Echoes: Unsupervised Debiasing via Pseudo-bias Labeling in an Echo Chamber

Rui Hu
Beijing Jiaotong University, China
rui.hu@bjtu.edu.cn

Yahan Tu
China University of Geoscience
Beijing, China
yahan.tu@cugb.edu.cn

Jitao Sang
[1]Beijing Jiaotong University, China
[2]Peng Cheng Lab, China
jtsang@bjtu.edu.cn

ABSTRACT

Neural networks often learn spurious correlations when exposed to biased training data, leading to poor performance on out-of-distribution data. A biased dataset can be divided, according to biased features, into bias-aligned samples (i.e., with biased features) and bias-conflicting samples (i.e., without biased features). Recent debiasing works typically assume that no bias label is available during the training phase, as obtaining such information is challenging and labor-intensive. Following this unsupervised assumption, existing methods usually train two models: a biased model specialized to learn biased features and a target model that uses information from the biased model for debiasing. This paper first presents experimental analyses revealing that the existing biased models overfit to bias-conflicting samples in the training data, which negatively impacts the debiasing performance of the target models. To address this issue, we propose a straightforward and effective method called *Echoes*, which trains a biased model and a target model with a different strategy. We construct an "echo chamber" environment by reducing the weights of samples which are misclassified by the biased model, to ensure the biased model fully learns the biased features without overfitting to the bias-conflicting samples. The biased model then assigns lower weights on the bias-conflicting samples. Subsequently, we use the inverse of the sample weights of the biased model for training the target model. Experiments show that our approach achieves superior debiasing results compared to the existing baselines on both synthetic and real-world datasets. Our code is available at https://github.com/isruihu/Echoes.

CCS CONCEPTS

• **Applied computing** → **Law, social and behavioral sciences**;
• **Computing methodologies** → **Machine learning**.
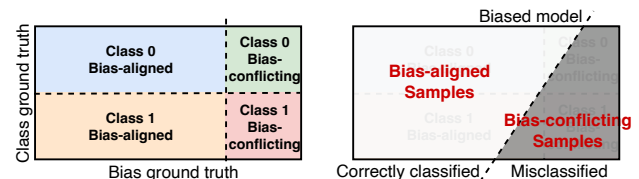
KEYWORDS

unsupervised debiasing; echo chamber

Figure 1: Illustration of using a biased model to differentiate the training samples in terms of bias. Left is the ground truth labels of data samples and right is the division result (i.e., pseudo-bias labels) by the predictions of the biased model.

1 INTRODUCTION

Despite the success of deep neural networks (DNNs), recent studies have revealed that many models have a tendency to learn shortcut cues [3, 15]. DNNs often pick up simple, discriminative cues from the training data, and ignore more complex but important ones [23, 26]. The correlations between the simple cues and the labels are often spurious. A DNN that relies on such simple spurious correlations for predictions will have poor performance on out-of-distribution (OOD) data, i.e., data without such correlations [7, 24]. The datasets with spurious correlations are typically said to be *biased*. Models trained on biased datasets that contain spurious correlations of protected attributes (e.g., gender, race) and labels may suffer from fairness problems. Biased datasets typically contain a majority of *bias-aligned* samples and a minority of *bias-conflicting* ones. For bias-aligned samples, ground truth labels are correlated with both target features and biased features. While for bias-conflicting samples, labels are correlated only with target features. As an example, suppose that the bird bodies and the backgrounds are target features and biased features for bird classification, respectively, and as birds often co-occur with the sky, birds in the sky are *bias-aligned*, and birds on the land are *bias-conflicting*. Clearly, maximizing the effect of bias-conflicting samples is the key point to preventing models from learning spurious correlations.

Existing debiasing methods can be divided into *supervised* and *unsupervised* debiasing in terms of the availability of bias information in the training phase. For supervised debiasing, some methods require bias annotations, i.e., each training sample is provided with a bias label [8, 21, 25]. Other methods use prior knowledge of the bias type, which enables them to design a debiasing network tailored for the predefined bias type [1, 4]. There are limitations to supervised debiasing: bias mitigation cannot be performed when bias labels or types are not available, and it requires expensive labor to manually identify the bias types and get bias labels. Since accessing bias

information in the training phase is unreasonable, recent works focus on the more challenging and realistic task - unsupervised debiasing, which mitigates the bias without annotation.

