# Fighting Algorithmic Bias Using Adversarial Networks

Shweta Chopra, Nupur Baghel, Shubham Annadate, and Rajalakshmi Dorairaj Shanmugasundaram
University of Pennsylvania
CIS 419/519 Applied Machine Learning

## 1 Introduction

The field of Algorithmic Bias is witnessing growing interest. Within the space of Natural Language Processing, cases of gender and racial bias (Sun et al.) have been encountered in Sentiment Analysis algorithms and Word Embeddings. While many cases of bias are largely driven by problems in the data used to train these algorithms, there is growing literature on possible algorithmic techniques that may be leveraged to produce fairer outcomes. The technique of Adversarial Debiasing (Wadsworth et al.) is one such method that provides an algorithmic solution to the problem of bias around protected identities.

In this project we will be developing a fair classifier, that given a dataset of comments from Wikipedia talk pages, is able to successfully identify toxic comments that may be obscene, threatening or attacking in nature. The challenge with such a task is that certain identity groups that are overrepresented in toxic comments, tend to become more associated with toxicity. In our case we will be focusing on gender bias in situations where females are overrepresented in toxic comments as compared to males. Indeed, a research study implemented by Amnesty International demonstrates how online platforms like Twitter tend to be toxic environments for women and other gender and sexual minorities (Tox). Thus, classifiers tend to overpredict toxicity for females versus for males, leading to a higher rate of false positives for the former. To mitigate this unwanted bias, we will utilize Adversarial Debiasing (Wadsworth et al.) and prevent our classifier from predicting toxicity on the basis of gender. Thus by weakening these unwanted associations between our target variable and our protected identity, we aim to arrive at a more 'just' and 'fair' classifier.

## 2 Literature Review

In 2018, Zhang et. al (Zhang et al., 2018) proposed adversarial networks as a technique for fighting model bias. This was a variation on generative adversarial networks proposed by Goodfellow, et. al (Goodfellow et al., 2014). The framework proposed involved the generator learning with respect to a protected attribute, like gender. This translated into a structure where the generator prevents the discriminator from being able to predict gender under a given overarching task. In their paper, Zhang et.al. (Zhang et al., 2018) were able to demonstrate an improvement in fairness on an income classification task, using their Adversarial Debiasing approach, facing only a 1.5% compromise in overall accuracy. This process of Adversarial Debiasing can be generalized to any setting where the model uses a gradient based learning including both regression and classification tasks and is hence, suitable for our task at hand.

The question of how much information the adversary is fed from the predictor is an important one. Zhang et. al. (Zhang et al., 2018) formulated a model where the output of the predictor forms the input to the adversary. Beutel et. al. (Beutel and Chi, 2017) earlier implemented a variation on this with the predictor and adversary having a shared hidden layer, resulting in a multi-headed model. Their performance on a recidivism dataset saw a compromise on the AUC score of only 0.02 (drop from 0.72 to 0.70). Beutel et. al. (Beutel and Chi, 2017) also provide important observations of the data representations necessary for strong debiasing performance. They demonstrated that having a balanced representation of protected identities in the data gives a strong boost to the adversary performance. Additionally, they were able to show that the Adversarial Debiasing approach can be successful for even small data sizes of 500 instances.

## 3 Objective

Following from prior implementations of Adversarial Debiasing, we aim to make the following contributions through our work.

1. Extend the limited implementations of this approach to a new domain of Toxicity Classification.

2. Examine the fairness-performance tradeoff

faced by our model, an issue that has not been commented on in much detail within previous implementations.

3. Test the generalizability of our debiased model on a dataset from a different domain, that it has not been trained on.

## 4 Dataset

The dataset has been taken from Kaggle (Dat) where it was made available by Google's Conversation AI team and the Civil Comments platform. It consists of 1.8 million comments from Wikipedia Talk Pages. It was labelled for both toxicity of the comments, as well as the presence of several protected identities.

After taking a subset of the data labelled for either male or female we were left with 60k records, out of which 34k belonged to females. We also noted that 12% of total comments were toxic, while 14.6% of female related comments were toxic, demonstrating an overrepresentation of women in toxic comments (Figure 1). Each comment was annotated for Toxicity and Identity by at least 4 people making it more trustworthy.

