Towards Fairness-aware Adversarial Network Pruning

Lei Zhang
Zhejiang University
zl.leizhang@zju.edu.cn

Zhibo Wang[‡]
Zhejiang University
zhibowang@zju.edu.cn

Xiaowei Dong
Wuhan University
xwdong@whu.edu.cn

Yunhe Feng
University of North Texas
yunhe.feng@unt.edu

Xiaoyi Pang
Wuhan University
xypang@whu.edu.cn

Zhifei Zhang
Adobe Research
zzhang@adobe.com

Kui Ren
Zhejiang University
kuiren@zju.edu.cn

Abstract

Network pruning aims to compress models while minimizing loss in accuracy. With the increasing focus on bias in AI systems, the bias inheriting or even magnification nature of traditional network pruning methods has raised a new perspective towards fairness-aware network pruning. Straightforward pruning plus debias methods and recent designs for monitoring disparities of demographic attributes during pruning have endeavored to enhance fairness in pruning. However, neither simple assembling of two tasks nor specifically designed pruning strategies could achieve the optimal trade-off among pruning ratio, accuracy, and fairness. This paper proposes an end-to-end learnable framework for fairness-aware network pruning, which optimizes both pruning and debias tasks jointly by adversarial training against those final evaluation metrics like accuracy for pruning, and disparate impact (DI) and equalized odds (DEO) for fairness. In other words, our fairness-aware adversarial pruning method would learn to prune without any handcraft rules. Therefore, our approach could flexibly adapt to variate network structures. Exhaustive experimentation demonstrates the generalization capacity of our approach, as well as superior performance on pruning and debias simultaneously. To highlight, the proposed method could preserve the SOTA pruning performance while significantly improving fairness by around 50% as compared to traditional pruning methods.

1. Introduction

With the massive growth of parameters in nowadays deep models, pruning techniques [10, 8, 7] have achieved appealing reductions in network memory footprint and time complexity. However, they tend to overlook the bias hidden behind high-accuracy predictions [2, 27, 24]. Thus, it is critical to improving fairness in network pruning for broad and reliable applications of AI systems. Intuitively, fairness could be considered as postprocessing after pruning, but it would come to a suboptimal solution since disparate objectives of these two tasks. Therefore, optimizing fairness during the pruning process would be a promising research direction, which motivates our work toward this track.

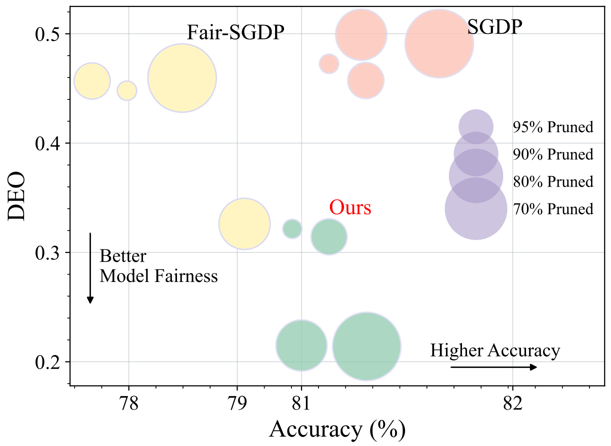


Figure 1: **Can existing pruning methods improve model fairness while maintaining accuracy?** We use the area of a circle to represent the size of a model after pruning, *i.e*., the smaller the area, the more compact the model. Towards the bottom-right corner of the figure, it represents a more accurate and fair model. Please note that the SOTA network pruning method SGDP [28] (pink) achieves high accuracy while lagging far behind in fairness on CelebA [21] dataset. Even with enhanced fairness via Adversarial Debiasing [42] before pruning, existing pruning methods (yellow) seriously degrade the fairness. Our method (green) significantly improves fairness and preserves a relatively high accuracy even with a high pruning ratio.

Various pruning techniques have been proposed to minimize degradation in accuracy after network pruning [3, 7,

[‡]Corresponding author.

37] while seldom focusing on the improvement in model fairness. We have demonstrated in Figure 1 that existing pruning methods do not consider fairness, *i.e*., compressed models via SOTA pruning approach [28] (green) achieve high accuracy but present strong biases against sensitive attributes. Even if we enhance the model fairness in advance, the compressed models (purple) still suffer from unfairness with significant accuracy degradation. Current pruning techniques tend to pursue high accuracy with a high pruning ratio but ignore inherent unfairness in deep models.

Method	PR.	ACC↑	DEO↓
Normal training	0%	81.63%	0.5229
FairGRAPE [19]	80%	80.04%	0.5155 (-0.0074)
Ours	80%	78.06%	**0.3390 (-0.1839)**

Method	PR.	ACC↑	DI↑
Normal training	0%	81.63%	0.3144
FairGRAPE [19]	80%	80.04%	0.2315 (-0.0829)
Ours	80%	79.96%	**0.4560 (+0.1416)**

Table 1: Comparison of our method and FairGRAPE on accuracy and demographic fairness. PR. denotes the number of parameters to be pruned.

Some recent works have started to explore fairness enhancement during pruning. Lin *et al*. [19] propose FairGRAPE to reduce the disparity in performance degradation on different sub-groups caused by pruning while contributing less to demographic fairness as shown in Tab. 1. Wu *et al*. [38] took pruning as a tool to improve model fairness. However, there is still space for improvement, since they targeted medical images that may require strong prior knowledge to design the method.

