Balancing out Bias: Achieving Fairness Through Balanced Training

Xudong Han[1] **Timothy Baldwin**[1,2] **Trevor Cohn**[1]
[1]The University of Melbourne
[2]MBZUAI
xudonghl@student.unimelb.edu.au, {tbaldwin,t.cohn}@unimelb.edu.au

Abstract

Group bias in natural language processing tasks manifests as disparities in system error rates across texts authorized by different demographic groups, typically disadvantaging minority groups. Dataset balancing has been shown to be effective at mitigating bias, however existing approaches do not directly account for correlations between author demographics and linguistic variables, limiting their effectiveness. To achieve Equal Opportunity fairness, such as equal job opportunity without regard to demographics, this paper introduces a simple, but highly effective, objective for countering bias using balanced training. We extend the method in the form of a gated model, which incorporates protected attributes as input, and show that it is effective at reducing bias in predictions through demographic input perturbation, outperforming all other bias mitigation techniques when combined with balanced training.[1]

1 Introduction

Natural Language Processing (NLP) models have achieved extraordinary gains across a variety of tasks in recent years. However, naively-trained models often learn spurious correlations with other demographics and socio-economic factors (Hendricks et al., 2018; Lu et al., 2018; Bolukbasi et al., 2016; Park et al., 2018), leading to disparities across author demographics in contexts including coreference resolution, sentiment analysis, and hate speech detection (Badjatiya et al., 2019; Zhao et al., 2018; Li et al., 2018a; Díaz et al., 2018).

Two popular approaches for mitigating such biases are: (1) balancing each demographic group in training, either explicitly via sampling (Zhao et al., 2018; Wang et al., 2019) or implicitly via balancing losses for each group (Höfler et al., 2005; Lahoti et al., 2020); and (2) removing demographic information from learned representations (Li et al., 2018a; Wang et al., 2019; Ravfogel et al., 2020; Han et al., 2021b).

While balancing methods have been shown to be successful, they have not been tested extensively in NLP. In this paper, we focus on Equal Opportunity fairness (EO: Hardt et al. (2016)), which requires non-discrimination across demographics within the "advantaged" outcome labels, and adapt three balanced training approaches for debiasing. In addition, we propose a new objective for balanced training, which can be used for proxy optimization of EO fairness. We first provide a theoretical justification for our approach, and then conduct experiments on two benchmark datasets which show that our proposed objective is highly effective at achieving EO fairness while maintaining competitive accuracy.

Even when the training data is balanced, ignoring demographic-specific features can lead to bias (Wang et al., 2019; Lahoti et al., 2020), due to differences in language use across demographics (Hovy, 2015). There is thus a fine line to be walked in terms of optimizing for linguistic variables associated with different demographic groups (potentially boosting overall model accuracy), and ensuring model fairness.

Inspired by work in domain adaptation on learning domain-specific representations that generalize across domains (Bousmalis et al., 2016; Li et al., 2018b), we propose a gated model, which incorporates author demographics as an *input* to generate group-specific representations but also generalizes across demographic groups. We show that when combined with instance reweighting during training, this technique leads to substantial bias reductions over leading debiasing techniques, typically with higher predictive accuracy. We also introduce a second means of bias reduction through tailoring gating coefficients of the trained model, which al-

[1] Code available at https://github.com/HanXudong/Achieving_Fairness_Through_Balanced_Training/

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11335–11350
December 7-11, 2022 ©2022 Association for Computational Linguistics

lows for fine-tuning of the accuracy–fairness trade-off. Our experiments over two benchmark datasets for language debiasing show that our techniques are competitive with much more complex state-of-the-art methods for debiasing in situations where the demographic attribute is not known at test time, and provide substantial gains over the state-of-the-art when the protected attribute is observed.

2 Balanced Training

Despite their simplicity and versatility, balanced training approaches have received limited attention in prior work in NLP. In this section, we propose a novel objective for balanced training, which we show to be a proxy for the EO. We further review three balanced training approaches, discuss their objectives, and highlight their differences over our proposed method.

2.1 Problem Formulation

In this paper, we focus on bias mitigation for NLP classification tasks. Formally, we assume a dataset $\mathcal{D} = \{(x_i, y_i, g_i)\}_{i=1}^{n}$ where $x_i \in X$ is a d-dimensional input text representation vector, $y_i \in Y$ denotes the main task label (e.g., sentiment), and $g_i \in G$ represents the private attribute associated with x_i (e.g., author gender).

A standard model M is trained to predict Y given X, while debiasing methods generally aim to learn a model M' that is fair *wrt* G by considering $X \times G$ together.

Let $\mathcal{X}$ be the task loss and n be the number of observed instances in the dataset $\mathcal{D}$. The overall empirical risk is written as $\mathcal{L} = \frac{1}{n}\sum_i \mathcal{X}(y_i, \hat{y}_i)$, which can be rewritten as the aggregation of subsets: $\mathcal{L} = \sum_{\mathrm{y}}\sum_{\mathrm{g}} \frac{n_{\mathrm{y,g}}}{n}\mathcal{L}_{\mathrm{y,g}}$, where $n_{\mathrm{y,g}}$ is the number of instances with target label y and demographic attribute g, and $\mathcal{L}_{\mathrm{y,g}}$ is the empirical loss corresponding to the subset: $\mathcal{L}_{\mathrm{y,g}} = \frac{1}{n_{\mathrm{y,g}}}\sum_i \mathcal{X}(y_i, \hat{y}_i)\mathbb{1}(y_i = \mathrm{y}, g_i = \mathrm{g})$.

Furthermore, we use $*$ to denote marginalization, for example, $n_{*,\mathrm{g}} = \sum_{\mathrm{y}} n_{\mathrm{y,g}}$. Let p be the target label distribution, and $\tilde{p}$ be the empirical probability based on the training dataset.

2.2 Fairness Measurement

Equality of Opportunity (EO) is widely used in previous work (Hardt et al., 2016; Ravfogel et al., 2020; Han et al., 2021a), and measures the difference in true positive rate (TPR, aka Recall) across all groups, based on the notion that the positive outcome represents 'advantage', such as getting a job or a loan. Essentially, the difference (gap) in TPR reflects the degree to which different groups lack equal opportunity (with lower numbers indicating greater equity).

2.3 Towards Equal Opportunity

Without loss of generality, we illustrate with the binary case of $y \in \{T, F\}$ and $g \in \{0, 1\}$. Recall that the equal opportunity metric is satisfied if a binary classification model has an equal positive prediction rate for the advantaged class. Assuming the advantaged class is $y = T$, the equal opportunity is measured by the TPR gap between protected groups, i.e., $\mathrm{Recall}_{\mathrm{g}=0} - \mathrm{Recall}_{\mathrm{g}=1}$. Our proposed objective function for equal opportunity is:

$$\begin{aligned}\mathcal{L}^{EO} &= \frac{n_{T,*}}{n}\frac{1}{2}\sum_{\mathrm{g}\in\{0,1\}}\mathcal{L}_{T,\mathrm{g}} + \sum_{\mathrm{g}\in\{0,1\}}\frac{n_{F,\mathrm{g}}}{n}\mathcal{L}_{F,\mathrm{g}}\\ &= \sum_{\mathrm{g}\in\{0,1\}}\frac{n_{T,\mathrm{g}}}{n}\frac{n_{T,*}}{2n_{T,\mathrm{g}}}\mathcal{L}_{T,\mathrm{g}} + \sum_{\mathrm{g}\in\{0,1\}}\frac{n_{F,\mathrm{g}}}{n}\mathcal{L}_{F,\mathrm{g}}\end{aligned}$$

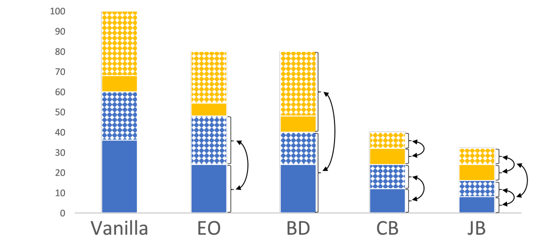
Compared to the vanilla objective, the weights of instances with target label T are adjusted. Specifically, the reweighting term $\frac{n_{T,*}}{2n_{T,\mathrm{g}}} > 1$ for the minority group, and < 1 for the majority group.