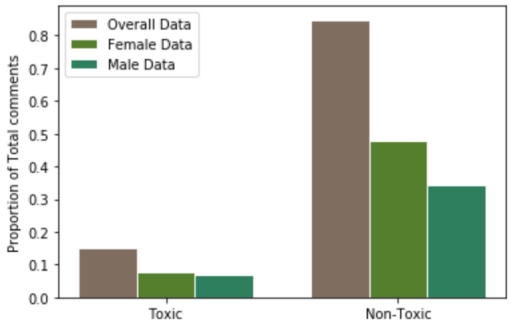


Figure 1: Distribution of Toxic and Non-Toxic Comments (Wikipedia Talk Pages)

Given our dataset, we expect to witness a biased model in the absence of any debiasing effort, with higher toxicity prediction and false positive rates for females. Eventually, with the introduction of our adversarial approach, we expect the model bias to drop significantly. Given prior literature that has explored adversarial debiasing, we would expect a small drop in model performance in exchange for improved fairness.

## 5 Methodology

### 5.1 Evaluation Metrics

Evaluation of both performance and fairness form an important part of our inquiry.

#### 5.1.1 Performance

Given the imbalanced distribution of toxicity within our dataset, we utilized the F1-Score as the primary measure of model performance. Accuracy has also been reported but must be read while keeping in mind the distribution of toxic and non-toxic comments in our dataset. During model training precision and recall were also separately monitored to ensure a relatively balanced F1-Score representation.

#### 5.1.2 Fairness

Rising interest in the field of algorithmic fairness has brought about several definitions of how model fairness can be measured. Here, we move forward with three commonly used measures (Zhang et al., 2018), defined below in the context of our implementation:

1. **Demographic Parity:** A predictor satisfies Demographic Parity if its prediction is independent of the protected class. In our case of gender being the protected class, and toxicity being the target class variable, this can be understood using the following simplification: $P(Y_{pred} = Toxic \mid Gender = Male) = P(Y_{pred} = Toxic \mid Gender = Female)$. Thus, the rate of toxicity prediction for both genders should be close to equal.

2. **Equality of Opportunity (True Positive Parity):** A predictor satisfies Equality of Opportunity with respect to a class Y, if the probability of a true prediction is independent of the protected class, conditioned on the true target class being Y. In our case of gender being the protected class, and toxicity being the target class variable, this can be understood using the following simplification: $P(Y_{pred} = Toxic \mid Gender = Male, Y_{true} = Toxic) = P(Y_{pred} = Toxic \mid Gender = Female, Y_{true} = Toxic)$. Thus, the rate of true positives for both genders should be close to equal.

3. **False Positive Parity:** A predictor satisfies False Positive Parity with respect to a class

Y, if the probability of a false prediction is independent of the protected class, conditioned on the true target class being Y. In our case of gender being the protected class, and toxicity being the target class variable, this can be understood using the following simplification: $P(Y_{pred} = Toxic|Gender = Male, Y_{true} = Non-Toxic) = P(Y_{pred} = Toxic|Gender = Female, Y_{true} = Non-Toxic)$. Thus, the rate of false positives for both genders should be close to equal.

Since these fairness measures are boolean (True-False) in nature, we implement them, instead, in the form of differences in probabilities for the Female and Male protected classes. This allows us to utilize a continuous measure of difference between scores for both genders which we aim to reduce to zero through our adversarial debiasing approach.

### 5.2 Models

We utilized three models for our experiments.

#### 5.2.1 Baseline XGBoost Classifier

To get a baseline understanding of how well a strong simple classifier would perform on our task of toxicity classification, we utilized the distributed gradient boosting library XGBoost. This was implemented on top of a TF-IDF based bag of words representation of our comment text. This allowed us to have a benchmark, against which we could measure the performance of our regular classifier, thus ensuring we built a strong classifier to begin with.

#### 5.2.2 Classifier

Our regular classifier was built in the form of a neural network. We utilized an uncased pretrained BERT layer to form our comment text representations, using a PyTorch implementation of Google Research's BERT model, created by HuggingFace (Git). On top of this, we added one dropout and three linear layers to create the Classifier.

#### 5.2.3 Classifier + Adversary

For our final model that implements the Adversarial Debiasing, we utilized our regular classifier, itself, as the predictor portion of the model. For the adversary, we utilized a distinct shallow network of two linear layers. The point of connection between the two networks, the classifier and the adversary, was that the hidden penultimate layer of the classifier formed the input to the adversary.

Running our data through these three models, we calculated both performance and fairness metrics on a held-out test dataset to report our results.

### 5.3 Generalization

Our final step was to test whether our model, trained on Wikipedia data to reduce gender bias in a toxicity classification task, can be generalized to another online platform. For this, we collected 1000 comments from the social news aggregation and discussion website Reddit. Since a pre-existing dataset annotated with gender and toxicity labels was not already available, we annotated the dataset ourselves.

