Classification of Toxic Comments and Identification of Unintended Bias

Wenyang Pan
Northwestern University
wenyangpan2021@u.northwestern.edu

Abstract

This paper describes a mechanism to develop models for classifying toxic comments and identify potential bias created by the training process for different identity subgroups. We train three different types of classification models and explore their potential bias from evaluation, explanation and documentation perspectives. The model achieves reasonable performance and sheds light on some important issues that can be further explored.

1 Introduction

With the widespread of toxic comments on the Internet, it is important to have a mechanism to identify and potentially remove those comments. One important toolkit for us is to use various natural language processing (NLP) models to classify the toxicity of each comment. However, these models might falsely associate some frequently attacked identity (e.g. “gay”) with toxicity.

This project aims to address those issues. Specifically, it wants to achieve two goals: (1) develop NLP models to classify whether a comment is a toxic comment; (2) have a mechanism to identify and document potential bias for these models.

The rest of the report is organized as the following: section 2 describes relevant literature for this project; section 3 discusses the dataset we use; section 4 discusses the methodology for classification and bias identification; section 5 reports the results and section 6 concludes.

2 Related Work

The relevant literature for this project falls into two categories. The first part is about methods for classification. The second part is about methods to identify and report potential bias from the model.

There are huge amount of literature for text classification and it is hard to summarize them in this short report. Kowsari et al. (2019) gives a nice survey for various methods. Among various methods, we identify three methods for this project: (1) a simple bag of word models with logistic regression; (2) word embeddings with implementation from fastText (Joulin et al., 2016); (3) fine-tuning of a DistilBert model (Sanh et al., 2019). These three approaches represent a nice combination of well-established baselines and recent cutting-edge methods.

For bias identification, there are three approaches: better evaluation metrics, explanation of predictions and comprehensive documentations. For evaluation metrics, Borkan et al. (2019) proposes metrics to identify the performance of a model in different subgroups of population. For model explanations, one common approach is local interpretable model-agnostic explanations (lime). (Ribeiro et al., 2016) These approaches can inform us whether the model uses some words that associate with certain identities to make predictions. For model documentations, Mitchell et al. (2019) proposes the model card method to better document various aspects of a model, including its fairness across population. The combination of these methods empowers us to better identify and document potential issues for a model.

3 Dataset

The dataset for this project comes from the Kaggle Competition - "Jigsaw Unintended Bias in Toxicity Classification". The data includes 1,804,874 online comments and their toxicity labels created by human raters. Among these comments, 1,660,540 comments (92 percent) are considered normal comments and 144,334 comments (8 percent) are labelled as toxic comments.

One sample toxic comment is like "What if his opinion is that most other commenters are idiots? :-)" and a non-toxic comment is like "This is so cool. It’s like, ’would you want your mother to read this??’ Really great idea, well done!".

Moreover, a subset of these comments are annotated with identity attributes, which represents identities that are mentioned in the comment. For example, the text "Not a good idea, considering that the elephants at the zoo are Asian elephants." will have the attribute "Asian".

We also have a separate hold-out test set with 10,000 rows to evaluate the performance of our model.

4 Method

4.1 Classification

We develop three types of classification models for this project. First, we encode the text by counting the frequency of the most frequent 5000 words after filtering out common stop words in the English and words that appear in less than 0.1 percent of the data or words that appear in more than 99 percent of the data. The reason for filtering out words that appear very infrequent or very frequent is that those words might not provide valuable information for the model to learn. Also, since the dataset is very large, filtering out words and then keeping the most frequent 5000 words can save the computation time. Then we fit a logistic regression model to this encoding. Support Vector Machine (SVM) model is not used because the evaluation metrics we use require the model outputs probabilities for each label. We will dive into more details in the next subsection.

Second, we train a classification with continuous word embeddings by using the implementation from fastText. For this model, we change all the text to lowercase and only keep a word if it occurs more than 5 times in the dataset to improve computational efficiency. For the rest of the hyper-parameters, we use the automatic hyperparameter tuning method from the fastText package to select hyper-parameter combinations that give the best f1-score in a 20 percent validation set.

Finally, we fine-tune a DistilBERT model for classifying toxic comments. Due to limited computational resources (even 10 percent of data takes more than 3 hours in a P100-GPU), we only train the model in 10 percent of the original dataset. The details of hyper-parameters can be found in the notebook in the Github repository discussed in the appendix. One particular choice we made is to use mixed precision for training the dataset as this approach allows us to use less GPU memory and potentially enables the model to generalize better in the test set. (Micikevicius et al., 2017) We will discuss the results for each model in the "Results" section. Before that, we will discuss how to identify potential bias in the model.