Above all, it is imperative to propose an effective to ameliorate fairness and preserve the accuracy and efficiency of network pruning. Since it is difficult to train a small subnetwork from scratch to achieve the same performance as its dense counterpart [34, 5, 22], the practical solution is to reduce a large-scale network with redundant and biased parameters to a compact and unbiased sub-network. The main challenge lies in searching for biased and redundant connections and improving model fairness while not hurting the accuracy of the pruned model.

In this paper, we propose the fairness-aware pruning technique to improve fairness and preserve the accuracy and efficiency of compressed models. To achieve this, we guide the pruning process to decide which connections to prune in terms of parameter redundancy and model bias. The key idea is to formulate the pruning step as adversarial learning between fairness and performance. Specifically, we design a discriminator to distinguish predictions from one sensitive group against others. During the training process, the discriminator is trained to remove the correlation between prediction and sensitive attribute while the pruning step is to train the sub-network to deceive the discriminator, thus accomplishing fairness-aware pruning in one shot. Exhaustive experimental evaluation demonstrates that our compressed networks simultaneously ameliorate fairness and maintain comparable accuracy and efficiency.

Recently, Ramanujan *et al*. [28] found the existence of hidden sub-networks with high benign accuracy within randomly initialized networks and Sehwag *et al*. [33] and Fu *et al*. [6] extend the finding to sub-networks with robust accuracy. Using our pruning technique, we further extend the finding to model fairness, where we uncover fair subnetworks within randomly initialized networks without any model training. This indicates that searching for the locations of a subset of weights within a randomly initialized network might be potentially as effective as adversarially debiasing weight values in comparable model sizes, which opens up a new respective for understanding model fairness.

In summary, the main contributions are in three-folds:

- We propose the fairness-aware pruning technique, which designs a discriminator to distinguish the correlation between predictions and fairness-related attributes. The pruning step is trained adversarially with a discriminator. This design effectively ameliorates fairness, achieves efficiency, and preserves accuracy on par with comparable-sized models.
- Exhaustive experiment validates the superior performance of our method. The compressed networks outperform the state-of-art pruning techniques on fairness and achieve comparable performance on accuracy.
- The proposed method can effectively search a fair subnetwork from a randomly initialized network without any training. This finding would open up a new perspective on model fairness.

2. Related Work

2.1. Model Compression

Model compression aims to reduce the parameters of networks while maintaining comparable model performance. Popular directions for network compression involve network pruning [10, 5, 22, 30], parameter quantization [9], knowledge distillation [13], and neural architecture search [44]. In our paper, we focus on network pruning, which eliminates redundant connections without assumptions about the structures of weights.

Existing research in network pruning mainly focuses on minimizing the degradation of performance after pruning and designing efficient pruning algorithms: how to extract informative connections [17, 20, 4, 25, 18], how to maintain the structure of the original model [16, 36], and when to conduct pruning during the training process [35, 39]. However, existing pruning methods do not account for the fairness of compressed models.

2.2. Unfairness Mitigation

Existing research on model unfairness mitigation can be divided into three categories according to targeting stages: pre-processing, in-processing, and post-processing. Pre-processing methods [26, 29, 43] mitigate biases in the training dataset before training. Prior works utilize representation transformation and distribution augmentation to mitigate bias in training sets. In-processing [41, 1, 42, 31, 32] methods design fairness-aware training algorithm, *i.e.*, introducing fairness penalty or employing adversarial strategies. Post-processing [23, 15] methods adjust model predictions after training according to certain fairness criteria. Existing works on post-processing cover a wide variety including replacing the biased classifier with a pre-trained fair classifier and manipulating model predictions based on group fairness criteria. Current works put more emphasis on full-size pre-trained models. However, few works focus on the fairness of pruned models. In this paper, we borrow the ideas from network pruning and unfairness mitigation and tend to achieve fairness-aware network pruning.

3. Preliminary

3.1. Network Pruning

Existing pruning methods perform various compression pipelines. One such highly successful approach is a three-step compression pipeline [10, 8]. It involves pre-training a network, pruning it, and then fine-tuning it. In the pruning step, a binary mask is obtained, which determines which connections to be pruned. In the fine-tuning step, only the non-pruned connections are updated in order to maintain the model performance. We refer to the network obtained after fine-tuning as the compressed network. Note that both pruning and fine-tuning steps can be alternatively repeated to perform multi-step pruning [10].

3.2. Model Fairness

In this paper, we mainly focus on visual classification models because of extensive academic efforts on them, as well as their broad industrial applications. Moreover, it is imperative to achieve equal treatment for people with different protected attributes, *e.g.*, nationality, gender, and ethnicity. Usually, disparate impact [40] and equalized odds [11] are used to measure model fairness.

In a binary classification task, *e.g.*, facial attribute classification, suppose target label $y \in \mathcal{Y} = \{-1, 1\}$, and sensitive attribute $z \in \mathcal{Z} = \{-1, 1\}$. $y = 1$ is set as favourable class (*e.g.*, attractive) and $z = 1$ is set as privileged group (*e.g.*, Blond Hair).

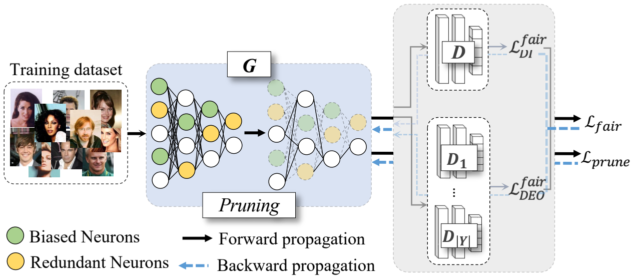


Figure 2: Overview of the proposed method, which consists of two learnable components, *i.e.*, a generator for network pruning, and discriminators for DI and DEO respectively for distinguishing correlation between sensitive attribute and model prediction.

