Domain Adaptation Multi-task Deep Neural Network for Mitigating Unintended Bias in Toxic Language Detection

Farshid Faal[1,2], Jia Yuan Yu[1,2] and Ketra Schmitt[1,2]
[1]*Concordia Institute for Information System Engineering, Canada*
[2]*Concordia University, Montreal, Canada*

Keywords: Toxic Language Detection, Unintended Bias, Multi-task Learning, Domain Adaptation, Deep Neural Network, Contextual Language Model.

Abstract: As online communities have grown, so has the ability to exchange ideas, which includes an increase in the spread of toxic language, including racism, sexual harassment, and other negative behaviors that are not tolerated in polite society. Hence, toxic language detection within online conversations has become an essential application of natural language processing. In recent years, machine learning approaches for toxic language detection have primarily focused on many researchers in academics and industries. However, in many of these machine learning models, non-toxic comments containing specific identity terms, such as gay, Black, Muslim, and Jewish, were given unreasonably high toxicity scores. In this research, we propose a new approach based on the domain adaptation language model and multi-task deep neural network to identify and mitigate this form of unintended model bias in online conversations. We use six toxic language detection and identification tasks to train the model to detect toxic contents and mitigate unintended bias in model prediction. We evaluate our model and compare it with other state-of-the-art deep learning models using specific performance metrics to measure the model bias. In detailed experiments, we show our approach can identify the toxic language in conversations with considerably more robustness to model bias towards commonly-attacked identity groups presented in online conversations in social media.

1 INTRODUCTION

Identifying potential toxicity within online conversations has become an essential topic for social media platforms. Social media has become a viable route for people to express their views, and this has been a boon to large numbers of people, including minorities, who are able to connect with one another, share experiences and organize. While expressing oneself on these platforms is a human right that must be respected, inducing, and spreading toxic speech towards another group is an abuse of this privilege.

Toxicity in online conversations is defined as textual comments with threats, insults, obscene, disrespectful or rude content, or racism. In the last few years, there have been several studies on applying machine learning methods to detect toxic language in online content (Burnap and Williams, 2015; Davidson et al., 2019; Kumar et al., 2018). With the recent growth in the use of machine learning methods for the toxic language detection task, several researchers have identified that these classifiers have been shown to capture and replicate biases common in society (Wiegand et al., 2019; Borkan et al., 2019). A specific problem found in these classifiers is their sensitivity to frequently attacked identity groups such as gay, Muslim, Jewish, and Black, which are only toxic comments when combined with the right context. The source of this bias is the unbalanced representation of identity terms in a training dataset: terms like "gay" were often used in toxic comments; hence the models are over-generalized and learned to associate those terms with the toxicity label unfairly (Borkan et al., 2019; Dixon et al., 2018; Park et al., 2018).

In this research, our primary focus is to explore and propose a method for mitigating the unintended model bias in toxic language detection task. We propose a multi-task deep neural network (MTDNN) framework based on a domain adaptation language model that detects and identifies toxic language within online conversations. Recent studies demonstrate that multi-task learning can improve performance on various natural language understanding tasks while revealing novel insights about lan-

Faal, F., Yu, J. and Schmitt, K.
Domain Adaptation Multi-task Deep Neural Network for Mitigating Unintended Bias in Toxic Language Detection.
DOI: 10.5220/0010266109320940
In *Proceedings of the 13th International Conference on Agents and Artificial Intelligence (ICAART 2021) - Volume 2*, pages 932-940
ISBN: 978-989-758-484-8

guage modeling (Suresh et al., 2018; Liu et al., 2019). Furthermore, we consider a large-pretrained transformer model introduced by (Devlin et al., 2019) for our MTDNN as the language model, and we continue its pretraining on our dataset to have a domain-specific language model that tuned for the toxic language detection task. For evaluating our proposed approach on real data, we use the "Unintended Bias in Toxicity Classification" dataset published by the Google Jigsaw team (Google, 2019), which contains 1,804,874 comments from the Civil Comments platform. Google Conversation AI Team extended annotation for this dataset by human raters for different toxic conversational attributes. In our work, we consider evaluation metrics specifically designed to measure bias in the toxic detection model and compare it with other state-of-the-art deep learning models.