Unsupervised debiasing methods work with the assumption that biased features are easier to be learned than target features, meaning that they are picked up earlier by the model during training [13, 17, 19]. Following this assumption, the works typically train two models: one is an auxiliary biased model specialized to learn biased features so that the predictions of the biased model can be used as pseudo-bias labels, i.e., whether a sample is bias-aligned or bias-conflicting (See Fig.1 for illustration). And the other is the desired target model that using pseudo-bias labels from the biased model for debiasing. Therefore, a well-designed biased model is critical to improve the debiasing performance of the target model [14].

However, our experimental analysis reveals that existing bias models, e.g., an ERM model [17] or a GCE model [13, 19] (see Sec 2.1), fail to provide accurate pseudo-bias labels, i.e., unable to identify bias-conflicting samples in the training data. Due to the scarcity of the bias-conflicting samples, the biased models are prone to memorize these samples to minimize the training loss, leading to the overfitting on them. For example, the error rate of an ERM model on the bias-conflicting sample of CelebA [18] is close to zero at the end of training. This, in turn, negatively impacts the debiasing performance of the target modes. Besides, through experiments, we find that partially biased data is sufficient for the model to learn biased features as they are easy-to-learned.

To address this, we propose a simple and effective unsupervised debiasing method called **Echoes** (**E**cho **ch**amber f**o**r d**e**bia**s**ing) which trains a biased model and a target model in an echo chamber. Unlike the training of existing biased model, we are inspired by the *Echo chamber* [1] phenomenon in media communication, and continually adjust the weights of the samples during the training phase like the information changing in an echo chamber, so that the biased model can distinguish bias-conflicting samples from bias-aligned ones. Specifically, we consider that in the early training phase, the majority of samples correctly classified by the biased model are bias-aligned, while the samples misclassified by the model consist of a minority of bias-aligned samples and a majority of bias-conflicting ones. Then we can "hide" these error samples by reducing their weights. The remaining samples that are correctly classified by the biased model will be retrained in the next training round, just like the echo in an echo chamber. In the end, the biased model sufficiently learns from bias-aligned samples, while underfitting for bias-conflicting ones, which indicates that more accurate bias information can be provided to the target model.

In the echo chamber, a lower sample weight indicates that it is more often misclassified by the biased model, and a higher weight indicates that it is more often correctly classified. This means that samples with lower weights are harder to be learned by the biased model, making them more important for the target model. Based on this analysis, we simply use the inverse of the biased model's sample weights as the target model's sample weights, so that the target model can focus on learning hard samples to prevent learning spurious correlations.
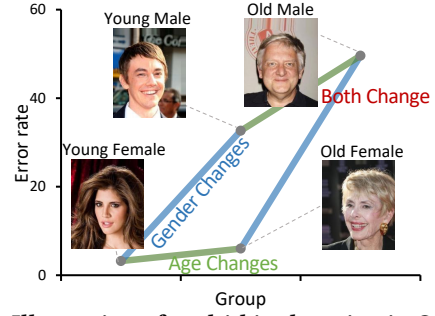

Figure 2: Illustration of multi-bias learning in *Smiling* classification. Model error rates increases when test samples change from bias-aligned (e.g., young female) to single bias-conflicting (e.g., young male), and reaching a maximum when both bias conflict (e.g., old male).

We evaluate our method on both the *single-biased* benchmarks that are commonly used in existing works and the *multi-biased* benchmarks that are more challenging and rarely considered. Fig.2 illustrates the phenomenon of multi-bias learning. Experimental results suggest that our method consistently outperforms prior methods in both multi-bias and single-bias settings. Ablation studies show the advantages of our biased model in biased feature learning compare to existing ones.

In summary, our contributions are as follows:

- We find that existing biased models for unsupervised debiasing fail to provide accurate pseudo-bias labels, which negatively impacts the debiasing performance.
- We propose a simple unsupervised debiasing method called Echoes, which trains an auxiliary biased model and a target model in an echo chamber.
- We perform extensive experiments with single-biased and multi-biased benchmarks, demonstrating that Echoes outperforms the existing methods.

2 A CLOSE LOOK AT THE TRAINING OF BIASED MODELS

This section describes the details of two analytical experiments in which we observe the overfitting of the biased model on the training data, and partially biased data is sufficient for the model to learn biased features. In Section 2.1, we first introduce the common choices of biased models, that we utilize for the observation. Then, we elaborate the results of the experiments in Section 2.2 and 2.3.