Definition 1 (Disparate Impact) If the value of z does not influence assigning a sample to the positive class, *i.e.*, model prediction $\hat{y} = 1 \perp z$, then the classifier satisfies demographic parity:

$$P(\hat{y} = 1 \mid z = -1) = P(\hat{y} = 1 \mid z = 1). \tag{1}$$

Disparate impact aims for the same positive prediction ratio for each sensitive attribute and ensures statistical parity of models.

Definition 2 (Equalized Odds) If the value of z can not influence the positive outcome for samples given y, i.e. $\hat{y} = 1 \perp z|y$, then the classifier satisfies equalized odds:

$$P(\hat{y} = 1|y, z = -1) = P(\hat{y} = 1|y, z = 1), y = \{-1, 1\}. \tag{2}$$

Equalized odds mean that positive output is statistically independent of the sensitive attribute given the target label. Samples in both the privileged and unprivileged groups have the same false positive rate and false negative rate. Equalized odds ensure the predictive parity of models.

4. Method

In this section, we will first describe the overview of our method, and then detail the design of the proposed method. Finally, we will further discuss the training strategy and details of our method.

4.1. Overview

The metric of the existing pruning methods puts more emphasis on prohibiting the degrading of accuracy instead of explicitly taking fairness into consideration. With that, biased neurons can still play negative roles in the compressed networks. Therefore, the unfairness in the compressed network could be largely due to the remaining biased connections after network pruning. Based on these analyses, in order to improve fairness while maintaining performance and efficiency, the key of our method is to appropriately prune both biased and redundant connections between model neurons. Specifically, we leverage the adversarial scheme on model pruning to construct a two-player game against model fairness and accuracy, which help us to locate and prune the redundant and biased connections to achieve substantial improvement in model fairness and sacrifice performance as little as possible.

The pipeline of the proposed method is shown in Figure 2, consisting of two learnable components: 1) the generator that prunes the network based on both performance and fairness, and 2) the discriminator that distinguishes the correlation between model predictions and the sensitive attribute. The generator is assumed to be a compressed network that is trained against the discriminator. Sharing the spirit of adversarial training, the discriminator is trained to distinguish correlations between sensitive attributes and model prediction, while the generator learns to fail the discriminator, thus guiding the pruning to search biased and redundant connections between neurons.

4.2. Fairness-aware Adversarial Pruning

In this part, we detail the loss functions of the method and the design of the discriminator mentioned above. As illustrated in Fig. 2, we assume the generator to be a compressed classification model f. Given an input x, whose target label is y, the predicted label $\hat{y} = f(x)$. The discriminator D is applied on predicted label $\hat{y}$ to distinguish a certain sensitive attribute z, and the generator G guides the pruned network to make fair and accurate predictions $\hat{y}$ based on input x.

Loss function of D: Intuitively, with a compressed model, the unfairness is mainly caused by the accuracy-prioritized pruning metric which still remains the strong correlation between predicted labels and the sensitive attribute after network pruning. Thus, the compressed network tends to become biased to the sensitive attribute as the original. Based on the above analysis, we first need to train the discriminator D to distinguish predictions $\hat{y}$ from one sensitive group z from others. With a well-trained discriminator D, the biased correlation between predicted label $\hat{y}$ and sensitive attribute z could be weakened. The detailed design of discriminator D is closely related to the fairness metrics considered. Here we consider two fairness constraints, *i.e.*, DI (Disparate Impact), and DEO (Equalized Odds), which are widely adopted in the classification model. We design functions $\mathcal{L}_{fair}$ for Discriminator D, which includes $\mathcal{L}_{DI}^{fair}$ and $\mathcal{L}_{DEO}^{fair}$ to for fairness metrics DI and DEO respectively.

For the fairness metric DI, the sensitive attribute z should be independent of the predicted label $\hat{y}$. Therefore, the discriminator D should ensure the equal probability of the predicted label $\hat{y}$ given the sensitive attribute z. Therefore, the discriminator loss for DI can be expressed as:

$$\mathcal{L}_{DI}^{fair} = \sum_{z \in \mathcal{Z}} \sum_{i: z^{(i)} = z} \frac{1}{m} \log D\left(\hat{y}^{(i)}\right), \tag{3}$$

where Z denotes the set of sensitive attributes and m denotes the number of samples.

For the fairness metric DEO, the positive output of predicted label $\hat{y}$ should be independent of the sensitive attribute z given the target label y. The discriminator D guarantees the equal probability of predicted label $\hat{y}$ given the sensitive attribute z based on the condition of target label y. Therefore, the discriminator loss for DEO can be expressed as below:

$$\mathcal{L}_{DEO}^{fair} = \sum_{y \in \mathcal{Y}} \sum_{z \in \mathcal{Z}} \sum_{i: \left(y^{(i)}, z^{(i)}\right) = (y, z)} \frac{1}{m} \log D_{z|y}\left(\hat{y}^{(i)}\right), \tag{4}$$

where $\mathcal{Z}, \mathcal{Y}$ denotes the set of sensitive attributes and target labels respectively and m represents the number of samples. Note that discriminator here $D_{z|y}$ can be further denoted as $D_{z|y=1}(\cdot), \ldots, D_{z|y=|\mathcal{Y}|}(\cdot)$ which is conditioned on target label y.