From CE to TPR Cross-entropy is an estimate of the TPR at the mini-batch level when considering a subset of instances with the same target label. Recall that the CE loss for binary classification of an instance is $-[y_i \cdot \log(\hat{p}(y_i)) + (1 - y_i) \cdot \log(1 - \hat{p}(y_i))]$, where $\hat{p}(y_i)$ is the predicted probability of y_i being True. Taking $y = T$ for a certain demographic group g as an example,

$$\begin{aligned}\mathcal{L}_{T,\mathrm{g}} &= \frac{1}{n_{T,\mathrm{g}}}\sum_i \mathcal{X}(y_i, \hat{y}_i)\mathbb{1}(y_i = T, g_i = \mathrm{g})\\ &= -\frac{1}{n_{T,\mathrm{g}}}\sum_i \log(\hat{p}(y_i))\mathbb{1}(y_i = T, g_i = \mathrm{g}).\end{aligned}$$

Essentially, minimizing $\mathcal{L}_{T,\mathrm{g}}$ is proportionate to maximizing the TPR of demographic group g. That is, at the minibatch level, $-\mathcal{L}_{T,\mathrm{g}}$ is an estimator of $\log(p(\hat{y} = T|y = T, g = \mathrm{g}))$, which is the log-TPR of group g. Given this, our proposed objective minimizes the TPR gap by focusing on the log-TPR difference across demographic groups.

Beyond binary labels & demographic attributes Our proposed objective generalizes to higher ordering labels and demographic attributes



	Vanilla	EO	BD	CB	JB
▨ N & 0	32	25.6	32	8	8
■ N & 1	8	6.4	8	8	8
▨ P & 0	24	24	16	12	8
■ P & 1	36	24	24	12	8

Figure 1: Balanced training through downsampling w.r.t. different objectives over a toy dataset for illustrative purposes. **N/P** = main task, **0/1** = protected attribute. The groups at either end of each arrow are resampled to be the same size. **EO** = our proposed objective in Section 2.3. **BD**, **CB**, and **JB** refer to balanced demographics, conditional balance, and joint balance, resp. (Section 2.4).

trivially. The equal opportunity metric was originally designed for binary classification, under the assumption of a single advantaged class $y = T$. To satisfy the multi-class target label case, we adjust the equal opportunity to consider the one-vs-all setting, and measuring the TPR of each target class. Our proposed objective then becomes $\sum_{\mathrm{y}} \sum_{\mathrm{g}} \frac{n_{\mathrm{y,g}}}{n} \frac{n_{\mathrm{y},*}}{|G| \times n_{\mathrm{y,g}}} \mathcal{L}_{\mathrm{y,g}}$.

2.4 Balanced Training Objectives

We now formally describe the objective functions of three established balanced training approaches, and discuss their applications. We provide a toy example in Figure 1 to illustrate the differences between these objectives. In Appendix E, we provide more details about the mapping from previous work to these objectives in our framework.

Balanced Demographics (BD) Zhao et al. (2018) augment the dataset according to the demographic label distribution (making $p(G)$ uniform) for bias mitigation in the context of coreference resolution. Although their gender-swapping approach is not directly applicable to our tasks, we adapt the general objective function as $\mathcal{L}^G = \frac{1}{|G|} \sum_{\mathrm{y}} \sum_{\mathrm{g}} \frac{n_{\mathrm{y,g}}}{n_{*,\mathrm{g}}} \mathcal{L}_{\mathrm{y,g}}$, where $|G|$ is the number of distinct labels of G.

Since $\mathcal{L}^G$ only encourages the model to equally focus on different demographic groups, it does not explicitly capture the correlation between G and Y, and as a result, does not achieve Equal Opportunity fairness.

Conditional Balance (CB) In a vision context, Wang et al. (2019) down-sample the majority demographic group within each class, so that on a per-class basis, it does not dominate the minority group (i.e. $p(G|Y)$ is uniform for all Y), giving the objective function: $\mathcal{L}^{G|Y} = \frac{1}{|G|} \sum_{\mathrm{y}} \frac{n_{\mathrm{y},*}}{n} \sum_{\mathrm{g}} \mathcal{L}_{y,\mathrm{g}}$.

This is the closest formulation to ours, as it also captures the conditional independence between G and Y. However, it captures both the TPR and TNR, while our method and EO fairness only focus on the TPR. In the multi-class target label case, our EO objective recovers the formulation of $\mathcal{L}^{G|Y}$.

Joint Balance (JB) Lahoti et al. (2020) employ instance reweighting for structural data classification such that demographics and classes are jointly balanced, leading to the objective: $\mathcal{L}^{G,Y} = \frac{1}{|G| \times |Y|} \sum_{\mathrm{y}} \sum_{\mathrm{g}} \mathcal{L}_{y,\mathrm{g}}$.

JB can be treated as a combination of the classic long-tail learning objective and the CB objective ($p(G, Y) = p(G|Y)p(Y)$). On the one hand, JB is equivalent to CB when Y has already been balanced, which is the case for the dataset Moji (Section 4.2), and CB is not a suitable objective for achieving EO fairness in this case. On the other hand, when Y is imbalanced, JB not only requires CB but also focuses more on long-tail main task classes, making it highly vulnerable to the size of minority groups.

2.5 Achieving the Objective

In this paper, we focus on two ways of achieving the target objective: (1) instance reweighting, which manipulates the weight of each instance during training; and (2) down-sampling, which preprocesses the dataset before training.

Taking the joint balance (JB) as an example, instance reweighting reweights each instance in inverse proportion to the frequency of the combination of its main label and demographic label, $\frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i, g_i) \in \mathcal{D}} \tilde{p}^{-1}(G = g_i, Y = y_i) \mathcal{X}(y_i, \hat{y}_i)$, where $\mathcal{X}$ is the task loss, and $\hat{y}_i$ denotes the model prediction given input text x_i.

The other approach, down-sampling, subsamples non-minority instances to derive a balanced training dataset, such that $\tilde{p}(g, y) = \frac{1}{|G| \times |Y|}, \forall g \in G, y \in Y$. Specifically, let $\mathcal{D}_{y,g} = \{(x_i, y_i, g_i) | y_i = y, g_i = g\}_{i=1}^{n}$ denote a subset for training. We sample without replacement to get a target subset $\mathcal{D}^*_{y,g}$ such that $|\mathcal{D}^*_{y,g}| = \min\{|\mathcal{D}_{y',g'}|, \forall y' \in Y, g' \in G\}$. The sampled subsets are merged to form the training set.

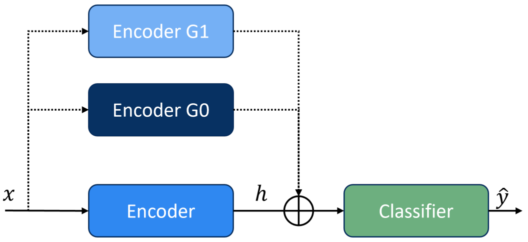


Figure 2: Gated model architecture. Given the input vector x, e.g. a text representation, the model has a shared encoder component and $|G|$ encoder components, one for each demographic group.

3 Demographic Factors Improve Fairness

Ignoring demographic-specific features can lead to bias even when the training data has been balanced (Wang et al., 2019; Lahoti et al., 2020). Instead, as suggested by Hovy and Yang (2021), addressing demographic factors is essential for NLP to get closer to the goal of human-like language understanding, and increase fairness. Our approach to dealing with this is, rather than removing demographic information, to use a gated model that uses demographic labels as input.