2.1 Background

Since annotating bias labels or identifying the bias types in advance is challenging and labor intensive, recent works consider an auxiliary model that specializes in learning biases to guide the training of the debiased model [13, 17, 19]. A good biased model should sufficiently learn biased features to have the ability to differentiate the training samples in terms of the biases. For instance, correctly classifying bias-aligned samples and misclassifying bias-conflicting ones, so that the target model can focus on learning the wrong samples based on the predictions of the biased model.

There are two common choices for a biased model: an ERM model or a GCE model. Liu et al. [17] train an ERM model as an
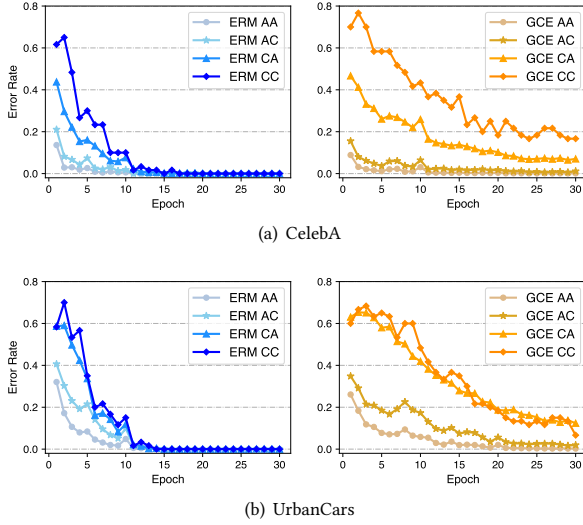
---

[1]In media, an echo chamber is an environment or ecosystem in which participants encounter beliefs that amplify or reinforce their preexisting beliefs by communication and repetition inside a closed system and insulated from rebuttal.

(a) CelebA

(b) UrbanCars

Figure 3: The classification error rates of ERM and GCE models on the training data of CelebA and UrbanCars.

identification model to select samples with high training loss. ERM minimizes the following objective:

$$J_{ERM}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(x_i, y_i; \theta) \tag{1}$$

where $\ell(x_i, y_i; \theta)$ is the cross-entropy loss of the model parameterized by θ for the sample (x_i, y_i). They consider that ERM models tend to fit samples with easy-to-learned spurious correlations, i.e., bias-aligned samples. Then they train a final model by upweighting the points in the error set identified by the ERM model. In addition, Nam et al. [19] use the Generalized cross entropy (GCE) loss [28] to encourage the model to focus on the samples with high probability values so that amplify the bias. The GCE loss is defined as:

$$GCE(p(x; \theta), y) = \frac{1 - p_y(x; \theta)^q}{q} \tag{2}$$

where $p(x; \theta)$ and $p_y(x; \theta)$ are softmax outputs of the model and the softmax probability of the target class y, respectively, and $q \in (0, 1]$ is a hyperparameter that controls the degree of amplification. In a biased dataset, the samples for which the model has high probability values are mostly bias-aligned samples, the GCE loss encourages the model to focus on such samples, leading the model to be biased.

2.2 Biased Models Overfit to the Bias-conflicting Samples

To confirm whether the existing biased models can achieve the expectation, i.e., discriminating the training samples in terms of the biases, we train ERM model and GCE model on CelebA [18] and UrbanCars [15]. Both datasets have one target attribute and two biased attributes. In CelebA, the target attribute is *smiling* and the biased attributes are *gender* and *age*. In UrbanCars, the target attribute is *car* and the biased attributes are *background* and *co-occurring object* (see Sec 4.1 for datasets details). Then, the training data can be divided into four groups based on the different biased

Table 1: The bias-aligned accuracy on the test sets of CelebA and UrbanCars. Training dataset size is 1.0 means using 100% training data for model training.

Train dataset size	Bias-aligned Acc.	
	CelebA	UrbanCars
1.0	98.4	92.4
0.5	99.2(+1.0%)	86.0(-6.9%)
0.2	94.4(-4.1%)	85.2(-7.8%)
0.1	94.0(-4.5%)	80.0(-13.4%)

attribute values. For example, the CelebA dataset can be divided into {young female, old female, young male, old male}, where the first group is the bias-aligned samples and the other three groups are the bias-conflicting samples. For convenience, we use **AA** to refer to the group which is aligned on both biased attributes (e.g., young female), **AC** to the group which is aligned on the first biased attribute and conflicting on the second biased attribute (e.g., old female), **CA** to the group which is conflicting on the first biased attribute and aligned on the second biased attribute (e.g., young male), and **CC** to the group which is conflicting on both biased attributes (e,g., old male), respectively.