1. **Gender Annotation:** This step was automated by utilizing gender specific nouns and pronouns to classify comments as either Male or Female. Comments containing both genders were left out of the dataset we prepared.

2. **Toxicity Annotation:** Each comment was annotated with 1 for toxic or 0 for non-toxic by a single annotator.

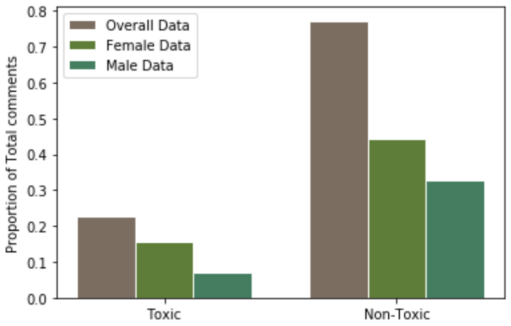


Figure 2: Distribution of Toxic and Non-toxic Comments (Reddit Dataset)

There may be subjective bias in the annotation, however we believe it is a satisfactory first step to test the generalizability of our model.

## 6 Analysis

### 6.1 Data Preparation

As suggested in literature, class imbalance limits the ability of the learner to differentiate between classes efficiently. Hence, as a part of preprocessing we upsampled our dataset for toxicity, considering an equal proportion of toxic and

non-toxic comments for training. The validation set, however was kept separate and stood for the true distribution of data. Other preprocessing steps were trimming sentences to 128 characters and tokenizing sentences to a format suitable for input to BERT.

### 6.2 Parameter Tuning

In order to tune the Adversary weighing parameter value $\lambda$, we plotted the model performance (F1 score) vs Fairness Metrics for varying $\lambda$ values. As $\lambda$ was increased from 1 to 10, the parity values decreased and our model became more fair (Figure 3). But, this came at the cost of reduced classification accuracy. As a sweet spot for both performance and fairness, $\lambda = 3$ was selected for further analysis. On our dataset, we noted that increased the $\lambda$ value beyond 3 led to steep drops in model performance on the task of toxicity classification itself.

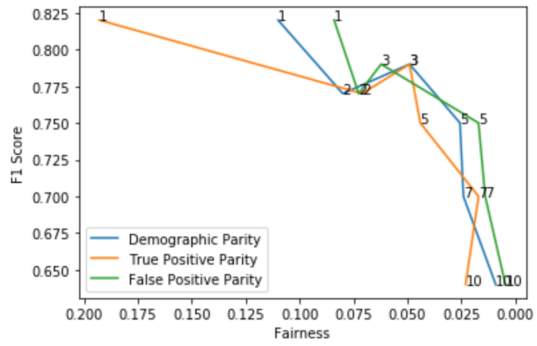


Figure 3: Effect of varying $\lambda$ on Performance(F1) and Fairness

Besides the $\lambda$ value we also tuned learning rates for different layers within our models. Ultimately a learning rate of 0.00001 was selected for the pretrained BERT layers, to minimize interference with the pretrained weights. For all other layers, a learning rate of 0.001 was used. We also experimented with different comment sequence lengths as inputs to our models. A maximum sequence length of 128 performed best given our dataset, given its underlying nature.

### 6.3 Adversarial Training

The Adversarial Training process chosen was adapted from the work of Stijn Tonk (Tonk). The process was divided into two stages:

1. **Pretraining Stage:** Both the Classifier and Adversary networks were pre-trained for 3 epochs each to bring down their losses on their individual tasks. While the Classifier was pretrained the Adversary weights were frozen. Then the Adversary was pretrained on top of the already trained Classifier, while keeping the latter's weights frozen. This ensured both networks were performing adequately on their individual tasks. Different number of epochs for pretraining were tested and 3 epochs were found to be satisfactory in terms of reasonably low loss, yet leaving room for the adversarial stage to have effect.

2. **Adversarial Stage:** The adversarial stage involves the "fight" between the Classifier and Adversary over some number of iterations. Within each iteration, the Adversary uses the penultimate hidden layer representation of the Classifier as an input, to predict gender, minimizing its own loss during training. This is carried out for 1 epoch. Following that, within the same iteration, the Classifier is trained for one sample batch of size 32 with the goal of minimizing its own loss, but maximizing the loss of the Adversary. A hyperparameter, $\lambda$ is used to tune the weight that the Classifier assigns to the task of maximizing the Adversary loss. The classifier, thus, updates its weights with the goal of making itself stronger at toxicity prediction but making the Adversary weaker as a gender predictor. The effect of this is to arrive at a Classifier that predicts toxicity without assigning much weight to gender related features in the data.