2 RELATED WORK

Research in the field of safety and security in social media has grown substantially in the last few years. A particularly relevant aspect of this research is offensive language detection in social networks. Previous studies have looked into various aspects of offensive languages, such as the use of abusive and aggression language (Kumar et al., 2018), bullying (Dadvar et al., 2013), hate and toxic language (Davidson et al., 2019; Burnap and Williams, 2015; Borkan et al., 2019). To this end, various datasets have been created to benchmark progress in the field (Wulczyn et al., 2017; Dixon et al., 2018; Google, 2019).

Recent studies introduced different machine learning methods for toxic language detection task (Mishra et al., 2019; Wulczyn et al., 2017; Dixon et al., 2018). The best performing systems introduced in these studies used deep learning approaches such as LSTMs and CNNs and Transformers (Dinan et al., 2019; Kumar et al., 2018). The unintended biases related to race, gender, and sexuality that yield high false-positive rates are investigated in recent studies (Burnap and Williams, 2015; Thomas et al., 2017). Furthermore, (Waseem, 2016) studied the correlation between annotation schemes, the annotators' identity, and reducing the effect of bias in machine learning models. A recent work (Dixon et al., 2018) investigated biases in the "Google Perspective API" classifier and revealed that several "social identity terms" are disproportionately represented in the dataset labeled as toxic, and this false-positive bias is caused by the model over-generalizing from the training data. In addition, recent works introduced metrics to quantify these unintended bias according to specific definitions (Friedler et al., 2019; Kleinberg et al., 2016; Menon and Williamson, 2018) and the importance of these metrics in evaluating the machine learning models is demonstrated in (Brennan et al., 2009; Buolamwini and Gebru, 2018). Among these works, Google conversation AI Team (Borkan et al., 2019) proposed metrics that are threshold agnostic, robust to class imbalances in the dataset, and provide more nuanced insight into the types of unintended bias present in the model. In our work, we use these particular evaluation metrics to evaluate the quality of our proposed approach in mitigating the model bias. The structure of our multi-task learning is influenced by Transformers-based multi-task learning frameworks introduced by (Liu et al., 2019). In (Liu et al., 2019), the author introduced a Multi-Task Deep Neural Network for learning representations across multiple natural language understanding tasks and demonstrate that multi-task learning leads to create more general representations to help adapt to new tasks and domains.

3 DATASET

In this work, for unintended bias evaluation on real data, we use the "Unintended Bias in Toxicity Classification" dataset published by the Google Jigsaw (Google, 2019). This dataset contains 1,804,874 comments from the Civil Comments platform made available at the end of 2017 to understand and improve online conversations. Google Conversation AI Team extended annotation for this dataset by human raters for different toxic conversational attributes. This dataset includes individual comments that are used to detect toxicity. Each comment in the dataset has a toxicity label with fractional values (between 0 and 1), representing the fraction of human raters who believed the attribute applied to the given comment. The comment with a label greater or equal to 0.5 will be considered the toxic class; otherwise, it is considered a non-toxic class. The total number of toxic comments in this dataset is 144334, which is 8% of all the comments are toxic comments. While all of the comments were labeled for toxicity, a subset of the dataset that includes 405130 comments has also been labeled with various identity attributes (non-exclusive), representing the presence of identities in the comments. Table 1 demonstrates all these identities with the number of toxic and non-toxic comments related to each one.

In our work, for training the MTDNN model, we create six tasks from the dataset. The first task, which is also the main task in our work, is toxic comment detection, which has two labels: toxic and non-toxic.

Table 1: Identities presented in the dataset with the number of toxic and non-toxic comments by each identity.