As can be seen in Figure 2, the gated model consists of $(1+|G|)$ encoders: one shared encoder, and a dedicated encoder for each demographic group in G.[2] Formally, let E denote the shared encoder, E_j denote the encoder for the j-th demographic group, C denote the classifier, and g_i be a 1-hot input such that $g_{i,j}$ is 1 if the instance (x_i, g_i, y_i) belongs to the j-th group, and 0 otherwise. The prediction for an instance is: $\hat{y}_i = C(h_i^s, h_i^g)$, where $h_i^s = E(x_i)$ and $h_i^g = \sum_{j=1}^{|G|} g_{i,j} E_j(x_i)$. The two inputs are concatenated and input to the classifier C.

Intuitively, the shared encoder learns a general representation, while each group-specific encoder captures group-specific representations.

Our setting differs from other debiasing methods in that we assume the demographic attribute is available at training and prediction time, while techniques such as adversarial training (Li et al., 2018a) and INLP (Ravfogel et al., 2020) only require the attribute for training. This richer input allows for more accurate predictions, courtesy of the demographic-specific encoder, but limits applicability at test time. For better applicability, we also relax this requirement by replacing demographic factors with a non-informative prior in Section 4.7.

4 Experimental Results

In this section we first introduce our experimental settings, and then report and discuss our results. In Appendix B, we provide detailed settings for reproducing our experiments.

4.1 Evaluation Metrics

Following Ravfogel et al. (2020), we use overall accuracy as the performance metric, and the separation criterion to measure fairness in the form of TPR GAP and TNR GAP: the true positive rate and true negative rate differences between demographic groups. For both GAP metrics, smaller is better, and a perfectly fair model will achieve 0. For multi-class classification tasks, we follow Ravfogel et al. (2020) in reporting the quadratic mean (RMS) of TPR GAP over all classes. In a binary classification setup, TPR and TNR are equivalent to the TPR of the positive and negative classes, respectively, so we employ the RMS TPR GAP in this case also.

Throughout this paper, we report accuracy and GAP results as mean values ± standard deviation over the test set, averaged across five independent runs with different random seeds.

In contrast to single-objective evaluation, evaluation of fairness approaches generally reports both fairness and performance at the same time. Typically, there is no single method that achieves both the best performance and fairness, making comparison between different fairness methods difficult. This problem has been widely studied in the literature on *multi-objective learning* (Marler and Arora, 2004). For ease of comparison between approaches, we adopt the *compromise solution* (Salukvadze, 1971) to fairness evaluation, and introduce 'distance to the optimum' (DTO). Specifically, the *compromise solution* aims to minimize the difference between the candidate point and a *utopia point*. In our case, the candidate points are ordered pairs $(\text{Accuracy}, \text{GAP})$, denoting the accuracy and fairness of debiasing methods, and the *utopia point* (optimum) represents the hypothetical system which achieves the highest-achievable accuracy and fairness for the dataset., i.e., $(\max(\text{Accuracy}), \min(\text{GAP}))$. Following Vincent and Grantham (1981); Vincent (1983), DTO

[2] Strictly speaking, it is possible to achieve a similar effect with $|G|$ encoders by merging one group with the shared encoder, and using post-hoc correction to separate out the general from the group-specific representation (Kang et al., 2020).

is calculated as the Euclidean distance between the optimum and the results for a given method. Lower is better for this statistic, with a minimum of 0. In Appendix C, we provide a more detailed explanation of DTO, including a step-by-step example calculation.

In addition, we are also interested in the efficiency of the different debiasing approaches and report each method's average training time.[3]

4.2 Dataset

Following Ravfogel et al. (2020), we conduct experiments over two NLP classification tasks — sentiment analysis (Moji) and biography classification (Bios) — using the same dataset splits. In Appendix A, we provide analysis of the dataset distribution.

Moji: This sentiment analysis dataset was collected by Blodgett et al. (2016), and contains tweets that are either African American English (AAE)-like or Standard American English (SAE)-like. Each tweet is annotated with a binary 'race' label (based on language use: either AAE or SAE), and a binary sentiment score determined by (redacted) emoji contained in it.

Bios: The second task is biography classification (De-Arteaga et al., 2019), where biographies were scraped from the web, and annotated for binary gender and 28 classes of profession.

4.3 Models

We first implement a "Standard" model on each dataset, without explicit debiasing. On the Moji dataset, we follow Ravfogel et al. (2020) in using DeepMoji (Felbo et al., 2017) as the encoder to get 2304d representations of input texts. Ravfogel et al. (2020) and Subramanian et al. (2021) used uncased BERT-base (Devlin et al., 2019) as their Standard model for the Bios dataset, taking the 'CLS' token as the source of a fixed text representation, without further fine-tuning. However, we found that taking the average of all contextualized token embeddings led to an accuracy improvement of 1.4% and GAP fairness improvement of 2.4%. Given this, we use 768d 'AVG' representations extracted from the pretrained uncased BERT-base model.

Model	**Accuracy**↑	**GAP** ↓	**DTO** ↓
Standard	82.3 ± 0.0	16.0 ± 0.5	0.093
BD (Zhao et al., 2018)	82.3 ± 0.0	15.6 ± 0.2	0.089
JB (Lahoti et al., 2020)	74.7 ± 0.3	7.4 ± 0.3	0.092
EO	79.4 ± 0.1	9.7 ± 0.6	**0.043**

Table 1: Results for balanced training methods on the Bios test set. **EO**: our proposed objective in Section 2.3. **BD** and **JB** are baselines from Section 2.4. **Bold** = best trade-off.

4.4 Balanced Training Approaches

Since the Moji dataset has been artificially balanced for main task and demographic labels, balanced training based on $p(g)$ makes no difference, and moreover, the results for $p(g|y)$ and $p(g, y)$ will be identical. Given this, we focus on the Bios dataset for comparing different balanced training objectives.[4]

Table 1 shows the results of balanced training using the different objectives. Compared to the Standard model, balanced training with different objectives are all able to reduce bias, and the objective proposed by Lahoti et al. (2020) achieves the lowest TPR GAP. However, in terms of accuracy–fairness trade-off, our proposed approach outperforms all other models, which is not surprising as it is designed to achieve better equal opportunity fairness. Based on these results, hereafter, we only report balanced training with our proposed EO objective (BTEO).

4.5 Main Results

We report results over the sentiment analysis and biography classification tasks in Table 2. The baseline models are: **Standard**, which is a naively-trained MLP classifier; **INLP** (Ravfogel et al., 2020), which removes demographic information from text representations through iterative nullspace projection; **Adv** (Li et al., 2018a; Wang et al., 2019; Zhao et al., 2019), which performs protected information removal through adversarial training; and **DAdv** (Han et al., 2021b), which also uses adversarial training but with multiple adversaries subject to an orthogonality constraint, and represents the current best baseline models.

On the Moji dataset, compared to the Stan-

[3] Testing on Titan X and RTX 3090, all models have roughly identical inference time.

[4] As Bios is a multi-class classification task and our proposed approach generalizes to **BD** in this case, there is no need to include Wang et al. (2019) in our comparison.