Overfitting on the bias-conflicting samples. We train the models for 30 epochs, and after each epoch, we calculate the error rate of the biased models on each group of training data. Fig.3 shows the experimental results. Apparently, as the training progresses, the models overfit to the bias-conflicting samples (AC, CA and CC). The error rates of all groups of the training data decrease, indicating that the biased model becomes less capable of discriminating between samples in terms of bias. Such performance is not consistent with what we expect from the bias models.

Early stopping is not good enough. Early stopping [27] is a common strategy to prevent overfitting of the model. However, for the training of biased models, there are several limitations to applying early stopping. On the one hand, early stopping usually requires a validation set for deciding when to stop. However, in our setting, we do not have information on bias and therefore cannot construct a validation set that can be used to prevent the model from overfitting to bias-conflicting samples. On the other hand, one can select models that are in the very early stages of training (e.g., models trained in one or two epochs). The models have not been sufficiently trained at that time and therefore not yet overfit to the bias-conflicting samples. However, from Fig.3, at the beginning of training, the biased features are also not fully learned by the models, leading to a non-negligible error rate on the bias-aligned samples. And due to the large number of bias-aligned samples, even a small error rate can lead to a number of bias-aligned samples being categorized by the model as bias-conflicting samples, which is detrimental to the training of the target model.

2.3 Part of Biased Data is Sufficient for Bias Learning

In addition to early stopping, reducing the size of the training data can mitigate model overfitting because the model cannot overfit to
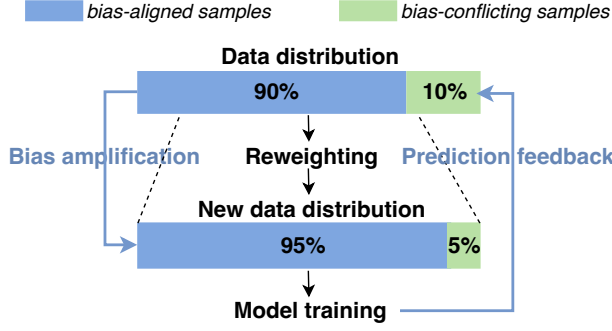
Figure 4: The training process of the biased model. The data distribution becomes increasingly biased with training, just like an echo chamber.

unseen samples. And we find part of biased data is sufficient for the biased model to learn biased features.

We train the ERM models on different ratios of the CelebA [18] and UrbanCars [15] datasets. To be specific, we use 100%, 50%, 20%, and 10% of the training samples to train the model respectively. For each ratio, we train the model for 100 epochs, and report accuracy of bias-aligned samples of test data, which reflects how well the model learned for biased features. Table 1 shows the results. The accuracy of the model on bias-aligned samples mostly decreases when the size of training data decreases. But the decrease in accuracy is small compared to the decrease in data size. For example, training with only 20% of the UbanCars training samples, the model accuracy dropped by only less than 8% compared to training with 100% of samples.

3 METHOD

Motivated by such an observation in Section 2.3, we first introduce our training strategy of a biased model, which assumes that biased features are "easy-to-learned" than target ones (Sec.3.1). Then, we introduce a debiasing approach based on samples re-weighting (Sec.3.2). Finally, we describe the complete method that combines the debiasing approach with our auxiliary biased model (Sec.3.3).

3.1 Biased Model

Training goal. In a highly biased dataset, bias-conflicting samples constitute only a small fraction of the training data, we must maximize the effect of them to prevent the target model from learning spurious correlations. However, through the observation in Section 2.2, both ERM and GCE models overfit the bias-conflicting samples in the training data. The goal of training a biased model is thus to ensure that the bias model fully learns the biased features while protecting the bias-conflicting sample from being learned.