Identity Group	Identity attributes	Non-Toxic	Toxic
Gender	Female	63264	10426
	Male	68382	11797
	Transgender	5038	1082
	Other gender	2296	427
Religion	Christian	55915	5445
	Jewish	9290	1615
	Muslim	21007	5643
	Hindu	1361	196
	Buddhist	1204	162
	Atheist	1974	279
	Other religion	14710	2022
Race or Ethnicity	Asian	9746	1229
	Black	14097	5466
	White	22135	7813
	Latino	5813	1123
	Other race or ethnicity	16169	2698
Sexual Orientation	Heterosexual	2735	718
	Homosexual-gay-or-lesbian	11459	3848
	Bisexual	2800	530
	Other sexual orientation	3697	811
Disability	Physical disability	2779	448
	Intellectual or learning disability	1823	825
	Psychiatric disability or mental illness	8253	2412
	Other disability	3088	457

Since all the comments were labeled for toxicity, we consider all the data for this task. The goal of the first task is to detect whether the comment is toxic or non-toxic. Furthermore, we want to reduce the model bias towards the specific social identities in non-toxic comments. For this purpose, we create five more tasks to help the model to reduce identity bias in the toxicity prediction task. Since the dataset includes five identity groups in which each group consists of different identity attributes (Table 1), we create a task for each identity group to predict the identity attributes related to its identity group. For this purpose, we only considered the comments labeled for subgroup identities; hence the size of data for each task varies. It is important to note that these five tasks are multi-label text classification tasks, and more than one label may assign to a single comment in each task. Table 2 demonstrates these five tasks with the number of toxic and non-toxic comments in each task.

4 METHODOLOGY

In this section, we discuss our proposed approach for toxic language detection model. The training procedure of our proposed model consists of two stages: Domain adaptation masked language model pre-training and multi-task learning, discussed in detail in the following section.

4.1 Language Model Domain Adaptation

The first step in the training of our model is domain adaption for language modeling. We continue the pretraining of the language model on our dataset prior to classification tasks in this step. Recent studies showed that further pretraining on the related domain corpus could further improve the ability of the language model and achieved the state of the art performance on several text classification datasets (Sun et al., 2019). In this work, we use the BERT model architecture as a pretrained language model. BERT model is a stack of 12 Transformer encoder layers with 12 attention heads, a hidden size of 768, and total parameters of 110M. The BERT is pretrained on two semi-supervised tasks: masked language modeling (MLM) that predicts randomly masked input tokens and next sentence prediction (NSP) that predicts if two input sentences are adjacent to each other (Devlin et al., 2019). The BERT model is pretrained on general domain corpus; the BooksCorpus with 800M

Table 2: Distribution of data for each toxic identification task.

Task	Number of labels	Non-Toxic	Toxic
Gender identification	4	106526	18060
Religion identification	7	80145	11340
Race or Ethnicity identification	5	51555	13199
Sexual Orientation	4	15890	4644
Disability	4	13243	3582

words (Zhu et al., 2015) and English Wikipedia with 2500M words. The data distribution for the toxicity detection task is different from BERT general domain corpus. Hence, we further pretrain BERT with MLM and NSP tasks on our domain-specific dataset. For this purpose, we continue pretraining BERT on the training set that we prepared and discussed in section 3. The Transformer encoder's parameters are initialized by the pretrained BERT model, and then two semi-supervised prediction tasks, MLM and NSP, are utilized to continue pretraining the model parameters. The pretraining details and the model hyperparameters will discuss in section 5.

4.2 Multi-task Learning Framework

In machine learning, we usually train a single model or an ensemble of models on the desired dataset to optimize the model for a specific metric. This approach studies extensively and generally gives good results on a single task; however, when we focus on a single task, we ignore the information from the training signals of related tasks. We can enable our model to better generalize our original task by sharing representations between all related tasks in a multi-task learning approach. In this work, we explore and propose a method for training on multiple tasks to eventually produce separate parameter settings that perform well on each specific task. As discussed in section 3, we consider six tasks in our multi-task learning framework: One task for toxic comment detection and five tasks for group identity detection. The model jointly trained on these tasks to mitigate the bias in model prediction towards commonly attacked identities in the toxic classification task.