Overall, loss function of D can be formulated as below:

$$\mathcal{L}_D = \alpha \times \mathcal{L}_{DI}^{fair} + (1 - \alpha) \times \mathcal{L}_{DEO}^{fair}, \tag{5}$$

where $\alpha \in [0, 1]$. The α is set to 1 for fairness metric DI and the α is set to 0 for fairness metric DEO, respectively.

Loss function of G: By contrast, the generator G aims to fail D, and an intuitive solution is to maximize the loss of discriminator D. The training of generator G is the network pruning process in our method. Through the pruning step, we obtain a binary mask $\hat{m}$, which determines which connections are fairness-related and over-parameterized. We achieve this by integrating the fairness training objective in the pruning strategy itself by formulating it as the following:

$$\mathcal{L}_G = \mathcal{L}_{prune}\left(f_{\theta \odot \hat{m}}(x), y\right) - \mathcal{L}_D, \tag{6}$$

where $\theta \odot \hat{m}$ refers to the element-wise multiplication of mask $\hat{m}$ with weight parameter θ of network f, $\mathcal{L}_{fair}$ denotes one of the loss functions $\mathcal{L}_{DI}^{fair}$ and $\mathcal{L}_{DEO}^{fair}$, and $\mathcal{L}_{prune}$ is classification loss, *e.g.*, cross-entropy loss. The learning of binary pruning mask $\hat{m}$ is formulated as:

$$\hat{m} = \underset{\hat{m} \in \{0,1\}^N}{\operatorname{argmin}} \mathcal{L}_G \quad s.t. \; \|\hat{m}\|_0 \leq k, \tag{7}$$

where N is the total number of network weights and k is the number of remaining weights. The predefined pruning ratio of the network can be written as $\left(1 - \frac{k}{N}\right)$.

4.3. Training Algorithm

Based on the Eq. 3, Eq. 4, and Eq. 7, the generator and discriminator are optimized alternatively. The generator G

Algorithm 1: Fairness-aware network pruning

```
Input: Training data $\mathcal{D}$, Number of remaining weights $k$, Maximum iteration $T$, Discriminator $D$, Pre-trained neural network $f$ parameterized $\theta$, Loss objective: $L_{prune}$, $L_{fair}$(i.e. $L_{DI}^{fair}$, $L_{DEO}^{fair}$), Binary pruning mask $\hat{m}$, Hyper-parameter $\alpha$.
Output: Compressed network $\theta_{compress}$
Initialize discriminator $D$ and pruning mask $\hat{m}$.
for $t \leftarrow 0$ to $T$ do
    Sample a batch of $n$ inputs $x$, labels $y$ and attributes $z$ from $\mathcal{D}$
    Calculate discriminator loss $\mathcal{L}_D$:
        $\mathcal{L}_D = \alpha \times \mathcal{L}_{fair}^{DI} + (1-\alpha) \times \mathcal{L}_{fair}^{DEO}$
    Update discriminator $D$:
        $D \leftarrow D - \eta_D \nabla_D \mathcal{L}_D$
    Calculate pruning loss $\mathcal{L}_G$:
        $\mathcal{L}_G = \mathcal{L}_{prune} - \alpha \times \mathcal{L}_{fair}^{DI} - (1-\alpha) \times \mathcal{L}_{fair}^{DEO}$
    Get binary pruning mask $\hat{m}$:
        $\hat{m} = \underset{\hat{m}}{argmin}\, \mathcal{L}_G \quad s.t.\ \|\hat{m}\|_0 \leq k$
    Prune network $f$ via pruning mask $\hat{m}$:
        $f = f_{\theta \odot \hat{m}}$
end
```

plays a min-max game with D where D maximizes the ability to predict a correlation between predicted label $\hat{y}$ and sensitive attribute z while G tries to minimize its ability. At the same time, G tries to let the compressed network f still recognize the right target label.

5. Experiment

In this section, we first describe our experimental setup. Then, we evaluate the proposed method on different datasets. Finally, we investigate the ablation effect of network architecture and pruning ratio.

5.1. Experimental Setup

Datasets. We adopt two face datasets in our evaluation, *i.e.*, CelebA [21] and LFW [14], which carry those commonly protected attributes like gender. In CelebA, we take gender as the protected attribute to measure the fairness of model prediction for target labels and choose *Attractive* and *Blond_Hair* as target labels. In LFW, we choose *Smiling* and *Young* as target labels and gender as the sensitive attribute.

Evaluation metrics. For fairness evaluation, we use the difference in disparate impact (DI) and the difference in equalized odds (DEO) to evaluate model fairness. Meanwhile, the accuracy (ACC) of predicting target labels will also be reported. We demonstrate the effectiveness of the proposed method by comparing models pruned by our method with full-sized models normally trained on DI, DEO, and accuracy. A higher DI and a lower DEO indicate that the samples in the privileged group are treated equally as those in the unprivileged group.

Baselines. We compare our method with three classes of baselines. The first class is training with the full-size network, *i.e.*, Normal Training and Adversarial Debiasing [42]. The second class is existing pruning techniques, where we choose two kinds of typical approaches: gradient-based [28] (SGDP) and magnitude-based [10] (LMW) pruning method. The third class is the combination of existing unfairness mitigation and pruning techniques (Fair-SGDP, Fair-LMW). We first utilize Adversarial Debiasing [42] as pre-processing to obtain an unbiased network and then employ pruning approaches on the debiased network. More details can be referred to Supplementary Material.