		Moji				Bios			
Method	**Model**	**Accuracy↑**	**GAP ↓**	**DTO ↓**	**Time↓**	**Accuracy↑**	**GAP ↓**	**DTO ↓**	**Time↓**
Baselines	Standard	71.6 ± 0.1	31.0 ± 0.3	0.261	1.0	82.3 ± 0.0	16.0 ± 0.5	0.110	1.0
	INLP	68.5 ± 1.1	33.8 ± 3.9	0.300	14.0	70.5 ± 0.5	6.7 ± 0.9	0.145	6.3
	Adv	74.3 ± 0.4	22.2 ± 3.7	0.163	36.1	81.1 ± 0.1	12.7 ± 0.3	0.077	1.3
	DAdv	74.5 ± 0.3	18.5 ± 2.0	0.123	109.4	81.1 ± 0.1	12.6 ± 0.3	0.076	2.4
Ours	BTEO	74.0 ± 0.2	21.5 ± 0.4	0.155	0.8	79.4 ± 0.1	9.7 ± 0.6	0.057	0.7
	BTEO +Gate *	74.9 ± 0.2	13.8 ± 0.3	0.072	0.8	79.4 ± 0.1	9.2 ± 0.2	**0.053**	0.7
	Gate *	64.8 ± 0.1	65.2 ± 0.9	0.640	1.0	82.4 ± 0.1	19.2 ± 0.3	0.144	1.0
	Gate$^{\text{soft}}_{\text{RMS}}$ *	73.5 ± 0.2	7.1 ± 0.3	**0.019**	1.0	80.5 ± 0.1	11.1 ± 0.3	0.063	1.0

Table 2: Results over the sentiment analysis (Moji) and biography classification (Bios) tasks. DTO is measured by the normalized Euclidean distance between each model and the ideal model, and lower is better. **Bold** = best trade-off within category. Normalized time is reported relative to Standard, which takes 35 secs and 16 mins for Moji and Bios, respectively. The reported times are the average times divided by that of Standard. * indicates that the model requires the demographic attribute at test time.

dard model, BTEO simultaneously increases accuracy and mitigates bias, leading to results competitive with Adv and better than INLP. Although BTEO does not outperform the best baseline models DAdv, it leads to performance–fairness trade-offs that are competitive with the other debiasing methods.

On the Bios dataset, BTEO again leads to performance–fairness trade-offs that outperform the baseline methods. However, different to the Moji dataset, BTEO does not further improve accuracy, improving fairness by 5.3% absolute at the cost of 2.9% accuracy.

In terms of training time, existing debiasing methods (esp. DAdv on Moji) incur a substantial overhead, while balanced training is much more frugal: around 1.3 times faster (because of the reduction in training data volume).

In addition to evaluating BTEO, we also combine Gate with BTEO, which achieves a better performance–fairness balance, as shown in Table 2. This is consistent with our argument that, rather than removing demographic information, properly used demographic factors can further reduce biases. Indeed, the BTEO +Gate consistently outperforms the current best baseline models model DAdv on both datasets.

In Appendix F.1, we also show that BTEO can be combined with DAdv and INLP, leading to better bias mitigation.

4.6 Gated Model

If the training dataset is imbalanced and contains spurious correlations between task labels and demographic attributes, a naively trained model will learn and possibly amplify dataset biases. The Gate model, with its explicit conditioning and group-specific encoding, will be particularly vulnerable to bias.

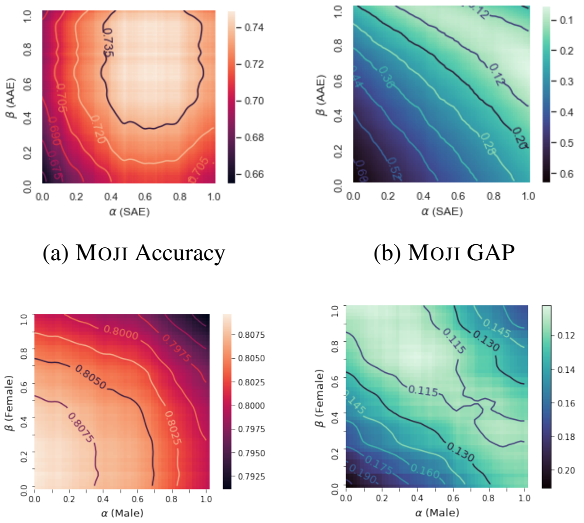
Table 2 shows that, on both datasets, the Gate model increases the accuracy but amplifies bias (e.g., GAP of 65 on Moji): as it uses demographic information directly to make predictions, it is highly vulnerable to bias in the training dataset.

Intuitively, the only objective of Gate training is standard cross-entropy loss, which has been shown to lead to bias amplification under imbalanced training without regularization. The gate components explicitly rely on demographic information, and thus become a strong indicator of predictions due to spurious correlations between the main task label and demographic labels in the training set.

Balanced training approaches act as regularizers in preventing the model from learning and amplifying spurious correlations in training.

4.7 Soft Averaging

Although the gated model naturally requires the demographic attribute at test time, we also evaluate a condition where this is not available. Instead, we take a Bayesian approach by evaluating $p(y|x) = \sum_g p(g)p(y|g,x)$, where we can control the prior explicitly. For example, under a uniform demographic attribute prior, we simply average the predictions $p(y|x,g)$ and $p(y|x,\neg g)$. This Bayesian approach can be approximated by soft averaging, whereby the activation of all demographic-specific encoders are uniformly averaged inside the model, i.e., $g_{i,j} = \frac{1}{|G|}$, rather than selecting only one in the standard gated model (i.e., $g_{i,:}$ is 1-hot).



(a) Moji Accuracy

(b) Moji GAP

(c) Bios Accuracy

(d) Bios GAP

Figure 3: Accuracy and GAP of α and β settings for Moji and Bios. The axes refer to the propensity to change the gold group in gating the encoder components, and the bottom left point $\alpha = \beta = 0$ is the Gate model using true demographic inputs. Lighter shading denotes better performance.

When the protected attribute is **observed** at test time the soft averaging method may still prove useful, which we use as a means for fine-tuning the balance between accuracy and bias. Figure 3 shows an example for the prior fine-tuning. Specifically, we consider non-uniform encoder averaging conditioned on the gold protected attribute, g^*. Let α and β denote to what extent the 1-hot labels are manipulated according to the value of g^* as 0 and 1 respectively, leading to the soft labels $\begin{bmatrix}\alpha & 1-\alpha\end{bmatrix}$ and $\begin{bmatrix}1-\beta & \beta\end{bmatrix}$. I.e., the two specific encoders are weighted by either α and $1-\alpha$, or $1-\beta$ and β, respectively, according to the value of g^*. Values of $\alpha, \beta < 0.5$ mean the protected label is (softly) preserved, while values > 0.5 mean the label is flipped.

In cases where the model is biased towards or against a demographic group, it may be advantageous to use these two additional parameters to correct for this bias, by disproportionately using the other group's encoder.

We now employ the Bayesian "soft averaging" approach to gating, and mitigating bias at inference time. Note that this does not involve retraining the model, as the soft averaging happens at test time.

Figure 3 shows accuracy and GAP results from tuning the coefficients on development data for the basic Gate model. The results show that $\alpha = \beta = 0.5$ is a reasonable default setting, however gains may be possible for non-uniform prior settings.

Model	Size	Accuracy↑	GAP ↓	DTO ↓
Standard	257k	82.3 ± 0.0	16.0 ± 0.5	0.093
RW + **BD**	257k	82.3 ± 0.0	15.6 ± 0.2	0.089
RW + **JB**	257k	74.7 ± 0.3	7.4 ± 0.3	0.092
RW + **EO**	257k	75.7 ± 0.2	13.9 ± 0.4	0.107
DS + **BD**	237k	82.1 ± 0.1	15.9 ± 0.3	0.092
DS + **JB**	5k	66.1 ± 0.1	10.9 ± 0.4	0.200
DS + **EO**	37k	79.4 ± 0.1	9.7 ± 0.6	**0.043**

Table 3: Results for balanced training methods on the Bios test set. "RW" = instance reweighting; "DS" = dataset down-sampling; and "Size" = the number of instances in the training dataset. **Bold** = best trade-off.