The architecture of our MTDNN model is shown in Figure 1. The model includes two main parts: shared layers that shared the domain-adaptive BERT model parameters across all tasks and task-specific layers that are unique for each task and produce output for each task separately. The shared layers' input is constructed by summing the corresponding token embeddings, segment embeddings, and position embeddings for a given input token. The BERT model is the shared representation across all tasks, and in a multi-task learning model, it learns the representations using multi-task objectives and the pretraining. The task-specific layers of the multi-task learning model include six separate modules dedicated to each task, where each module contains a feed-forward neural network with the number of outputs equals to the number of labels in each task. During training, each task-specific module takes the contextualized embeddings generated by BERT layers from input sequences and produces a probability distribution for its target labels. Different factors, such as the size of a dataset and the task's complexity, must be considered to set the proportion of data for training of each task. Furthermore, in multi-task learning, achieving good performance on one task can hinder the performance of other tasks (Raffel et al., 2019; McCann et al., 2018). Given these concerns, exploring a proper strategy for setting the proper proportion of data for each task is necessary.

A recent study (McCann et al., 2018) showed that in multi-task training for natural language processing application, the anti-curriculum schedules strategy (Bengio et al., 2009) provides better results compared to the fully joint sampling strategy. Among our six tasks, toxic detection with two labels (task-1) is a less complicated classification task than the other five group identification tasks with multiple identity labels. We use the anti-curriculum schedules strategy, and we start the training with five group identification tasks (task-2 to task-6); then, after the model train for two epochs, we add the first task and continue the training with all six tasks in a fully joint sampling strategy for four epochs. For training our multi-task neural network, first, we randomly initialized the task-specific model parameters. Then, for the first two epochs, a mini-batch is selected among five group identification tasks (task-2 to task-6), and the model is trained according to the task-specific objectives. After two epochs, for the rest of the training, task-1 will be added, and the training with all six tasks in a fully joint sampling strategy continues. In our work, the cross-entropy loss is used as the objective for all tasks. Since the toxic detection task (task-1) is a binary classification task, the toxic detection loss L_{tc} is defined as:

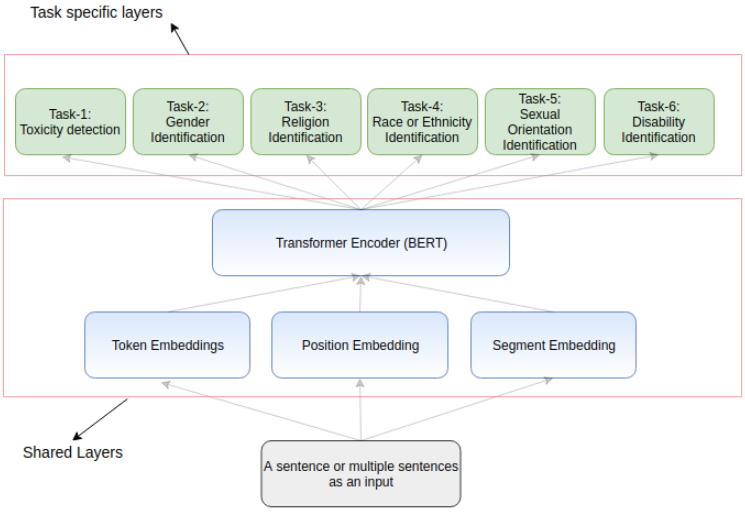

Figure 1: The architecture of the multi-task deep neural network model.

$$L_{tc} = -\sum_{i}^{N} [c_i \log \hat{y}_i + (1 - c_i) \log (1 - \hat{y}_i)] \quad (1)$$

Where c is ground-truth labels, $\hat{y}_i$ is the probability predicted by the model as class c and N is the number of training data. The loss function for identity detection tasks (task-2 to task-6) is defined as:

$$L_{id} = -\sum_{i}^{N} [c_i \log \sigma(\hat{y}_i) + (1 - c_i) \log \sigma (1 - \hat{y}_i)] \quad (2)$$

Where $\sigma(.)$ is the Sigmoid function. The total model loss L_{total} is calculated as $L_{total} = \sum_{t=1}^{T} L_t$ where L_t is the loss for each task, and T is the total number of tasks ($T = 6$ in our work). The anti-curriculum schedules strategy for training the MTDNN is summarized in Algorithm 1, and the fully joint sampling strategy is summarized in Algorithm 2.