Training details. We experiment with ResNet-18 [12] network architecture. In the pre-training stage, we train the model for 30 epochs with a batch size of 64 using Adam optimizer with a learning rate of 1e-4. In the fairness-aware pruning stage, the settings of network pruning which is the generator in the algorithm, follow the gradient-based pruning method [28]. The architecture of discriminator is a neural network with one hidden layer and there are 8 nodes in the hidden layer. We train the discriminator with Adam optimizer with a learning rate of 0.001. More details of training can be referred to Supplementary Material.

5.2. Evaluation

As shown in Tab. 2 and Tab. 3, our method effectively mitigates unfairness and maintains accuracy after network pruning on various datasets. For instance, DEO reduces from 0.5352 to 0.2149 and DI increases from 0.3144 to 0.4849 after pruning 80% parameters in Tab. 2a and Tab 3a while the accuracy negligibly drops less than 1%. As compared to Adversarial Debiasing [42] without pruning, the gaps between it and ours are just 0.0535 and 0.0133 on DEO and DI, respectively. It demonstrates that our method largely improves model fairness while not explicitly harming the accuracy and efficiency after pruning a large number of parameters.

Compared with training a sparse network with 20% parameters from scratch, our method outperforms in terms of both model fairness and accuracy. It can be observed in Tab. 2c that our method is 0.2824 lower and 0.12% higher on DEO and accuracy, respectively. This indicates that it is challenging to train a well-performing and fair spare network from scratch and our method proposes an effective solution.

Compared with SOTA network pruning methods, *i.e.*, SGDP [28] and LMW [10], our method substantially ameliorates the fairness of compressed model and achieves comparable accuracy. As shown in Tab. 3b, the increase in DI is less than 0.07 after pruning by SGDP and LMW, how-

Attractive	**PR.**	**ACC ↑**	**DEO ↓**
Normal training	0%	81.68%	0.5352
Adversarial Debiasing [42]		80.24%	0.1614
Train from Scratch	80%	78.73%	0.5071
SGDP [28]		81.48%	0.4991
LMW [10]		78.51%	0.5041
Fair-SGDP	80%	79.18%	0.3261
Fair-LMW		78.66%	0.3640
Ours		**81.00%**	**0.2149**

(a) Results on CelebA when the target label is *Attractive*

Wavy_Hair	**PR.**	**ACC ↑**	**DEO ↓**
Normal training	0%	79.64%	0.1297
Adversarial Debiasing [42]		74.02%	0.0541
Train from Scratch	80%	79.16%	0.1161
SGDP [28]		74.76%	0.1094
LMW [10]		73.44%	0.0774
Fair-SGDP	80%	79.32%	0.0729
Fair-LMW		74.44%	0.1004
Ours		**77.62%**	**0.0576**

(b) Results on LFW when the target label is *Wavy_Hair*

Blond_Hair	**PR.**	**ACC ↑**	**DEO ↓**
Normal training	0%	95.38%	0.5027
Adversarial Debiasing [42]		94.80%	0.2399
Train from Scratch	80%	94.80%	0.5401
SGDP [28]		95.32%	0.4800
LMW [10]		94.68%	0.4615
Fair-SGDP	80%	94.61%	0.4067
Fair-LMW		94.22%	0.4076
Ours		**94.92%**	**0.2580**

(c) Results when on CelebA the target label is *Blond_Hair*

Young	**PR.**	**ACC ↑**	**DEO ↓**
Normal training	0%	82.98%	0.5302
Adversarial Debiasing [42]		81.76%	0.1484
Train from Scratch	80%	82.87%	0.4813
SGDP [28]		82.56%	0.4033
LMW [10]		80.06%	0.4561
Fair-SGDP	80%	81.55%	0.3676
Fair-LMW		81.60%	0.3099
Ours		**80.97%**	**0.2064**

(d) Results on LFW when the target label is *Young*

Table 2: Results of our method FARPrune on **DEO** improvement of CelebA and LFW dataset. **PR.** denotes the pruning ratio, the higher, the more efficient the model. For fairness criterion, the lower **DEO**, the more fair the model.

Attractive	**PR.**	**ACC ↑**	**DI ↑**
Normal training	0%	81.68%	0.3144
Adversarial Debiasing [42]		80.24%	0.4982
Train from Scratch	80%	78.36%	0.3802
SGDP [28]		81.48%	0.2735
LMW [10]		78.39%	0.3825
Fair-SGDP	80%	77.44%	0.4733
Fair-LMW		79.17%	0.4419
Ours		**80.66%**	**0.4849**

(a) Results on CelebA when the target label is *Attractive*

Wavy_Hair	**PR.**	**ACC ↑**	**DI ↑**
Normal training	0%	79.64%	0.7344
Adversarial Debiasing [42]		74.02%	0.8533
Train from Scratch	80%	79.00%	0.7743
SGDP [28]		78.47%	0.8050
LMW [10]		73.38%	0.7838
Fair-SGDP	80%	79.43%	0.8105
Fair-LMW		74.39%	0.8471
Ours		**77.25%**	**0.9297**

(b) Results on LFW when the target label is *Wavy_Hair*

Blond_Hair	**PR.**	**ACC ↑**	**DI ↑**
Normal training	0%	95.38%	0.0546
Adversarial Debiasing [42]		94.80%	0.1729
Train from Scratch	80%	94.61%	0.0704
SGDP [28]		95.32%	0.0765
LMW [10]		94.64%	0.0875
Fair-SGDP	80%	94.61%	0.1115
Fair-LMW		94.22%	0.1188
Ours		**94.80%**	**0.1181**

(c) Results on CelebA when the target label is *Blond_Hair*

Young	**PR.**	**ACC ↑**	**DI ↑**
Normal training	0%	82.98%	0.2032
Adversarial Debiasing [42]		81.76%	0.3178
Train from Scratch	80%	82.82%	0.2239
SGDP [28]		80.06%	0.2883
LMW [10]		80.81%	0.2065
Fair-SGDP	80%	79.64%	0.2290
Fair-LMW		79.22%	0.2259
Ours		**80.54%**	**0.3116**

(d) Results on LFW when the target label is *Young*

Table 3: Results of our method FAPRune on **DI** improvement of CelebA and LFW dataset. **PR.** denotes the pruning ratio, the higher **PR.**, the more efficient the model. For fairness criterion, the higher **DI**, the more fair the model.

ever, our method achieves an increase of nearly 0.2. This indicates that the proposed method formulates model fairness as an explicit objective in network pruning which effectively searches the biased connections.