To demonstrate the power of adjusting these parameters, we take the trained Gate model, and then optimize α and β over the development set, and report the corresponding results on the test set. We select the parameter values that achieve the lowest development GAP, provided accuracy is above a threshold. [5] The results are reported in Table 2, under Gate $^{\text{soft}}_{\text{RMS}}$. On the Moji dataset, our results show that Gate with soft averaging consistently outperforms the Standard and Gate models without balanced training. In terms of GAP, the model is substantially better than all other models, while remaining competitive in terms of accuracy. The Bios dataset is noisier, meaning there are bigger discrepancies between the development and test datasets. However, we achieve a good performance–fairness trade-off at a level comparable to the much more complex INLP and DAdv models.

5 Analysis

5.1 Reweighting vs. Down-sampling

Table 3 shows the results of the naively-trained MLP model ("Standard") and six balanced-training methods, all based on the same architecture as Standard. Corpus down-sampling ("DS") removes instances from majority groups and thus leads to less training data and overall lower accuracy than instance reweighting ("RW").

When using **BD** as the objective, both RW and DS perform similarly to the Standard model, as the overall gender distribution is quite balanced,

[5] The $[\alpha, \beta]$ values are [0.64, 0.99] and [0.38, 0.72] over Moji and Bios, respectively, We also experimented with adjusting the gating coefficients for Gate + BTEO, in which case there was no benefit from using non-zero α or β.

Model	**Parameters**	**Accuracy**	**GAP**
Standard	782402	71.6	31.0
BTEO	782402	74.0	21.5
BTEO +Gate	2346002	74.9	13.8
Standard Large	2887202	71.7	31.4
BTEO Large	2887202	73.9	20.9

Table 4: Results over Moji. Large modes have larger hidden sizes to achieve similar number of parameters of our proposed Gate model.

which can also be seen in the size of the training data for DS + **BD**. Both RW + **JB** and RW + **EO** reduce bias and performance, but RW + **JB** outperforms RW + **EO** in terms of the performance–fairness trade-off, in that RW + **JB** achieves similar performance but substantially better fairness (6.6% absolute improvement in GAP). However, **JB** is not as effective as **EO** when combined with DS, due to the big drop in the volume of training data.

5.2 Gate vs. Large

Compared to Standard, Gate involves more model parameters, and an important question is whether the gains of Gate models are simply because of the larger parameter space.

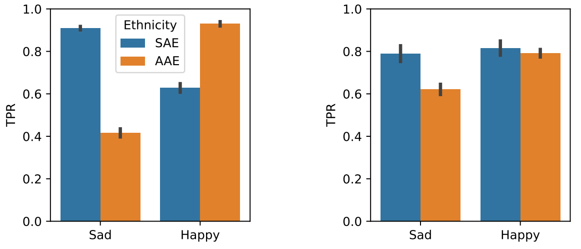
To explore this question, we conduct experiments over Moji by controlling the number of parameters of Standard and compare it with Gate models. Specifically, we employ a larger Standard model, namely Large, which has more hidden units within each hidden layer, leading to roughly the same number of parameters as the Gate model.

Table 4 shows the results for Standard Large and BTEO with Large. Comparing Standard and BTEO with the Large versions, it can be seen that increasing the number of parameters does not increase performance or fairness.

Despite BTEO +Gate having fewer parameters than BTEO Large, it achieves a substantial improvement in terms of fairness, confirming our argument that Gate is superior to existing models.

5.3 Debiased predictions

Figure 4 compares the true positive rates (TPR) of the Standard method and our proposed method (BTEO + Gate), on the basis of which the TPR GAP is measured. Compared with the Standard model, the debiased model improves the TPR results of the worse-performing groups for both classes at the cost of a slight reduction in TPR for the better-performing group. Overall, however, BTEO + Gate results in an improvement in TPR of around 6.7%.



(a) Standard

(b) BTEO +Gate

Figure 4: True positive rates (± standard deviation) over the Moji datasets broken down by author demographics and sentiment labels. Results are averaged over 5 random runs with different random seeds.

Similar trends are observed over the Bios dataset: debiasing methods generally improve minority group performance. However, we also noticed that the trends for different occupations can be different. For example, the TPR performance for both gender groups improves for *architect* and *paralegal*, but decreases for *professor* and *accountant*. We hypothesise that this is due to the target class skew in the Bios datasets, and leave further investigation of this effect to future work.

5.4 Balancing toward anti-stereotyping

As shown in Table 2, even with DS or RW balancing, the model still shows biases in its predictions. We conduct preliminary experiments on Moji with RW and DS, while controlling for stereotyping skew in training using values for 0.8 to 0.2. In standard rebalancing we use as target 0.5, which describes a balanced situation. A larger skew > 0.5 will amplifying stereotyping, and < 0.5 describes a different type of stereotyping operating in the opposite direction. Balancing towards a 0.4 training skew leads to the best test results, with an accuracy of 71.7% and GAP of 11.8% for DS, and accuracy of 74.5% and GAP of 11.3% for RW. Comparing to the corresponding values in Table 2 (rows Balance DS and RW, for Moji), both results show a substantial reduction in GAP.

This idea is related to existing reweighting approaches in long-tail learning. For example, Cui et al. (2019) infer the effective number of samples which group each instance with its neighbours within a small region instead of using all data points, and reweight the loss of each class inversely proportional to the effective number of samples. We leave this further exploration of this line of research to future work.

We also experiment with GATE +RW and GATE +DS with a 0.4 training skew, however, the gated model does not show the same behaviour, as it just amplifies the training biases. This implies that, for the gated model, balanced training can help remove spurious correlations between protected attributes and main task labels, which is similar in nature to the effects of adversarial training.

6 Related Work

Fairness Much work on algorithmic fairness has focused on group fairness, i.e., disparities in error rates across groups defined by protected attributes, such as gender, age, or race. Many criteria have been proposed for group fairness, such as statistical parity (Dwork et al., 2012) and equal opportunity (Hardt et al., 2016). Broadly speaking, fairness can be classified into three categories: independence, separation, and sufficiency (Barocas et al., 2019), with the most recent work addressing separation criteria, i.e, potential correlations between main task labels and protected attributes.

Mitigating bias Many approaches for bias mitigation haven been proposed recently, including removing protected information form hidden representations (Li et al., 2018a; Ravfogel et al., 2020; Han et al., 2021b), preprocessing data to remove bias (Zhao et al., 2018; Vanmassenhove et al., 2018; Saunders and Byrne, 2020), modifying the training algorithm (Badjatiya et al., 2019), and post-hoc correction (Hardt et al., 2016).

In the context of NLP, the best results have been achieved through protected information removal. Iterative nullspace projection (Ravfogel et al. (2020)) takes hidden representations and projects them onto the nullspace of the weights of a linear classifier for each protected attribute. The classifier training and projection are carried out over multiple iterations to more comprehensively remove protected information.

Another popular approach is adversarial training, which jointly optimizes the removal of sensitive information and main task performance, through the incorporation of adversarial discriminator(s) to identify protected attributes from the hidden representations (Li et al., 2018a; Elazar and Goldberg, 2018; Wang et al., 2019). Differentiated adversarial learning (Han et al. (2021b)) uses an ensemble of adversaries for each protected attribute, subject to an orthogonality constraint.

Comparison with mixture of experts One line of work that is similar to our gated model is mixture of experts (Ma et al., 2018; Fedus et al., 2021). Technically, the gated model is similar to the MoE model in the sense that an expert can be largely aligned with a group-specific encoder in our model. However, there are several key differences: (1) instead of making independent predictions by each expert, our group-invariant encoder acknowledges the shared patterns across demographic groups; (2) MoE employs an extra softmax gating network to mix experts' predictions, while our method does an argmax based on group labels; and (3) we use the group-specific information jointly together with the group-invariant encoder' outputs for making the final predictions while the MoE model has one output layer for each expert.

7 Conclusions and Future Work

This paper proposed the adoption of balanced training approaches to mitigate bias, and demonstrated their effectiveness relative to existing methods, as well as their ability to further enhance existing methods. We also proposed a gated model based on demographic attributes as an input, and showed that while the simple version was highly biased, with a simple Bayesian extension at inference time, the method was highly effective at mitigating bias.

For future work, it is important to consider settings where there are multiple protected attributes, such as author age, gender, and ethnicity. A simple extension would be to treat G as being *intersectional classes*, defined as the Cartesian product of the multiple demographic groups. E.g., k binary groups would result in 2^k intersectional classes.

Acknowledgements

We thank the anonymous reviewers for their helpful feedback and suggestions. This work was funded by the Australian Research Council, Discovery grant DP200102519. This research was undertaken using the LIEF HPC-GPGPU Facility hosted at the University of Melbourne. This Facility was established with the assistance of LIEF Grant LE170100200.

Ethical Considerations

This work aims to advance research on bias mitigation in NLP. Although our proposed method requires access to training datasets with protected attributes, this is the same data assumption that is made by other related work such as adversarial training and INLP, and our target is to make fair predictions at inference time. One limitation of methods in this area, including ours, is the difficulty of training and evaluating when we don't have access to demographic attributes. To avoid user harm, we only use attributes which users have self identified in our experiments. Moreover, our proposed method is able to make fairer predictions either with or without demographic information. All data in this study is publicly available and used under strict ethical guidelines.

Limitations

(1) We have only investigated bias mitigation over English datasets. While nothing in the proposed methods is language-specific, it would be valuable to validate the methods over datasets for other languages from different language families.

(2) Consistent with previous work, we conduct experiments over MOJI using ethnicity labels, and BIOS using binary gender labels. We acknowledge that there are potentially subtle interactions between protected attributes, and the possibility that debiasing with respect to one protected attribute could negatively impact on a second (unannotated) protected attribute. To investigate this effect, we would need to experiment with datasets with multiple protected attributes (and debias with respect to certain attributes while measuring the impact on bias with respect to held-out attributes).

(3) As discussed in Sections 3 and 4.6, GATE models assume that the demographic attribute is available at prediction time to make fairer predictions. However, we also proposed a method using a non-informative prior in Section 4.7 to remove this requirement.

(4) For both INLP and DADV, we follow experimental setup from the original papers. However, the `fairlib` (Han et al., 2022) — which is presented after this work was done — recently show that both methods can obtain better results with a larger budget for hyperparameter fine-tuning. Based on our most recent experimental results, our proposed method BTEO itself is still competitive with other debasing methods, and BTEO + GATE still achieves better performance–fairness tradeoffs than the better-tuned INLP and DADV.

References

Pinkesh Badjatiya, Manish Gupta, and Vasudeva Varma. 2019. Stereotypical bias removal for hate speech detection task using knowledge-based generalizations. In *The World Wide Web Conference*, pages 49–59.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2019. *Fairness and Machine Learning*. `http://www.fairmlbook.org`.

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.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357.

Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. 2016. Domain separation networks. In *Advances in Neural Information Processing Systems*, volume 29.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Conference on Computer Vision and Pattern Recognition*.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 120–128.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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.

Mark Díaz, Isaac Johnson, Amanda Lazar, Anne Marie Piper, and Darren Gergle. 2018. Addressing age-related bias in sentiment analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–14.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 214–226.

Yanai Elazar and Yoav Goldberg. 2018. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 11–21, Brussels, Belgium. Association for Computational Linguistics.

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *arXiv preprint arXiv:2101.03961*.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021a. Decoupling adversarial training for fair NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 471–477.

Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021b. Diverse adversaries for mitigating bias in training. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2760–2765.

Xudong Han, Aili Shen, Yitong Li, Lea Frermann, Timothy Baldwin, and Trevor Cohn. 2022. fairlib: A unified framework for assessing and improving classification fairness. *arXiv preprint arXiv:2205.01876*.

Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 29:3315–3323.

Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. 2018. Women also snowboard: Overcoming bias in captioning models. In *Proceedings of the European Conference on Computer Vision*, pages 793–811.

Michael Höfler, Hildegard Pfister, Roselind Lieb, and Hans-Ulrich Wittchen. 2005. The use of weights to account for non-response and drop-out. *Social Psychiatry and Psychiatric Epidemiology*, 40(4):291–299.

Dirk Hovy. 2015. Demographic factors improve classification performance. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 752–762.

Dirk Hovy and Diyi Yang. 2021. The importance of modeling social factors of language: Theory and practice. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 588–602, Online. Association for Computational Linguistics.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. 2020. Decoupling representation and classifier for long-tailed recognition. In *Eighth International Conference on Learning Representations (ICLR)*.

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. 2020. Fairness without demographics through adversarially reweighted learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 728–740.

Yitong Li, Timothy Baldwin, and Trevor Cohn. 2018a. Towards robust and privacy-preserving text representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 25–30.

Yitong Li, Timothy Baldwin, and Trevor Cohn. 2018b. What's in a domain? learning domain-robust text representations using adversarial training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 474–479, New Orleans, Louisiana. Association for Computational Linguistics.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. 2018. Gender bias in neural natural language processing. *arXiv preprint arXiv:1807.11714*.

Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1930–1939.

R Timothy Marler and Jasbir S Arora. 2004. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6):369–395.

Ji Ho Park, Jamin Shin, and Pascale Fung. 2018. Reducing gender bias in abusive language detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2799–2804.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256.

M Ye Salukvadze. 1971. Concerning optimization of vector functionals. i. programming of optimal trajectories. *Avtomat. i Telemekh*, 8:5–15.

Danielle Saunders and Bill Byrne. 2020. Reducing gender bias in neural machine translation as a domain adaptation problem. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7724–7736.

Shivashankar Subramanian, Xudong Han, Timothy Baldwin, Trevor Cohn, and Lea Frermann. 2021. Evaluating debiasing techniques for intersectional biases. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Eva Vanmassenhove, Christian Hardmeier, and Andy Way. 2018. Getting gender right in neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3003–3008.

Thomas L Vincent. 1983. Game theory as a design tool.

Thomas L Vincent and Walter Jervis Grantham. 1981. *Optimality in parametric systems*. Wiley.

Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. 2019. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5310–5319.

Han Zhao, Amanda Coston, Tameem Adel, and Geoffrey J Gordon. 2019. Conditional learning of fair representations. In *International Conference on Learning Representations*.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20.

A Dataset distribution

A.1 MOJI

This training dataset has been artificially balanced according to demographic and task labels, but artificially skewed in terms of race–sentiment combinations, as follows: AAE–happy = 40%, SAE–happy = 10%, AAE–sad = 10%, and SAE–sad = 40%. We used the train, dev, and test splits from Han et al. (2021b) of 100k/8k/8k instances, respectively.

A.2 BIOS

Since the data is not directly available, in order to construct the dataset, we re-scrape the data with the scripts of Ravfogel et al. (2020), leading to a dataset with 396k biographies, which we randomly split into train (65%), dev (10%), and test (25%).

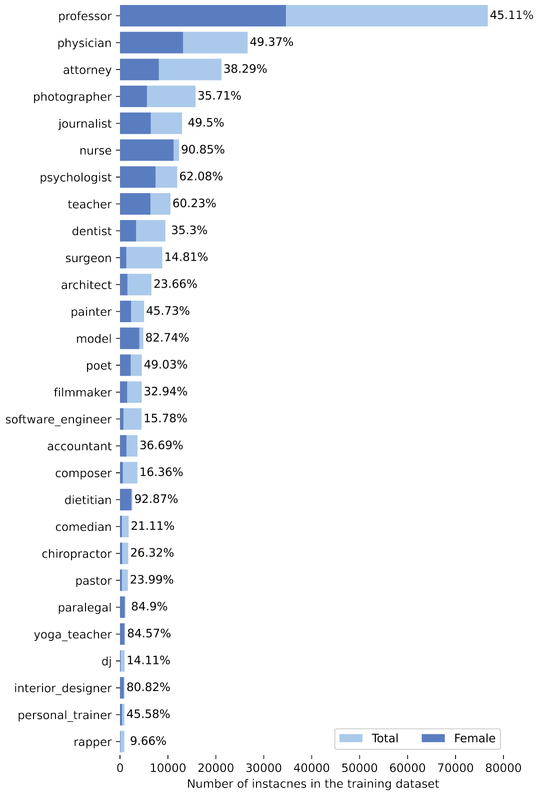


Figure 5: Bios dataset statistics.

Figure 5 shows the statistic of the BIOS dataset. Each row corresponds to a profession, including the total number of instances and number of female instances. Besides, each profession is also annotated with the percentage of female instances. There are slight discrepancies in the dataset composition due to data attrition: the original dataset (De-Arteaga et al., 2019) had 399k instances, while 393k were collected by Ravfogel et al. (2020).

B Reproducibility

B.1 Models

For INLP (Ravfogel et al., 2020), we take the fixed STANDARD model for the given dataset, and iteratively train a linear classifier and perform nullspace projection over the learned representation. For the other baseline models — ADV and DADV— we jointly train the adversarial discriminators and classifier. In order to ensure a fair comparison, we follow Han et al. (2021a) in using a model consisting of the same fixed-parameter encoder as ours followed by a trainable 3-layer MLP.

B.2 Hyperparameter Tuning

All approaches proposed in this paper share the same hyperparameters as the standard model. Hyperparameters are tuned using grid-search, in order to maximise accuracy for the standard model, and to minimise the fairness GAP for debiasing methods, subject to the accuracy exceeding a given threshold. The accuracy threshold is chosen to ensure the selected model achieves comparable performance to baseline methods, defined as up to 2% less than best baseline accuracy. Taking RW as an example, the best baseline accuracy on the BIOS development dataset is 75.7% and accordingly the (development) accuracy threshold is set to 73.7%; among models in the hyperparameter search space that exceed this threshold, we take the model with minimum GAP. We report test results for the selected models.

In terms of the baseline models, both DADV and INLP have additional hyperparameters: for DADV these are the weight of the adversarial loss, which controls the performance–fairness trade-off; the number of sub-adversaries; and the weight of the difference loss, to better remove demographic information; while INLP also has a trade-off hyperparameter, the number of null-space projection iterations, and other hyperparameters related to linear attackers and classifiers.

The trade-off hyperparameter makes such models more flexible in performing model selection. However, it also requires manual selection for better trade-offs, and different strategies have been introduced. For example, INLP manually selects the model at a iteration where the accuracy is minimally damaged while the fairness improves greatly. Similar manual selection for better trade-offs is also required for ADV and DADV, but the strategies proposed in the original papers are slightly different to one another, and are also task-specific.

In order to reproduce previous methods, we follow the original paper in setting the accuracy threshold, and then tuning hyperparameters for the best fairness.

For the ADV and DADV models, following the work of Han et al. (2021b), we tune extra hyperparameters separately, such as the trade-off hyperparameter, while using the same shared hyperparameters to the selected base models. Similarly, the number of iterations for the INLP model is tuned once other hyperparameters have been fixed.

B.3 Training Details

We conduct all our experiments on a Windows server with a 16-core CPU (AMD Ryzen Threadripper PRO 3955WX), two NVIDIA GeForce RTX 3090s with NVLink, and 256GB RAM.

B.3.1 MOJI

For all baseline models, we follow the method of Han et al. (2021b). Specifically, we train the STANDARD model for 100 epochs with the Adam optimizer (Kingma and Ba, 2015), learning rate of 3×10^{-5}, and batch size of 1024. For ADV, the main model is jointly trained together with adversaries which are implemented as 3-layer MLP, and the weight of adversarial loss is 0.8. For each iteration (epoch) of the main model, an adversary is trained for 60 epochs, keeping the checkpoint model that performs best on the dev set. Three sub-adversaries are employed by the DADV, with the difference losss weight of $10^{3.7}$. For INLP, logistic regression models are used for both identifying null-space to the demographic information at each iteration, and making the final predictions given debiased hidden representations. Since the number of iterations in INLP is highly affected by the random seed at each run, we re-select it at each iteration.

As for our models, the DS model is trained with the learning rate of 10^{-5} and batch size of 512; the RW is trained with the learning rate of 10^{-4} and batch size of 1024; and the GATE is trained with the the set of hyperparameters to the base model.

B.3.2 BIOS

Models are trained with similar hyperparameters as models on the MOJI dataset. We thus only report main differences for each of them: the STANDARD model is trained with the batch size of 512 and learning rate of 3×10^{3}; DS models are trained with the batch size of 128 and learning rate of 10^{-3}, and RW models are trained with the batch of 256 and learning rate of 3×10^{-5}.

We train the ADV model with the adversarial loss weight of $10^{-2.3}$, learning rare for adversarial training of 10^{-1}, learning rate of 10^{-3}, and batch size of 128. The DADV is trained with same setting as the ADV, excepting the difference loss weight of 10^{2}. For details of the assignment of other hyperparameters and hyperparameter searching space, refer to Supplementary Materials.

C The calculation of DTO

For ease of comparison between approaches, we introduce 'distance to the optimum' (DTO), a single metric to incorporate accuracy and GAP into a single figure of merit, which is calculated by: (1) converting GAP to $1 - \text{GAP}$ (denoted as fairness; higher is better); (2) normalizing each of accuracy and fairness, by dividing by the best result for the given dataset (i.e., highest accuracy and fairness); and (3) calculating the Euclidean distance to the point $(1, 1)$, which represents the hypothetical system which achieves highest accuracy and fairness for the dataset. Lower is better for this statistic, with minimum 0.

We calculate DTO based on all results shown in Table 7. Taking the DAdv model on the Moji dataset for example, the trade-off is calculated as follows:

1. Find the best accuracy and fairness (1-GAP) separately; i.e., 74.9 (GATE + RW) and 92.9 (GATE $^{\text{soft}}_{\text{RMS}}$), resp.
2. Normalize the accuracy and fairness metric of DADV, resulting in $0.995 = \frac{74.5}{74.9}$ and $0.877 = \frac{81.5}{92.9}$.
3. Calculate the Euclidean distance between $(1, 1)$ and $(0.995, 0.877)$, giving 0.123.

D Training time estimation

Given that the training time is affected by factors such as batch size, hidden size, and learning rate, to perform a fair comparison between different models, we estimate the training time of a model based on hyperparameter tuning results, over a shared search space of base hyperparameters (i.e., the hyperparameters related to the standard model), with any other approach-specific hyperparameters fixed.

		Best assignment						
Hyperparameter	**Search space**	STANDARD	ADV	DADV	DS	RW	DADV + DS	DADV + RW
number of epochs	-	100						
patience	-	10						
encoder	-	DeepMoji (Felbo et al., 2017)						
embedding size	-	2304						
hidden size	-	300						
number of hidden layers	*choice-integer*[1, 3]	2						
batch size	*loguniform-integer*[64, 2048]	1024	1024	1024	512	1024	512	1024
output dropout	*uniform-float*[0, 0.5]	0.4	0.4	0.4	0.5	0.5	0.2	0.1
optimizer	-	Adam (Kingma and Ba, 2015)						
learning rate	*loguniform-float*[10^{-6}, 10^{-1}]	3×10^{-5}	3×10^{-5}	3×10^{-5}	10^{-5}	10^{-4}	3×10^{-5}	3×10^{-4}
learning **r**ate **s**cheduler	-	reduce on plateau						
LRS patience	-	2 epochs						
LRS reduction factor	-	0.5						
ADV loss weight	*loguniform-float*[10^{-4}, 10^{2}]	-	$10^{-0.1}$	$10^{-0.1}$	-	-	$10^{0.2}$	$10^{0.0}$
ADV hidden size	*loguniform-integer*[64, 1024]	-	256	256	-	-	256	256
number of adversaries	*choice-integer*[1, 8]	-	1	3	-	-	3	3
DADV loss weight	*loguniform-float*[10^{-5}, 10^{5}]	-	-	$10^{3.7}$	-	-	10^{2}	$10^{2.6}$