Algorithm 1: Multi-task training with anti-curriculum schedules strategy.

```
Initialize the model parameters θ randomly ;
Initialize the shared layers for all five identity
  detection tasks (task-2 to task-6) with
  domain-adaptive pretrained BERT ;
Pack the dataset of five tasks into
  mini-batches of D_2, D_3, D_4, D_5 and D_6 ;
for two epochs do
    Merge mini-batches to create D' where
      D' = D_2 ∪ D_3 ∪ D_4 ∪ D_5 ∪ D_6;
    foreach mini-batch in D' do
        Compute task-specific loss based on
          Equation 2 ;
        Compute total loss L' as sum of all
          losses from each task: L' = Σ_{t=2}^{6} L_t ;
        Update the model parameters based
          on total loss ;
    end
end
```

5 EXPERIMENTS

In this section, we first describe the hyperparameters of our model and then compare the performance of our model to three other baseline models that we will discuss in the following.

5.1 Experimental Settings

We follow the settings prescribed for pretraining BERT by (Devlin et al., 2019) to continue pretraining on our training dataset. We continue pretraining the BERT with a batch size of 32 and maximum tokens of 512 for two epochs over the training set. We use Adam algorithm with weight decay fix (Loshchilov and Hutter, 2018) with learning rate of $5e-5$, Adam beta weights of $\beta_1 = 0.9$, $\beta_2 = 0.999$, Adam epsilon of $1e-6$ and weight decay of 0.01. The dropout probability of 0.1 is used on all layers.

The implementation of our multi-task learning is based on the PyTorch implementation described in (Liu et al., 2019). For multi-task training, we use AdamW algorithm with learning rate of $2e-5$, Adam beta weights of $\beta_1 = 0.9$, $\beta_2 = 0.999$, Adam epsilon of $1e-6$ and weight decay of 0.01. The maximum number of epochs was set to 6 with a batch size of 32. we also set the dropout rate of all the task-specific layers as 0.1. Furthermore, we use the

Algorithm 2: Multi-task training with fully joint strategy.

```
Initialize the model parameters Θ from
  anti-curriculum schedules strategy
  (Algorithm 1) ;
for four epochs do
    Merge mini-batches to create D^total where
      D^total = D1 ∪ D2 ∪ D3 ∪ D4 ∪ D5 ∪ D6;
    foreach mini-batch in D^total do
        Compute task-specific loss based on
          Equation 1 and 2 ;
        Compute total loss L^total as sum of all
          losses from each task:
          L^total = Σ_{t=1}^{6} L_t ;
        Update the model parameters based
          on total loss ;
    end
end
```

wordpieces tokenizer with the maximum sequence length of 256 tokens. In our experiments, we perform 6-fold cross-validation on the dataset. In each fold, 90% of the training data is set aside for training, and 10% is used for validation.

5.2 Comparison Models

We compare our proposed model with two other deep learning models that are described as follows:

- **BERT + Fine-tuning:** This model was introduced in (Devlin et al., 2019) and considered as the current state-of-the-art workflow for fine-tuning the BERT for a specific single task. In this model, we use pretrained BERT as a language model, and for each task, we fine-tune the BERT separately and independently.
- **Domain-adaption BERT + Fine-tuning:** In this model, we continue pretraining BERT on the training dataset, and then we fine-tuned BERT for each task independently. We name this model as Adaptive-BERT-fine-tuning in our evaluations. We compare the state of the art baseline (BERT-fine-tuning) with this model to observe the model performance improvement yields through language model adaptation in our task.

5.3 Evaluation Metrics

In our work, we consider two groups of evaluation metrics. The first group includes Precision, Recall, and F1-scores metrics, and the second group of evaluation metrics is unintended bias evaluation metrics introduced and specified by the Google conversation AI team (Borkan et al., 2019). Google Jigsaw introduced three metrics to measure mitigation of unintended bias by a model, namely Subgroup AUC, Background Positive Subgroup Negative (BPSN) AUC, and Generalized Mean of Bias AUCs (GMB-AUC). To calculate these three metrics, we divide the data by identity subgroup and the metrics compare the subgroup to the rest of the dataset, which we call the "background" data. By dividing the dataset into background and identity subgroups, four distinct subsets were created: negative (non-toxic) examples in the background, negative examples in the subgroup, positive (toxic) examples in the background, and positive examples in the subgroup. Hence, three AUCs are defined to measure negative and positive misordering between these four subsets. Let D_{bg}^{-} be the negative examples in the background set, D_{bg}^{+} be the positive examples in the background set, D_{is}^{-} be the negative examples in the identity subgroup, and D_{is}^{+} be the positive examples in the identity subgroup. We can define four bias identification metrics as follows:

Subgroup-AUC
The Subgroup-AUC is defined as follows:

$$\text{AUC}_{sub} = \text{AUC}\left(D_{bg}^{-} + D_{bg}^{+}\right) \tag{3}$$

The AUC_{sub} calculates AUC using only the examples from the subgroup and indicates an understanding of the model within a specific subgroup. A High value represents that the model can distinguish between toxic and non-toxic comments in the subgroup.

BPSN-AUC
The BPSN-AUC is defined as:

$$\text{AUC}_{bpsn} = \text{AUC}\left(D_{is}^{+} + D_{bg}^{-}\right) \tag{4}$$

A model with a high BPSN-AUC score is capable of reducing biases towards a specific subgroup identity, and it is less prone to confuse non-toxic comments that mentioned the identity subgroup with toxic comments that did not mention.

The GMB-AUC
The GMB-AUC combines the per-subgroup bias AUCs into one overall metric and defined as follows:

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

where M_p is the $p-th$ power-mean function, m_s is the bias metric calculated for subgroup s and N is the number of identity subgroups.

5.4 Results Analysis

The performance of our proposed model for toxicity detection (task-1) is summarized in Table 3. The Adaptive-BERT-MTDNN model outperforms all other baselines in all four metrics. As we can observe from Table 3, the Recall and Precision both have improved in Adaptive-BERT-MTDNN, which means the model can identify more toxic comments in a dataset with less false-positive rates. The GMB-AUC metric in Table 3 indicates how much the model can distinguish valid toxic comments from non-toxic comments that contain specific subgroup identities. Hence, in our work, the improvement in the value of this metric indicates an improvement in reducing the unintended bias in model prediction, which is our primary goal in this work. The results in Table 3 indicate that our proposed approach is capable of classifying toxic comments and distinguish non-toxic comments from toxic comments with any identities presented in the comments better than other states of the art baselines. Furthermore, when we compare the multi-task learning approach with single-task learning approaches, it is observed that utilizing the multi-task learning framework improved the quality of the model to distinguish between toxic and non-toxic comments when the specific identities appear in the context. By comparing the improvement obtained between the multi-task learning approach and the domain-adaptation language model method, we can conclude that the multi-task learning approach has a much more significant impact on metrics improvement than the domain adaptation language model. However, combining the domain-adaptation language modeling with the multi-task learning approach brings the best improvement for toxic identification and bias mitigation in toxic language detection task.

Table 4 indicates the average Subgroup-AUC metric, and Table 5 indicates the average BPSN-AUC metric related to each identity group. We calculate these two metrics by averaging all Subgroup-AUC and BPSN-AUC values in each identity group. As the results in Table 4 indicate, the Adaptive-BERT-MTDNN outperforms the BERT-fine-tuning for all identity groups except the "Disability" group, which there is no improvement for this group; and also the most significant improvement belongs to the "Race or Ethnicity" identity group with 1.1% improvement. For average BPSN-AUC values in Table 5, it observed Adaptive-BERT-MTDNN outperforms the BERT-fine-tuning for all identity groups, where the most significant improvement belongs to the "Sexual Orientation" identity group with 2.6% improvement. Figures 2 demonstrate the Subgroup-AUC and BPSN-AUC values for each identity obtained with our approach, and Figure 3 demonstrates these two metrics for the BERT-fine-tuning model for comparison. As we can see from these two figures, the most significant improvement in the BPSN-AUC metric belongs to "homosexual gay or lesbian" subgroup with 2.9% improvement and, for the Subgroup-AUC metric, the most significant belongs to "bisexual" with 4.5% improvement. As we discussed earlier, the amount of data in multi-task learning is a critical factor for this approach; hence in subgroups with a higher proportion of data, the improvement is more stable than subgroups with much fewer data. Overall, the results show that learning multiple group identification tasks in parallel improved the shared language model between tasks and mitigated the unintended model bias, which was our main goal in this work.