Compared with the combination of unfairness mitigation and network pruning, *i.e.*, Fair-SGDP and Fair-LMW, we propose an outperforming method that achieves improvements in model fairness and maintenance of accuracy for compressed networks. As shown in Tab. 2d, our method achieves the lowest DEO and comparable accuracy after network pruning. Moreover, our method employs adversarial training to achieve an end-to-end pipeline, which is more efficient than the two-stage combination.

Extensive experiments demonstrate that our method significantly improves model fairness and preserves accuracy and efficiency after pruning a large number of network parameters.

PR.	ACC ↑	DI ↑
0%	81.68%	0.3144
70%	79.18%	0.4903
80%	80.66%	0.4849
90%	80.66%	0.4244
95%	81.51%	0.3891

PR.	ACC ↑	DEO ↓
0%	81.68%	0.5352
70%	81.37%	0.2181
80%	81.00%	0.2149
90%	81.03%	0.3143
95%	80.88%	0.3215

Table 4: Accuracy and fairness of ResNet-18 networks on CelebA at different sparsity levels. The target label is *Attractive* and the sensitive attribute is *Gender*.

5.3. Ablation Study

In this section, we would like to validate generalization of our method by analyzing the factor of hyper-parameters, including the pruning ratio and network architecture.

On Pruning Ratio. We conduct experiments on CelebA dataset and set *Attractive* as the target label and *Gender* as the sensitive attribute. We compress ResNet-18 network under various pruning ratios, *e.g.*, 70%, 80%, 90%, and 95%. Tab. 4 demonstrates that our method effectively ameliorates the fairness of the pre-trained model across different pruning ratios. We also observe that both the accuracy and fairness metrics, *i.e.*, DI and DEO perform better at relatively lower pruning ratios. It can be explained that a higher pruning ratio would prune some meaningful and unbiased connections, which degrades performance and fairness. Tab. 4 shows 70% and 80% is optimal for CelebA dataset.

On Network Architecture. We conduct experiments on different network architectures, including ResNet18, ShuffleNet v2, and MobileNet v2. We set the pruning ratio as 80%. As shown in Tab. 5, our method consistently maintains comparable accuracy and improves fairness criterion *i.e.* DEO and DI after pruning. For instance, our method improves 0.31 and 0.16 on DEO for ResNet18 and ShuffleNet v2 while only sacrificing 0.68% and 0.93% on accuracy. This indicates that our method generalizes well on various network architectures.

Network	ACC ↑	DI ↑
ResNet18 (Dense)	81.68%	0.3144
ResNet18 (80%)	**80.66%**	**0.4849**
ShuffleNet v2 (Dense)	79.34%	0.3566
ShuffleNet v2 (80%)	**78.58%**	**0.4808**
MobileNet v2 (Dense)	81.63%	0.2893
MobileNet v2 (80%)	**79.96%**	**0.4560**

(a) Accuracy and DI of different network architectures.

Network	ACC ↑	DEO ↓
ResNet18 (Dense)	81.68%	0.5352
ResNet18 (80%)	**81.00%**	**0.2149**
ShuffleNet v2 (Dense)	79.34%	0.5086
ShuffleNet v2 (80%)	**78.41%**	**0.3582**
MobileNet v2 (Dense)	81.63%	0.5229
MobileNet v2 (80%)	**78.06%**	**0.3390**

(b) Accuracy and DEO of different network architectures.

Table 5: Accuracy and fairness of different network architectures on CelebA. Target and sensitive label is *Attractive* and *Gender*, respectively.

5.4. Searching fair sub-networks within randomly initialized networks without model training

We have already demonstrated that the success of our method stems from finding a set of connections which, when pruned, incurs the least degradation of accuracy and ameliorates the fairness of the pre-trained network. *Can we find fair sub-networks within randomly initialized networks without any model training?* To answer this question, we use our method to prune a fair sub-network from a randomly initialized network. These results are presented in Tab. 6 where the pruning ratio for each sub-network is 80% and 90% with the ResNet-18 network on CelebA dataset.

Our results show that there exist sub-networks with inborn, matching, or surpassing the fairness of the debiased networks with comparable model size, within randomly initialized networks without any model training. As shown in Tab. 6d, the sub-network through our method achieves 0.3073, 0.0484 lower on DEO than normally pretrained, adversarially debiased network with comparable model size. Furthermore, we demonstrate the consistent existence of fair sub-networks under different sparsity patterns in Tab. 3. We effectively search fair sub-networks under different pruning ratios on different fairness criteria, *e.g.*, DI and DEO. It can be observed that pruning less or more connections, *e.g.*, less than 75% or more than 90%, would harm both accuracy and fairness compared to moderate pruning ratios, *e.g.*, 80% and 85%.