Constructing an echo chamber. To achieve the goal, our idea is to constructing an "echo chamber" environment when training the biased model. Based on the assumption that the model gives priority to learning biased information, we assign weights in the echo chamber to achieve homogeneity and diffusion of information during sample selection, thus achieving full learning of biased features. Our insight is that samples that are correctly classified are easy-to-learned for the biased model, i.e., mostly bias-aligned

---

Algorithm 1 Training of the biased model

1: **Input**: dataset $D = \{(x_i, y_i)\}$, model f_θ, number of iterations T, batch size B, sample weights W_B, hyperparameter α.
2: **Output**: a biased model f_θ
3: Initialize θ, and initialize $W_B = \mathbb{1}$
4: **for** t = 1,...,T **do**
5:    /** Train model f_θ **/
6:    **for** b = 1,...,B **do**
7:        Draw a batch $\mathcal{B} = \{(x_i, y_i)\}_{i=1}^{n}$
8:        $\theta \leftarrow \theta - \nabla W \mathcal{L}_{CE}(\mathcal{B})$
9:    **end for**

10:    /** Update weights W **/
11:    Get predictions $\hat{Y} = \{\hat{y}_i\}$ of the model f_θ on D
12:    Count the error rate for each class $E = \{e_i\}_{i=1}^{C}$
13:    **for** $c = 1, \ldots, C$, **do**
14:        **if** $e_c < t_{error}$ **do**
15:            $W_c \leftarrow \alpha W_c$
16:        **end if**
17:    **end for**
18: **end for**

---

samples, while samples that are misclassified by the model contain a majority of bias-conflicting samples and a minority of bias-aligned samples. Therefore, reducing the weights of misclassified samples is equivalent to constructing an echo chamber environment that allows the model to learn only the samples it tends to learn.

Approach. Fig.4 illustrates the approach. At the beginning of the training, we have a biased dataset that each sample has with equal weights for each sample (e.g., 1). Then, after the model is trained for one epoch, each sample is re-weighted with the model's prediction results. The re-weighting rule can be formulated as:

$$w^i(x) = \begin{cases} w^{i-1}(x), & if \ f_\theta(x) = y \\ \alpha w^{i-1}(x), & if \ f_\theta(x) \neq y \end{cases} \tag{3}$$

where $w^i(x)$ is the weight of sample x at the i-th epoch, $f_\theta(x)$ is the prediction result of the biased model for the sample x, y is the ground truth label, and $\alpha \in [0, 1]$ is a hyperparamter that controls the degree of weight reduction of the sample. If $\alpha = 0$, it is equivalent to removing the samples that the model predicts incorrectly from the training data after each epoch, and if $\alpha = 1$, it degenerates into standard ERM training. To improve the stability of the training, we use a hyperparameter t_{error} (i.e., the threshold of error rate) to control whether to perform the re-weighting operation after a training epoch. We only re-weight those samples which are in a class with an error rate less than t_{error}. The high error rate of a class indicates that the model has not learned the features of this class well, thus the error samples of this class are not bias-conflicting at this time. We define the biased loss as follows:

$$L_{biased} = W_{biased}CE(P(X; \theta), Y) \tag{4}$$

where CE is the cross-entropy loss and $P(X; \theta)$ is the softmax output of the biased model. The pseudo-code of the training of the biased model is presented in Alg. 1.
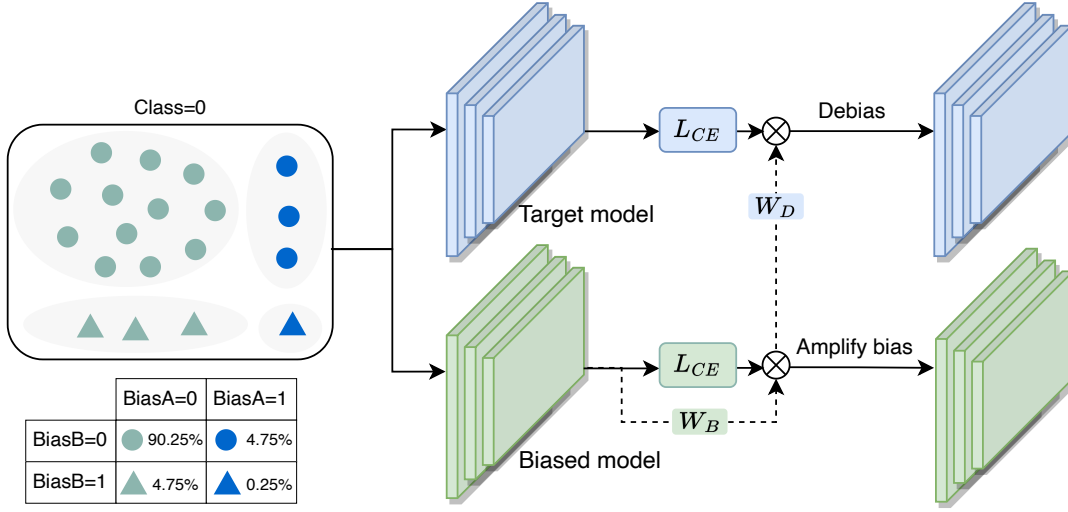
Figure 5: Illustration of our proposed method *Echoes*. We jointly train a biased model and a target model with the biased training data, where the sample weights W_B (for the biased model) and W_D (for the target model) change continuously as the models training.