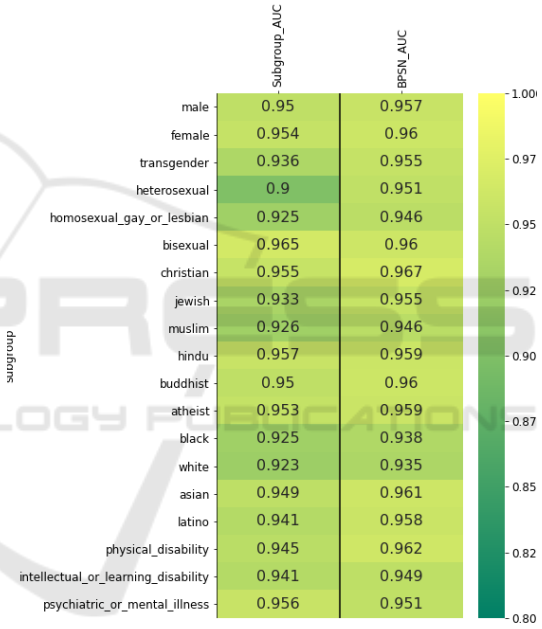

Figure 2: The Subgroup-AUC and BPSN-AUC metrics obtained from Adaptive-BERT-MTDNN for each identity subgroups.

6 CONCLUSION

We introduce a new approach for mitigating the unintended model bias towards commonly attacked identities in the toxic language detection task based on a multi-task deep neural network and a domain-adaptation language model. We show that the multi-task deep neural network classifier that is jointly trained on multiple identity detection tasks is indeed more robust to unintended model bias towards commonly attacked identities in online conversations. Furthermore, we demonstrate that continuing pre-

Table 3: Binary classification performance of all models on toxic detection task.

Model	Precision	Recall	F1-score	GMB-AUC
BERT-fine-tuning	0.8533	0.7293	0.7864	0.9499
Adaptive-BERT-fine-tuning	0.8586	0.7622	0.8075	0.9508
Adaptive-BERT-MTDNN	0.8708	0.8995	0.8849	0.9567

Table 4: The average Subgroup-AUC metric for each identity group.

Model	Gender	Religion	Race or Ethnicity	Sexual Orientation	Disability
BERT-fine-tuning	0.938	0.939	0.924	0.924	0.947
Adaptive-BERT-MTDNN	0.947	0.946	0.935	0.93	0.947

Table 5: The average BPSN-AUC metric for each identity group.

Model	Gender	Religion	Race or Ethnicity	Sexual Orientation	Disability
BERT-fine-tuning	0.94	0.946	0.933	0.926	0.944
Adaptive-BERT-MTDNN	0.959	0.958	0.948	0.952	0.954

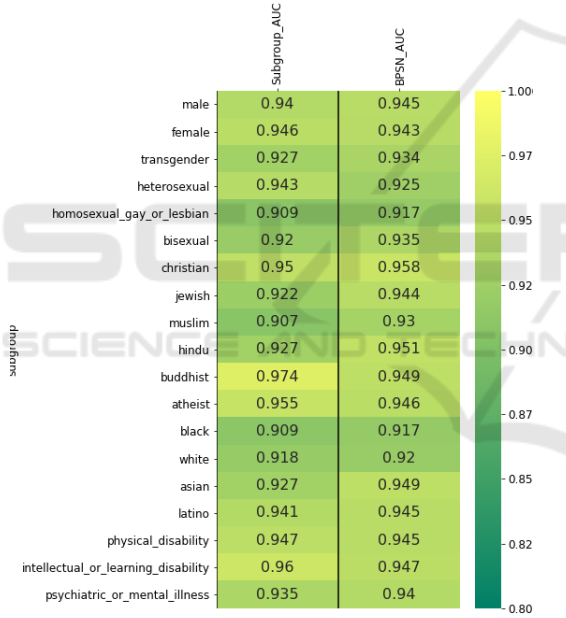

Figure 3: The Subgroup-AUC and BPSN-AUC metrics obtained from BERT-fine-tuning for each identity subgroups.

training the language model on domain related to the task dataset gives improved model performance in the toxic detection task. To evaluate our approach, we choose the dataset that includes more than 1.8 million online comments released by Google Jigsaw, and we compare our approach with another state-of-the-art deep learning model in terms of specific metrics designed to measure the unintended bias. The evaluation results demonstrate that our approach brings state-of-the-art results in toxic language detection task and mitigates the unintended biases in a model without harming the overall model quality.