We demonstrate that our method can effectively search

Attractive	**PR.**	**ACC ↑**	**DI↑**
Normal Training	0%	81.68%	0.3144
Adversarial Debiasing [42]	0%	76.17%	0.6126
SGDP [28]	80%	77.60%	0.4293
Ours	80%	**77.08%**	**0.5351**
SGDP [28]	90%	77.15%	0.3355
Ours	90%	**77.26%**	**0.5015**

(a) Results when the target label is *Attractive*

Attractive	**PR.**	**ACC ↑**	**DEO↓**
Normal Training	0%	81.68%	0.5352
Adversarial Debiasing [42]	0%	78.77%	0.1551
SGDP [28]	80%	73.79%	0.4418
Ours	80%	**78.40%**	**0.1959**
SGDP [28]	90%	78.27%	0.4675
Ours	90%	**76.18%**	**0.3444**

(b) Results when the target label is *Attractive*

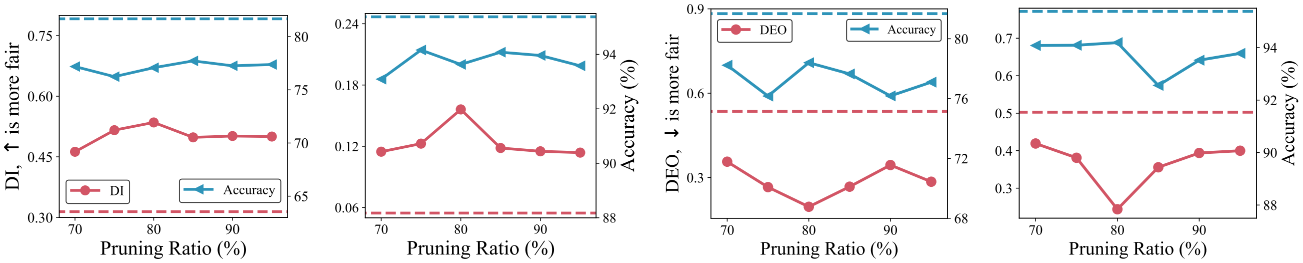
Smiling	**PR.**	**ACC ↑**	**DI↑**
Normal Training	0%	92.50%	0.7090
Adversarial Debiasing [42]	0%	90.91%	0.7557
SGDP [28]	80%	90.78%	0.6604
Ours	80%	**90.17%**	**0.7895**
SGDP [28]	90%	92.54%	0.7115
Ours	90%	**90.94%**	**0.7678**

(c) Results when the target label is *Smiling*

Blond_Hair	**PR.**	**ACC ↑**	**DEO↓**
Normal Training	0%	95.38%	0.5027
Adversarial Debiasing [42]	0%	94.19%	0.2438
SGDP [28]	80%	94.18%	0.4467
Ours	80%	**94.12%**	**0.1954**
SGDP [28]	90%	95.37%	0.4845
Ours	90%	**93.52%**	**0.3939**

(d) Results when the target label is *Blond_Hair*

Table 6: Illustration of the existence of fair sub-networks within unfair randomly initialized networks. We demonstrate the accuracy and fairness metrics DI and DEO of sub-networks with different pruning ratios in ResNet-18 on CelebA with the sensitive attribute *Gender*.



(a) Accuracy and DI when target label is *Attractive* and *Blond Hair* (b) Accuracy and DEO when target label is *Attractive* and *Blond Hair*

Figure 3: Illustrating consistent existence of fair sub-networks within unfair networks with different pruning ratios in ResNet-18 on CelebA. The sensitive attribute is *gender*. The accuracies and fairness criterion of trained original dense networks are annotated using dashed lines.

fair sub-networks within the randomly initialized networks without any model training. This is attributed to the proposed searching process which ensures the searched sub-networks effectively identify redundant and biased weight locations. This indicates that searching for the locations of a subset of weights within a randomly initialized network might be potentially as effective as adversarially debiasing the weight values in the comparable model sizes.

6. Conclusion

In this work, we study the interplay between neural network pruning and fair training objective. We propose to integrate the fair training objective in the pruning technique itself by formulating pruning and debiasing as a two-player adversarial game and enhancing fairness while preserving the accuracy and efficiency of network pruning. Our proposed method consistently performs well across different datasets, network architectures, and pruning ratios. Moreover, we show for the first time that there exist fair sub-networks within randomly initialized networks without any model training. In further work, we would like to go deeper into exploring the existence of and systematically studying the properties of such sub-networks across different network architectures and hyper-parameters

7. Acknowledgements

This work was supported by National Key R&D Program of China (Grant No. 2020AAA0107705), National Natural Science Foundation of China (Grants No. 62122066, U20A20182) and Key R&D Program of Zhejiang (Grant

No. 2022C01018).

References

[1] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075*, 2017.

[2] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91, 2018.

[3] Xiaohan Ding, Tianxiang Hao, Jianchao Tan, Ji Liu, Jungong Han, Yuchen Guo, and Guiguang Ding. Resrep: Lossless cnn pruning via decoupling remembering and forgetting. In *ICCV*, pages 4510–4520, 2021.

[4] Xuanyi Dong and Yi Yang. Network pruning via transformable architecture search. In *NIPS*, pages 759–770, 2019.

[5] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *ICLR*, 2019.

[6] Yonggan Fu, Qixuan Yu, Yang Zhang, Shang Wu, Xu Ouyang, David D. Cox, and Yingyan Lin. Drawing robust scratch tickets: Subnetworks with inborn robustness are found within randomly initialized networks. In *NIPS*, pages 13059–13072, 2021.