3.2 Target Model

Training goal. The training goal of the target model f_D is to learn debiased representation. Since we do not have bias information about the training data, we need to use the information from the biased model f_B to guide the training of the target model.

Debiasing f_D via f_B. Since biased features are "easy-to-learned" than target ones, the cross-entropy loss values of bias-conflicting samples are relatively higher than those of bias-aligned ones. Recent works [19, 22] then re-weight the sample weights $w(x)$ for training the target model with such a characteristic. Specifically, Nam et al. [19] formulated $w(x)$ as:

$$w(x) = \frac{\mathcal{L}_{CE}(f_B(x), y)}{\mathcal{L}_{CE}(f_B(x), y) + \mathcal{L}_{CE}(f_D(x), y)} \qquad (5)$$

where $f_B(x)$ and $f_D(x)$ indicate the prediction outputs of f_B and f_D, respectively. With this formula, the bias-conflicting samples will have high re-weighting values, and bias-aligned samples have low ones. However, Nam et al. [19] use the GCE model as the biased model, which overfit to bias-conflicting samples at the end of the training (Sec.2.2), thus bias-conflicting samples have the same low cross-entropy loss as bias-aligned ones.

Our biased model has no such problem because we protect the bias-conflicting samples from being learned by reducing their weights with the biased predictions. In other words, bias-conflicting samples have lower weights compared to bias-aligned ones. Therefore, we can naturally use these weights directly for the training of the target model. Specifically, we simply use the inverse of the sample weights for biased model as the ones of the target model:

$$w(x) = \frac{1}{w_{biased}(x)} \qquad (6)$$

thus the bias-conflicting samples have high re-weighting values.

Algorithm 2 Echoes

1: **Input**: dataset $D = \{(x_i, y_i)\}$, biased model f_θ, target model g_Θ, number of iterations T, batch size B, biased sample weights W_B, debiased sample weights W_D, hyperparameter λ.
2: **Output**: a biased model f_θ and a target model g_Θ
3: Initialize θ, Θ, and initialize $W_B = \mathbb{1}$, $W_D = \mathbb{0}$
4: **for** t = 1,...,T **do**
5:   **for** b = 1,...,B **do**
6:    Draw a batch $\mathcal{B} = \{(x_i, y_i)\}_{i=1}^n$ from D
7:    $\mathcal{L} = W_B \mathcal{L}_{CE}(f_\theta) + \lambda W_D \mathcal{L}_{CE}(g_\Theta)$
8:    Update θ and Θ with $\nabla \mathcal{L}$
9:    Update W_B (see Alg. 1)
10:    Assign $W_D = \frac{1}{W_B}$
11:    Class-level balancing on W_D (see Eq.7)
12:   **end for**
13: **end for**

Balancing class weights. After re-weighting the samples using the above formula, the sum of weights for different classes of samples is likely to be different, resulting in class imbalance. To avoid this problem, we perform class-level weights balancing after applying Eq.6. Specifically, for a sample (x, y) that $y = c \in C$, the weight adjustment is as follows:

$$w(x) := \frac{\prod_{g \in C}(\sum_{j \in I_g} w_j)}{\sum_{i \in I_c} w_i} w(x) \qquad (7)$$

where I_c is the index set of samples that the label $y = c$. After the adjustment, each class has the same sum of sample weights. The debiased loss for the target model is as follows:

$$L_{debiased} = W_{debiased} CE(P(X; \theta), Y). \qquad (8)$$

Table 2: Debiasing results on CelebA dataset. For classification performance, we report average group accuracy and worst group accuracy. For model fairness, we report gender gap, age gap, and the average of them. We blod the best results for each column.