   Adversary Objective:
   $min_{\theta adv}[Loss_z(\theta_{clf}, \theta_{adv})]$
   Classifier Objective:
   $min_{\theta clf}[Loss_y(\theta_{clf}) - \lambda Loss_z(\theta_{clf}, \theta_{adv})]$

## 7 Results

### 7.1 Pre-Training

During the initial phase of the experiment, the Classifier and Adversary were sequentially pretrained, while keeping the weights of the other network static. This helped to stabilize each model's performance (Figure 4).

### 7.2 Impact of Debiasing

The debiased Classifier + Adversary model was able to comfortably beat the baseline XGBoost

| Model | Train Accuracy | Train F1 | Test Accuracy | Test F1 |
|---|---|---|---|---|
| XGBoost | 83.13 | 83.12 | 59 | 64.12 |
| Classifier | 94.56 | 94.50 | 80.6 | 82.63 |
| Classifier + Adversary | 92.08 | 92.07 | 76.04 | 80.08 |

Table 1: Comparison of Accuracies and F1 scores among models

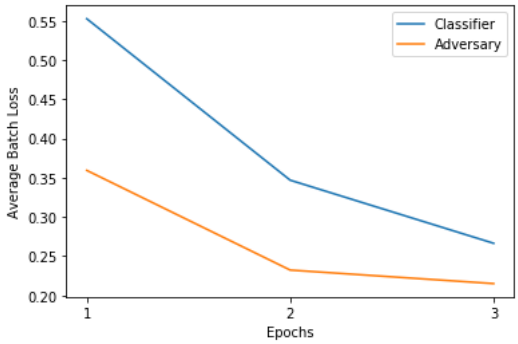


Figure 4: Results of Pre-Training Phase

Model. We observed that the Adversary caused a small drop in performance over the biased Classifier (Table 1).

The Classifier + Adversary model performed best at 12 iterations (Figure 5), with an F1 score of 80.08 and an average Fairness Metric of 0.013. The $\lambda$ value was chosen to be 3 through tuning.

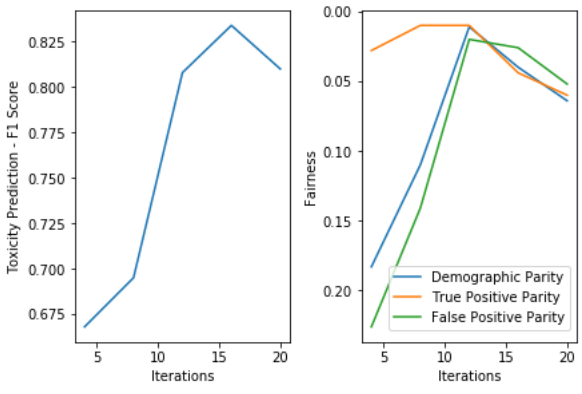


Figure 5: Adversarial Training Phase

The biased classifier demonstrated bias across all three of the fairness metrics we examined. As compared to the biased Classifier, the debiased Classifier + Adversary had a significant improvement in fairness across all the three fairness metrics we had chosen (Figure 6).

### 7.3 Generalizability

Both the Classifier and Classifier + Adversary models generalized well to the new dataset, in terms of toxicity performance. The performance dropped a little after implementing the Classifier + Adversary, however fairness did substantially improve. The regular Classifier model did demonstrate considerable bias as expected.

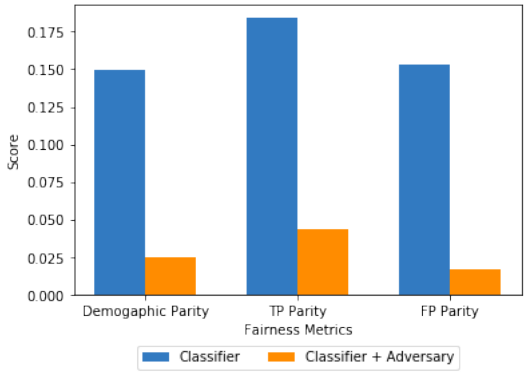


Figure 6: Comparison of Fairness Metrics between Classifier and Classifier + Adversary

| Model | Test Accuracy | Test F1 |
|---|---|---|
| Classifier | 78.0 | 79.0 |
| Classifier + Adversary | 75.6 | 77.7 |

Table 2: Comparison of performance before and after debiasing on Test Dataset

On training the Classifier + Adversary model with the Wikipedia data and testing it against the Reddit dataset, we observed that the performance was best around 12 iterations as well (Figure 7). The F1 score was 77.7 (Table 2), and the average of the three Fairness metrics was 0.037.