[7] Shangqian Gao, Feihu Huang, Weidong Cai, and Heng Huang. Network pruning via performance maximization. In *CVPR*, pages 9270–9280, 2021.

[8] Yiwen Guo, Anbang Yao, and Yurong Chen. Dynamic network surgery for efficient dnns. In *NIPS*, pages 1379–1387, 2016.

[9] Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding. In *ICLR*, 2016.

[10] Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural network. In *NIPS*, pages 1135–1143, 2015.

[11] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. In *NIPS*, page 3323–3331, 2016.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[13] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[14] Gary B. Huang, Vidit Jain, and Erik Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007.

[15] Michael P Kim, Amirata Ghorbani, and James Zou. Multiaccuracy: Black-box post-processing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 247–254, 2019.

[16] Carl Lemaire, Andrew Achkar, and Pierre-Marc Jodoin. Structured pruning of neural networks with budget-aware regularization. In *CVPR*, pages 9108–9116, 2019.

[17] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *ICLR*, 2017.

[18] Mingbao Lin, Rongrong Ji, Yuxin Zhang, Baochang Zhang, Yongjian Wu, and Yonghong Tian. Channel pruning via automatic structure search. In *IJCAI*, pages 673–679, 2020.

[19] Xiaofeng Lin, Seungbae Kim, and Jungseock" Joo. Fairgrape: Fairness-aware gradient pruning method for face attribute classification. In *ECCV*, pages 414–432, 2022.

[20] Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *ICCV*, pages 2755–2763, 2017.

[21] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, pages 3730–3738, 2015.

[22] Zhuang Liu, Mingjie Sun, Tinghui Zhou, Gao Huang, and Trevor Darrell. Rethinking the value of network pruning. In *ICLR*, 2019.

[23] Pranay K Lohia, Karthikeyan Natesan Ramamurthy, Manish Bhide, Diptikalyan Saha, Kush R Varshney, and Ruchir Puri. Bias mitigation post-processing for individual and group fairness. In *ICASSP*, pages 2847–2851, 2019.

[24] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Comput. Surv.*, 54(6):115:1–115:35, 2021.

[25] Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *CVPR*, pages 11264–11272, 2019.

[26] Novi Quadrianto, Viktoriia Sharmanska, and Oliver Thomas. Discovering fair representations in the data domain. In *CVPR*, June 2019.

[27] Inioluwa Deborah Raji, Timnit Gebru, Margaret Mitchell, Joy Buolamwini, Joonseok Lee, and Emily Denton. Saving face: Investigating the ethical concerns of facial recognition auditing. In *AAAI/ACM Conference on AI, Ethics, and Society*, pages 145–151, 2020.

[28] Vivek Ramanujan, Mitchell Wortsman, Aniruddha Kembhavi, Ali Farhadi, and Mohammad Rastegari. What's hidden in a randomly weighted neural network? In *CVPR*, pages 11890–11899, 2020.

[29] Vikram V. Ramaswamy, Sunnie S. Y. Kim, and Olga Russakovsky. Fair attribute classification through latent space de-biasing. In *CVPR*, 2021.

[30] Alex Renda, Jonathan Frankle, and Michael Carbin. Comparing rewinding and fine-tuning in neural network pruning. In *ICLR*, 2020.

[31] Yuji Roh, Kangwook Lee, Steven Whang, and Changho Suh. Fr-train: A mutual information-based approach to fair and robust training. In *ICML*, pages 8147–8157, 2020.

[32] Mhd Hasan Sarhan, Nassir Navab, Abouzar Eslami, and Shadi Albarqouni. Fairness by learning orthogonal disentangled representations. In *ECCV*, pages 746–761, 2020.

[33] Vikash Sehwag, Shiqi Wang, Prateek Mittal, and Suman Jana. HYDRA: pruning adversarially robust neural networks. In *NIPS*, 2020.

[34] Maying Shen, Pavlo Molchanov, Hongxu Yin, and Jose M. Alvarez. When to prune? A policy towards early structural pruning. In *CVPR*, pages 12237–12246, 2022.

[35] Maying Shen, Pavlo Molchanov, Hongxu Yin, and Jose M. Alvarez. When to prune? a policy towards early structural pruning. In *CVPR*, pages 12247–12256, June 2022.

[36] Zi Wang, Chengcheng Li, and Xiangyang Wang. Convolutional neural network pruning with structural redundancy reduction. In *CVPR*, pages 14913–14922, 2021.

[37] Paul Wimmer, Jens Mehnert, and Alexandru Condurache. Interspace pruning: Using adaptive filter representations to improve training of sparse cnns. In *CVPR*, pages 12517–12527, 2022.

[38] Yawen Wu, Dewen Zeng, Xiaowei Xu, Yiyu Shi, and Jingtong Hu. Fairprune: Achieving fairness through pruning for dermatological disease diagnosis. In *MICCAI*, volume 13431, pages 743–753, 2022.

[39] Haoran You, Chaojian Li, Pengfei Xu, Yonggan Fu, Yue Wang, Xiaohan Chen, Richard G. Baraniuk, Zhangyang Wang, and Yingyan Lin. Drawing early-bird tickets: Toward more efficient training of deep networks. In *ICLR*, 2020.

[40] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *WWW*, pages 1171–1180, 2017.

[41] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pages 962–970, 2017.

[42] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, page 335–340, 2018.

[43] Yi Zhang and Jitao Sang. Towards accuracy-fairness paradox: Adversarial example-based data augmentation for visual debiasing. In *MM*, pages 4346–4354, 2020.

[44] Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. In *ICLR*, 2017.