4.2 Bias Identification

The first approach to identify bias is to use appropriate evaluation metrics. For our task, we will use a combination of three types of AUC-ROC discussed in Borkan et al. (2019), which includes: Subgroup AUC, BPSN (Background Positive, Subgroup Negative) AUC, and BNSP (Background Negative, Subgroup Negative) AUC. Specifically, Subgroup AUC refers to the AUC calculated when we only use comments that mention a specific identity group. BPSN AUC refers to the AUC calculated when we only use non-toxic comments that mention a specific identity and toxic comments that do not. BNSP AUC refers to the AUC calculated when we only use toxic comments that mention a specific identity and non-toxic comments that do not. These three types of AUC give a good indication about whether the model makes fair predictions for different identity subgroups. For this project, we will focus on the following subgroups: male, female, homosexual, Christian, Jewish, Muslim, black, white, and psychiatric.

To be consistent with the evaluation metrics used in the Kaggle Competition and summarize the metric for each identity group together, we will use a generalized mean for various subgroups. (Jigsaw, 2019) The mean is defined as the following:

$$M_5\left(m_s\right) = \left(\frac{1}{N}\sum_{s=1}^{N} m_s^5\right)^{\frac{1}{5}}, \tag{1}$$

where m_s is the bias metric m calculated for each subgroup s, and N is the number of identity groups. This generalized mean combines the AUC for different identities subgroups and we also want to combine different types of AUC we discussed above into a single score. We will define the score as the following:

$$score = w_0 AUC_{overall} + \sum_{a=1}^{A} w_a M_5(m_{s,a}) \tag{2}$$

, where $w_0 = w_a = 0.25$ and A is the number of metrics (3), $M_5(m_{s,a})$ is the bias metric for identity subgroup s using sub-metric a. We will call this *mixed AUC score* in the rest of this paper.

In addition to appropriate metrics, we try to document the potential bias of a model by using the model card method proposed in Mitchell et al. (2019). Specifically, the model card will describe the training process, performance metrics and potential limitation and bias for a model. We provide a link to a sample model card in appendix A.3. Moreover, we want to explore how the model makes the prediction and identify potential bias from the explanation. To achieve this goal, we will use the lime method. The lime method learns "a interpretable model locally around the prediction". (Ribeiro et al., 2016) In appendix A.2, we provide a link for a deployed app of our model and each prediction from the DistilBERT model in the app comes with an explanation from lime.

5 Results

We evaluate each model we discussed in section 4.1 in a hold-out test set with 10,000 observations. For each model, we calculate the accuracy, f1-score and the mixed AUC score discussed in section 4.2. Table 1 summarizes the results of different models.

The model from fastText achieves second best performance in accuracy and f1-score yet performs slightly worse than the basic logistic regression model. The fine-tuning version of DistilBERT achieves the best performance in all three metrics. We expect the DistilBERT model to achieve an even better result if we use the whole dataset instead of 10 percent of the dataset for training.

The good performance in mixed AUC score of DistilBERT models aligns with our intuition. Since attention-based model can infer the representation of word based on its context, it can better avoid falsely associating some frequently attacked identity (e.g. “gay”) with toxicity.

However, one drawback for the DistilBERT models is that it performs badly the for Muslim identity subgroup as it has very low subgroup AUC. If we pass the sentence "Muslims are people who follow or practice Islam, an Abrahamic monotheistic religion." into the model, it will classify this comment as toxic and the lime method will show that "Mulism" is the keyword that leads to this prediction. We will discuss in next section about some further works to solve this issue.

6 Discussion

In conclusion, our models achieve reasonable results to identify toxic comments and the metrics and explanation methods enable us to better identify potential bias of a model. The model card approach helps us better document limitations of a model.

There are several directions to further explore. First, the identification of several toxic comments can even be controversial among humans. Thus, a consistent standard for identifying toxic comments and a review of potential labeling errors in the dataset might help improve the performance. Second, this project mainly focuses on identification and documentation of potential bias. An important direction for further work will be how to mitigate those biases. As we discuss in the result section, the DistilBERT model can show some bias toward the Muslim group. Some potential methods might include curating a clean dataset and adding more penalties for biased behavior of the model.

In short, this project lays a foundation for the efforts of identifying toxic comments and mitigating potential biased behavior of NLP models.

References

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion proceedings of the 2019 world wide web conference*, pages 491–500.

Conversation AI Jigsaw. 2019. Competition evaluation. [Online; accessed 29-Nov-2021].

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Kamran Kowsari, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes, and Donald Brown. 2019. Text classification algorithms: A survey. *Information*, 10(4):150.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2017. Mixed precision training. *arXiv preprint arXiv:1710.03740*.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 220–229.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144.

Model	**Accuracy**	**F1**	**mixed AUC score**
Logistic Regression	0.938	0.483	0.832
Word2Vec with fastText	0.941	0.528	0.824
DistillBERT	0.944	0.592	0.873

Table 1: Results of Different Models

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

A Appendix

A.1 Github Repository

The source code and documentation for this project are stored in this Github Repository.

A.2 Web App

We deploy a web app to show how different models behave. You can visit the web app here.

A.3 Sample Model Card

A sample model card for documenting model behavior can be found here.