Table 5: Search space and best assignments on the MOJI dataset

		Best assignment						
Hyperparameter	**Search space**	STANDARD	ADV	DADV	DS	RW	DADV + DS	DADV + RW
number of epochs	-	100						
patience	-	10						
encoder	-	uncased BERT-base (Devlin et al., 2019)						
embedding size	-	768						
embedding type	*choice*{**‘CLS’**, **‘AVG’**}	**‘AVG’**						
hidden size	-	300						
number of hidden layers	*choice-integer*[1, 3]	2						
batch size	*loguniform-integer*[64, 2048]	512	128	128	128	256	256	512
output dropout	*uniform-float*[0, 0.5]	0.5	0.3	0.2	0.3	0.5	0.2	0.4
optimizer	-	Adam (Kingma and Ba, 2015)						
learning rate	*loguniform-float*[10^{-6}, 10^{-1}]	3×10^{-3}	10^{-3}	10^{-3}	10^{-3}	3×10^{-5}	3×10^{-3}	3×10^{-4}
learning **r**ate **s**cheduler	-	reduce on plateau						
LRS patience	-	2 epochs						
LRS reduction factor	-	0.5						
ADV loss weight	*loguniform-float*[10^{-8}, 10^{2}]	-	$10^{-2.3}$	$10^{-2.3}$	-	-	$10^{-2.8}$	10^{-5}
ADV hidden size	*loguniform-integer*[64, 1024]	-	256	256	-	-	256	256
number of adversaries	*choice-integer*[1, 8]	-	1	3	-	-	3	3
DADV loss weight	*loguniform-float*[10^{-5}, 10^{5}]	-	-	10^{2}	-	-	10^{3}	$10^{3.3}$

Table 6: Search space and best assignments on the BIOS dataset

E Mapping of previous objectives

As for the jointly balance, Lahoti et al. (2020) state that “In addition to vanilla inverse probability weighting (IPW), we also report results for an IPW variant with inverse probabilities computed jointly over protected-features and class-label reported as IPW(S+Y)”. It clearly shows that instance from group g with class y is weighted by $\frac{1}{n_{y,g}}$. Adding this weight to the unweighted loss function leads to $\sum_y \sum_g \frac{n_{y,g}}{n} \frac{1}{n_{y,g}} \mathcal{L}_{y,g}$, which exactly the objective that is shown in our paper.

In terms of the conditionally balance, in the section ‘Alternative Data Splits’ of Wang et al. (2019),

		Moji				Bios			
Method	**Model**	**Accuracy↑**	**GAP ↓**	**DTO ↓**	**Time↓**	**Accuracy↑**	**GAP ↓**	**DTO ↓**	**Time↓**
Baselines	Standard	71.6 ± 0.1	31.0 ± 0.3	0.261	1.0	82.3 ± 0.0	16.0 ± 0.5	0.110	1.0
	INLP	68.5 ± 1.1	33.8 ± 3.9	0.300	14.0	70.5 ± 0.5	6.7 ± 0.9	0.145	6.3
	Adv	74.3 ± 0.4	22.2 ± 3.7	0.163	36.1	81.1 ± 0.1	12.7 ± 0.3	0.077	1.3
	DAdv	74.5 ± 0.3	18.5 ± 2.0	**0.123**	109.4	81.1 ± 0.1	12.6 ± 0.3	**0.076**	2.4
Balance	DS	71.9 ± 0.1	23.2 ± 0.2	0.178	0.5	79.4 ± 0.1	9.7 ± 0.6	**0.057**	0.3
	RW	74.0 ± 0.2	21.5 ± 0.4	**0.155**	1.0	74.7 ± 0.3	7.4 ± 0.3	0.095	1.0
Gate	Gate	64.8 ± 0.1	65.2 ± 0.9	0.640	1.0	82.4 ± 0.1	19.2 ± 0.3	0.144	1.0
	Gate + DS	72.5 ± 0.0	16.3 ± 0.7	0.104	0.6	79.4 ± 0.1	9.2 ± 0.2	**0.053**	0.3
	Gate + RW	74.9 ± 0.2	13.8 ± 0.3	**0.072**	1.1	74.9 ± 0.2	7.1 ± 0.2	0.092	1.0
Bayesian	$\text{Gate}^{\text{soft}}_{0.5}$	72.7 ± 0.2	30.2 ± 0.3	0.250	1.0	80.8 ± 0.1	11.6 ± 0.3	0.066	1.0
	$\text{Gate}^{\text{soft}}_{\text{Acc}}$	74.8 ± 0.2	20.3 ± 0.3	0.142	1.0	81.1 ± 0.1	19.8 ± 0.4	0.151	1.0
	$\text{Gate}^{\text{soft}}_{\text{RMS}}$	73.5 ± 0.2	7.1 ± 0.3	**0.019**	1.0	80.5 ± 0.1	11.1 ± 0.3	**0.063**	1.0
Combination	DAdv + DS	72.2 ± 0.2	14.3 ± 0.2	**0.085**	72.1	79.3 ± 0.1	9.9 ± 0.2	**0.059**	2.3
	INLP + DS	72.1 ± 1.6	18.4 ± 3.1	0.127	6.3	73.2 ± 0.6	5.9 ± 0.8	0.112	1.3
	DAdv + RW	74.6 ± 0.1	18.9 ± 0.3	0.127	108.2	74.1 ± 0.2	7.2 ± 0.4	0.102	3.0
	INLP + RW	72.3 ± 1.9	15.7 ± 3.1	0.099	13.9	73.6 ± 0.6	5.6 ± 0.7	0.107	6.3

Table 7: Results over the sentiment analysis (Moji) and biography classification (Bios) tasks. Trade-offs are measured by the normalized Euclidean distance between each model and the ideal model, and lower is better. **Bold** = best trade-off within category. Training time is reported relative to Standard, which takes 35 secs and 16 mins for Moji and Bios, respectively.

they obtain the dataset by resampling such that the number of occurrences of men with label y and of women with label y is close, i.e., the size of different demographic groups are almost identical for each class-label. This is equivalent to assign equal weights to difference demographic groups within each class-label, i.e., $\sum_y \sum_g \frac{n_{y,g}}{n} \frac{n_{y,*}}{n_{y,g}} \mathcal{L}_{y,g}$.

Finally, for the balanced demographics, Zhao et al. (2018) balance the distribution of demographic groups which reweights instance of a demographic group inversely to its proportion, leading a weight $\frac{n}{n_{*,g}}$. As a result, the final objective is $\sum_y \sum_g \frac{n_{y,g}}{n} \frac{n}{n_{*,g}} \mathcal{L}_{y,g}$.

F Further extensions

F.1 Combining balanced training with benchmark methods

The baseline methods INLP and DAdv as presented above were used in a manner consistent with their original formulation, i.e., without balanced training. An important question is whether balanced training might also benefit these methods. It is trivial to combine downsampling with INLP and DAdv, as the method simply prunes the training dataset, but does not impact the training objective. To combine instance reweighting with DAdv, we modify the training objective such that the cross-entropy term is scaled by $\tilde{p}^{-1}$, while leaving the adversarial term unmodified, i.e., solve for $\min_M \max_A \sum_{(x_i,y_i,g_i)\in\mathcal{D}} \tilde{p}^{-1}\mathcal{X}(y_i, \hat{y}_i) - \lambda_{\text{adv}}\mathcal{X}(g, \hat{g})$. For INLP, we simply train a BTEO model, and then iteratively perform INLP linear model training and nullspace projection over the learned representations.

Results are presented in the final section of Table 7 ("Combination"), and indicate that the combined methods appreciably outperform both the standalone demographic removal methods and balanced training approaches, without extra training time cost. That is, demographic information removal and balanced training appear to be complementary.