Task	Method	bias label	Accuracy		Bias reliance		
			Avg group acc	Worst group acc	Gender gap↓	Age gap↓	Avg bias gap↓
Smiling	Vanilla	✗	$76.0_{\pm0.2}$	$36.0_{\pm1.7}$	$41.0_{\pm0.6}$	$11.1_{\pm1.0}$	$26.1_{\pm0.2}$
	GroupDRO	✓	$76.2_{\pm0.5}$	$37.9_{\pm1.6}$	$40.6_{\pm1.3}$	$11.3_{\pm0.9}$	$25.9_{\pm1.0}$
	LfF	✗	$75.0_{\pm0.2}$	$35.5_{\pm2.0}$	$42.1_{\pm0.2}$	$9.8_{\pm1.7}$	$25.9_{\pm0.9}$
	JTT	✗	$\mathbf{77.8_{\pm0.6}}$	$38.7_{\pm2.4}$	$38.1_{\pm0.5}$	$11.1_{\pm1.7}$	$24.6_{\pm0.7}$
	DebiAN	✗	$74.5_{\pm0.6}$	$41.1_{\pm4.3}$	$38.0_{\pm1.4}$	$11.7_{\pm1.3}$	$24.9_{\pm1.4}$
	BPA	✗	$76.3_{\pm0.6}$	$37.1_{\pm3.8}$	$37.6_{\pm1.6}$	$12.1_{\pm0.9}$	$24.9_{\pm1.2}$
	Echoes	✗	$69.5_{\pm2.7}$	$\mathbf{58.7_{\pm4.6}}$	$\mathbf{7.3_{\pm1.9}}$	$\mathbf{6.8_{\pm1.8}}$	$\mathbf{7.1_{\pm1.0}}$
Narrow Eyes	Vanilla	✗	$60.6_{\pm0.9}$	$13.1_{\pm0.9}$	$62.7_{\pm1.0}$	$19.4_{\pm0.3}$	$41.1_{\pm0.6}$
	GroupDRO	✓	$59.8_{\pm0.7}$	$13.1_{\pm0.9}$	$63.1_{\pm1.5}$	$19.3_{\pm1.6}$	$41.2_{\pm0.8}$
	LfF	✗	$61.2_{\pm0.4}$	$14.7_{\pm0.7}$	$56.9_{\pm0.6}$	$20.9_{\pm0.2}$	$38.9_{\pm0.2}$
	JTT	✗	$61.5_{\pm0.5}$	$12.5_{\pm1.3}$	$60.1_{\pm1.3}$	$21.9_{\pm0.3}$	$41.0_{\pm0.6}$
	DebiAN	✗	$\mathbf{62.4_{\pm0.9}}$	$16.3_{\pm1.2}$	$56.1_{\pm1.3}$	$19.5_{\pm0.8}$	$37.8_{\pm0.9}$
	BPA	✗	$61.6_{\pm0.9}$	$15.5_{\pm3.4}$	$57.5_{\pm2.0}$	$18.7_{\pm1.4}$	$38.1_{\pm1.1}$
	Echoes	✗	$62.1_{\pm0.3}$	$\mathbf{48.3_{\pm3.1}}$	$\mathbf{7.9_{\pm4.0}}$	$\mathbf{15.7_{\pm1.8}}$	$\mathbf{11.8_{\pm2.1}}$
Arched Eyebrows	Vanilla	✗	$61.9_{\pm0.4}$	$9.6_{\pm1.5}$	$63.5_{\pm0.5}$	$19.9_{\pm0.4}$	$41.7_{\pm0.4}$
	GroupDRO	✓	$60.4_{\pm0.5}$	$12.3_{\pm3.1}$	$62.2_{\pm1.0}$	$\mathbf{18.2_{\pm1.3}}$	$40.2_{\pm0.2}$
	LfF	✗	$60.8_{\pm0.4}$	$14.9_{\pm1.7}$	$56.9_{\pm1.0}$	$22.5_{\pm0.6}$	$39.7_{\pm0.6}$
	JTT	✗	$\mathbf{62.4_{\pm0.3}}$	$12.8_{\pm0.6}$	$59.5_{\pm0.9}$	$21.8_{\pm0.6}$	$40.7_{\pm0.7}$
	DebiAN	✗	$61.3_{\pm0.3}$	$13.9_{\pm0.9}$	$58.3_{\pm1.3}$	$18.7_{\pm1.2}$	$38.5_{\pm0.3}$
	BPA	✗	$61.7_{\pm0.1}$	$13.2_{\pm1.0}$	$58.1_{\pm0.6}$	$22.9_{\pm1.4}$	$40.5_{\pm0.4}$
	Echoes	✗	$58.0_{\pm1.6}$	$\mathbf{35.5_{\pm2.9}}$	$\mathbf{7.5_{\pm3.1}}$	$21.7_{\pm3.0}$	$\mathbf{14.6_{\pm0.5}}$

3.3 Complete Method

We now have an approach for training a biased model and an approach for training a target model once we have a biased model. We follow the training strategy of Nam et al. [19] that jointly train the bias model and the target model using followed loss:

$$\mathcal{L} = \mathcal{L}_{biased} + \lambda\mathcal{L}_{debiased} \tag{9}$$

where λ is a hyperparameter for balancing the loss. Fig.5 illustrates the complete method, and the pseudo-code of *Echoes* is presented in Alg.2.

