11830 Report: Civility in Communication

Jiyang Tang
jiyangta@andrew.cmu.edu

1 Introduction

Toxic speech detection using machine learning has been a hot research topic in recent years. In this report, we investigate the methods of performing this task, using data sampled from the SemEval2019 challenge (Zampieri et al., 2019) and TwitterAAE dataset (Blodgett et al., 2016).

2 Method

We build three types of toxic speech classifiers and test their classification performance and the biases on out-of-domain non-toxic data.

2.1 PerspectiveAPI-based Classifier

We first use PerspectiveAPI to build a rule-based classifier. Perspective score represents the toxicity level of a piece of text. The classifier recognizes a sentence as offensive if its Perspective score is larger than 0.8.

2.2 Linear Classifier using Word Count Vectors

The second baseline model is a linear classifier using word count feature vectors. The model can recognize offensive words in text but may fail in other situations.

2.3 RoBERTa Sentence Classifier

The third classifier is built on top of the RoBERTa (Liu et al., 2019). The model is appended with a linear layer that transforms the first token prediction to a binary label prediction, with 1 indicating toxic speech and 0 meaning non-toxic speech. There are several advantages of using RoBERTa. Its text tokenizer uses binary pair encoding (BPE) (Sennrich et al., 2015), which means the model utilizes Unicode characters such as emojis to perform classification. As a language model, RoBERTa recognizes contextual text information which should in theory improve the classification performance on hard cases. The model is also pretrained on a large amount of data, so we only need to finetune it for a small number of iterations.

3 Experiments

For all three classifiers, we train them using the train set of SemEval2019 and report their F1 score, precision, recall, and accuracy on the development set. We also present their false-positive rate (FPR) on TwitterAAE data for each demographic group. This FPR can be an indicator of the level of bias learned from the data.

3.1 Linear Classifier

For the linear classifier, we clean the text before extracting word count features. We use `spaCy` (Honnibal et al., 2020) `en_core_web_sm` model to tokenize raw strings into lists of lower-case words and remove all punctuations. Then we use `Ekphrasis` (Baziotis et al., 2017) library to normalize the text. `Ekphrasis` specializes in processing social media text which contains typos, URLs, emojis, and so on. We use it to normalize such components, unpack hashtags, fix elongated words, and convert emoticons to text. Finally, we use `sklearn` (Pedregosa et al., 2011) `CountVectorizer` to convert text into word count vectors. Additionally, both L1 and L2 regularization are used to avoid overfitting the training data.

3.2 RoBERTa Sentence Classifier

For the RoBERTa classifier, we feed the text directly to the tokenizer and rely on the tokenizer and the model to handle special text components mentioned in the previous section. We use Huggingface's `transformers` library (Wolf et al., 2019) to load a pre-train RoBERTa model and finetune it for one epoch. We set the batch size to 28, learning rate to 10^{-5}, and weight decay of the

Model	Accuracy	F1	Precision	Recall	FPR AA	FPR White	Hispanic	Other
Perspective	0.76	0.67	0.82	0.66	0.20	0.07	0.10	0.01
Linear	0.76	0.70	0.74	0.69	0.22	0.10	0.12	0.01
RoBERTa	0.79	0.76	0.77	0.75	0.26	0.14	0.16	0.01

Table 1: Results of three classifiers.

Model	F1 (non-toxic)	F1 (toxic)	Precision (non-toxic)	Precision (toxic)	Recall (non-toxic)	Recall (toxic)
Perspective	0.85	0.49	0.75	0.89	0.98	0.33
Linear	0.83	0.58	0.78	0.69	0.89	0.49
RoBERTa	0.85	0.67	0.83	0.71	0.87	0.64

Table 2: Results of three classifiers.

AdamW optimizer (Loshchilov and Hutter, 2017) to 0.01.

4 Results and Discussion

Table 1 shows that the best-performing classifier is RoBERTa, as it has the highest F1 score and recall. However, the Perspective model has the highest precision. Note that the performance gap between RoBERTa and the other two classifiers is not big. With such a big increase in the number of trainable parameters but a small performance increase compared the linear classifier, it is likely that the model is learning only on a surface level. The RoBERTa model is also the most biased model as it has the highest FPR in almost all demographic groups, particularly in African Americans. Meanwhile, the Perspective model is the least biased.

Table 2 shows the F1 score, precision, and recall of the classifiers on toxic and non-toxic text separately. The Perspective model has the highest recall of non-toxic speech but the worst recall of toxic speech, as a result of being the least biased model. On the contrary, the RoBERTa model has the biggest increase in its recall of toxic content, implying the potential of BERT-based models in toxic speech detection.

A drawback of the linear classifier is shown by its low precision of toxic speech. For example, having curse words in a sentence does not necessarily express toxic intent. It could be a joke instead. The linear classifier can be easily misled in such cases.

Our results show the potential of using machine learning to combat abusive language. However, it is equally important to mitigate the biases learned by the model. As shown in Table 1, the FPR in African American demographic group is higher than any other group. This phenomenon has also been discovered in other studies (Zhou et al., 2021) It is not acceptable for such a model to be deployed in real products and incorrectly classify content made by a particular demographic group as harmful content. In addition, it is hard to define offensive language (Fortuna et al., 2022). As shown in the result of the linear classifier, harmful language does not equal bad words. Finally, biases in data collection and annotation should be taken into consideration when developing new debiasing methods (Clark et al., 2019).

5 Conclusion

In this report, we create three toxic speech classifiers and analyze their performance and biases. Our experiment results show that the best-performing model is also the most biased model. Therefore, debiasing techniques are needed to build a fair toxic speech detection model.

References

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, Vancouver, Canada. Association for Computational Linguistics.

Su Lin Blodgett, Lisa Green, and Brendan O'Connor. 2016. Demographic dialectal variation in social media: A case study of African-American English. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1119–1130, Austin, Texas. Association for Computational Linguistics.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4069–4082, Hong Kong, China. Association for Computational Linguistics.

Paula Fortuna, Monica Dominguez, Leo Wanner, and Zeerak Talat. 2022. Directions for NLP practices applied to online hate speech detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11794–11805, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *CoRR*, abs/1508.07909.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2021. Challenges in automated debiasing for toxic language detection. *CoRR*, abs/2102.00086.