REFERENCES

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009). Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA. Association for Computing Machinery.

Borkan, D., Dixon, L., Sorensen, J., Thain, N., and Vasserman, L. (2019). Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, page 491–500, New York, NY, USA. Association for Computing Machinery.

Brennan, T., Dieterich, W., and Ehret, B. (2009). Evaluating the predictive validity of the compas risk and needs assessment system. *Criminal Justice and Behavior*, 36(1):21–40.

Buolamwini, J. and Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91.

Burnap, P. and Williams, M. L. (2015). Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. *Policy & Internet*, 7(2):223–242.

Dadvar, M., Trieschnigg, D., Ordelman, R., and de Jong, F. (2013). Improving cyberbullying detection with user context. In *European Conference on Information Retrieval*, pages 693–696. Springer.

Davidson, T., Bhattacharya, D., and Weber, I. (2019). Racial bias in hate speech and abusive language detection datasets. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 25–35, Florence, Italy. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. 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), pages 4171–4186.

Dinan, E., Humeau, S., Chintagunta, B., and Weston, J. (2019). Build it break it fix it for dialogue safety: Robustness from adversarial human attack. In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4536–4545. Association for Computational Linguistics.

Dixon, L., Li, J., Sorensen, J., Thain, N., and Vasserman, L. (2018). Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, page 67–73, New York, NY, USA. Association for Computing Machinery.

Friedler, S. A., Scheidegger, C., Venkatasubramanian, S., Choudhary, S., Hamilton, E. P., and Roth, D. (2019). A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 329–338.

Google (2019). Unintended bias in toxicity classification, https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification.

Kleinberg, J. M., Mullainathan, S., and Raghavan, M. (2016). Inherent trade-offs in the fair determination of risk scores. *CoRR*, abs/1609.05807.

Kumar, R., Ojha, A. K., Malmasi, S., and Zampieri, M. (2018). Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 1–11, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Liu, X., He, P., Chen, W., and Gao, J. (2019). Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496.

Loshchilov, I. and Hutter, F. (2018). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

McCann, B., Keskar, N. S., Xiong, C., and Socher, R. (2018). The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Menon, A. K. and Williamson, R. C. (2018). The cost of fairness in binary classification. volume 81 of *Proceedings of Machine Learning Research*, pages 107–118, New York, NY, USA. PMLR.

Mishra, P., Tredici, M. D., Yannakoudakis, H., and Shutova, E. (2019). Author profiling for hate speech detection. *CoRR*, abs/1902.06734.

Park, J. H., Shin, J., and Fung, P. (2018). Reducing gender bias in abusive language detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2799–2804, Brussels, Belgium. Association for Computational Linguistics.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2019). Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*.

Sun, C., Qiu, X., Xu, Y., and Huang, X. (2019). How to fine-tune bert for text classification? In Sun, M., Huang, X., Ji, H., Liu, Z., and Liu, Y., editors, *Chinese Computational Linguistics*, pages 194–206, Cham. Springer International Publishing.

Suresh, H., Gong, J. J., and Guttag, J. V. (2018). Learning tasks for multitask learning: Heterogenous patient populations in the icu. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 802–810. ACM.

Thomas, D., Dana, W., Michael, M., and Ingmar, W. (2017). Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media. ICWSM*.

Waseem, Z. (2016). Are you a racist or am I seeing things? annotator influence on hate speech detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142, Austin, Texas. Association for Computational Linguistics.

Wiegand, M., Ruppenhofer, J., and Kleinbauer, T. (2019). Detection of Abusive Language: the Problem of Biased Datasets. 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)*, pages 602–608, Minneapolis, Minnesota. Association for Computational Linguistics.

Wulczyn, E., Thain, N., and Dixon, L. (2017). Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, page 1391–1399, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*.