We observed that the Classifier + Adversary model had a significant improvement on the test dataset with Reddit comments across all the three Fairness Metrics (Figure 8).

## 8 Discussion

In comparison to the Adversarial Debiasing implementations we discussed in our Literature Review, our debiased model performed at par. We witnessed only a small drop in performance in terms

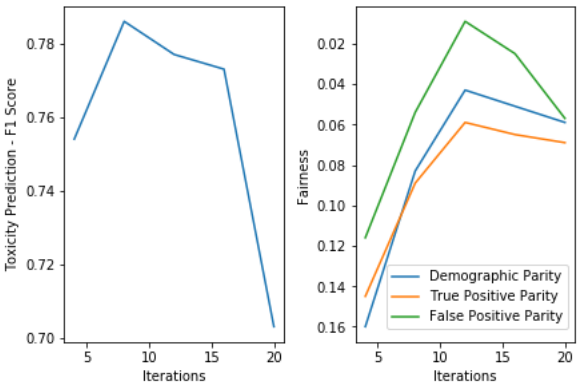


Figure 7: F1 Score and Fairness Metrics on Reddit Dataset

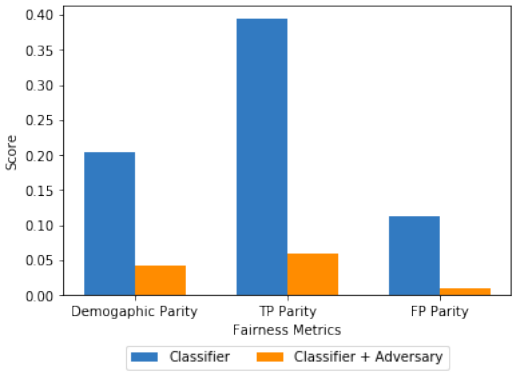


Figure 8: Classifier vs Classifier + Adversary Fairness Metrics on Reddit Dataset

of toxicity classification, measured by the F1-Score. On the other hand we demonstrated a drastic improvement in terms of fairness metrics. No other implementation we found, attempted to generalize and test their model on an unseen dataset from a related but different domain. In that regard, we were largely successful in our ability to generalize to the Reddit dataset we created, with both our Regular Classifier and in particular our debiased Classifier + Adversary models, generalizing well. A few points to note in relation to these results:

1. The successful performance of our toxicity classifier, post Adversarial Debiasing indicates that the dataset we used was rich and contained adequate representations of toxicity that were not gendered in nature. Thus, even after attempting to eliminate the role of gendered language within our toxicity classifier, we saw strong results.

2. One of the implementations (Zhang et al., 2018) covered in our literature review provided less information as input to its Adversary Network. It limited the input to the output layer of the classifier. In our initial attempts, this structure did not work well for us. Given time constraints once we had a successful model we were unable to go back to reattempt different input data representations for the Adversary. In our future work that will be an interesting line of inquiry for us, to see the effects of the chosen representation, on performance.

3. The successful generalization of our models to a dataset from a different online platform signals two things. One, that Reddit comments and Wikipedia comments may share some amount of structure that allowed for successful generalization. This may not hold true incase of platforms like Twitter that are less like discussion portals. Second, our underlying model built using pretraining BERT seems relatively strong and navigates across similar domains with ease.

## 9 Conclusion

In this project we successfully developed a debiased, fair toxicity classifier using the technique of Adversarial Debiasing. While our implementation faced a tradeoff between performance and fairness, the final model performed at par with other implementations in literature. We also successfully generalized our fair classifier to a dataset from another online platform, Reddit, an indication that a comprehensive solution to the problem of bias in toxicity classification may be possible to build. The code and a video presentation of the project can be found online.

As next steps, we would be interested in extending this implementation to a multi-label case as besides gender, there are several other identity groups that also may be targeted in online comments. We also limited our fairness measures to just three, in this project. In recent times newer measures of fairness have been defined that offer most customizable definitions of fairness, given a particular context. We would be interested to study these measures in more depth, select more appropriate metrics for our task and explore the impact of our debiased classifier on them.

## References


Huggingface github repository.

Kaggle competition: Jigsaw unintended bias in toxicity classification.

Toxic-twitter methodology.

Chen J. Zhao Z. Beutel, A. and E. H. Chi. 2017. Data decisions and theoretical implications when adversarially learning fair representations.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. in advances in neural information processing systems.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. Mitigating gender bias in natural language processing:literature review.

Stijn Tonk. Towards fairness in ml with adversarial networks.

Christina Wadsworth, Francesca Vera, and Chris Piech. Achieving fairness through adversarial learning: an application to recidivism prediction.

Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning.