed the best among others except the recall from LR with character n-grams model. The paper suggests the Long Short-Term Memory (LSTM) to utilise the sequential nature of comments. Such a LSTM based model was proposed by [4]–the features for input layers were computed using word-based frequency vectorisation. The network consisted of sigmoid activation LSTM layer followed by a dense layer using ReLU activation. The final output is obtained by ensemble of multiple LSTM classifiers based on voting rule. The ensemble classifier outperformed all single classifiers, and the F_1-scores increased with a user's tendency to write hate-speech.

Study [7], unlike traditional toxic language detection, takes up the task of preemptive detection, i.e. predicting whether a particular conversation thread, in future, will incite a toxic comment using thread-level feature extraction. Data set of Wikipedia conversation threads with labelled comments is used in [7]. Neural networks based encoders (implemented using BiLSTM models) were used to get vector representations of each comment in the data set. Two models were constructed: i) considering single comments only, ii) taking into account the context from preceding comments (thread-approach). It was found that the context-sensitive models did not significantly outperform context-agnostic ones.

3 Concluding Remarks

Considering the legal and moral issues associated with hate speech, it becomes important to accurately differentiate both. [2] observes that hate speech can be targeted at a person or a group of people. Future work, such as [3], took into consideration the different contexts. It is also pointed out how homophobic and racist tweets were tagged as hate speech more than often then sexist ones in the data set. This can lead to social bias and prejudice creeping into the models and one should be able to correct for it. [7] left us with future prospects of enriching input features with information (such as the previous comment-behaviour of the user, better encoding etc.) to achieve better preemptive detection. In general, results obtained from CNN and LSTM gave promising results and performed better than simpler classifiers.

4 References

[1] A. Gaydhani, V. Doma, S. Kendre, and L. Bhagwat, ‘Detecting Hate Speech and Offensive Language on Twitter using Machine Learning: An N-gram and TFIDF based Approach’, arXiv:1809.08651 [cs], Sep. 2018, Accessed: Oct. 31, 2020. [Online]. Available: http://arxiv.org/abs/1809.08651.

[2] T. Davidson, D. Warmsley, M. Macy, and I. Weber, ‘Automated Hate Speech Detection and the Problem of Offensive Language’, arXiv:1703.04009 [cs], Mar. 2017, Accessed: Oct. 31, 2020. [Online]. Available: http://arxiv.org/abs/1703.04009.

[3] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, ‘Predicting the Type and Target of Offensive Posts in Social Media’, in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, Minnesota, Jun. 2019, pp. 1415–1420, doi:10.18653/v1/N19-1144 .

[4] G. K. Pitsilis, H. Ramampiaro, and H. Langseth, ‘Effective hate-speech detection in Twitter data using recurrent neural networks’, Appl Intell, vol. 48, no. 12, pp. 4730–4742, Dec. 2018, doi: 10.1007/s10489-018-1242-y.

[5] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, ‘Abusive Language Detection in Online User Content’, in Proceedings of the 25th International Conference on World Wide Web - WWW ’16, Montreal, Quebec, Canada, 2016, pp. 145–153, doi: 10.1145/2872427.2883062.

[6] B. Gambäck and U. K. Sikdar, ‘Using Convolutional Neural Networks to Classify Hate-Speech’, in Proceedings of the First Workshop on Abusive Language Online, Vancouver, BC, Canada, Aug. 2017, pp. 85–90, doi: 10.18653/v1/W17-3013.

[7] M. Karan and J. Šnajder, ‘Preemptive Toxic Language Detection in Wikipedia Comments Using Thread-Level Context’, in Proceedings of the Third Workshop on Abusive Language Online, Florence, Italy, Aug. 2019, pp. 129–134, doi: 10.18653/v1/W19-3514.