4 EXPERIMENT

We now validate the effectiveness of the proposed method on debiasing benchmarks. We first introduce the experimental settings in Section 4.1, then, we introduce the experimental results in Section 4.2, 4.3 and 4.4.

4.1 Setup

Datasets.

CelebA [18] is a large-scale face dataset for face image recognition, containing 40 attributes for each image, of which *Gender* and *Age* are protected attributes that should be prevented from being learned by the model. However, most existing works design and evaluate methods under the assumption that a single bias is present in the data. We set both *gender* and *age* as the biased attributes, and set *Smiling*, *Narrow Eyes* and *Arched Eyebrows* as the target attributes. We set the skew ratio to 95% for both *gender* and *age*, and sample
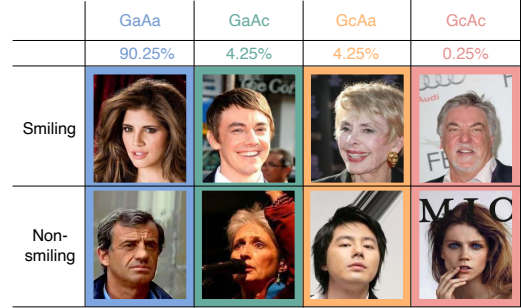

Figure 6: Unbalanced groups in CelebA's training set based on two biased attributes: *Gender* and *Age*. And *GaAc* means samples that are gender-aligned and age-conflicting.

8000 images in the original CelebA as the training set. The data can be divided into 8 groups using two bias attributes and one target attribute. Fig.6 shows the training data distribution for the *Smiling* classification task. For the test set, we sample 1000 images, and each group has an equal number of samples (i.e., 125).
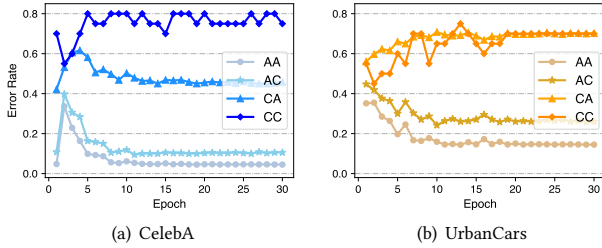
UrbanCars is a synthetic car dataset created by Li et al. [15] which has two biased attributes: *background (bg)* and *co-occurring object (co-obj)*. Li et al. [15] segment and recombine the images of the three datasets: Stanford Cars [12] for car images, Places [29] for background images, and LVIS [5] for co-occurring object images. The cars, the backgrounds, and the co-occurring objects can all be divided into {*urban, country*}. The skew ratios of the biased attributes in the train set and the test set are the same as the CelebA

Table 3: Debiasing results on UrbanCars dataset. For model fairness, we report background (BG) gap, co-occurring object (Co-obj) gap, and the average of BG gap and Co-obj gap.

Task	Method	bias label	Accuracy		Bias reliance		
			Avg group acc	Worst group acc	BG gap↓	Co-obj gap↓	Avg bias gap↓
Car Object	Vanilla	×	60.6±0.3	19.2±1.5	48.9±1.0	23.7±0.5	36.3±0.4
	GroupDRO	✓	60.7±0.7	20.8±1.1	46.8±2.3	20.4±0.8	33.6±0.7
	LfF	×	64.0±1.5	35.7±5.4	27.7±3.8	**5.7±2.8**	16.7±1.8
	JTT	×	**69.5±2.3**	33.3±6.9	39.3±3.1	13.9±0.7	26.6±1.8
	DebiAN	×	62.4±0.9	27.2±2.5	38.1±2.5	15.0±1.5	26.5±1.9
	BPA	×	63.8±0.8	21.6±4.9	32.3±0.8	29.7±2.1	31.0±0.7
	Echoes	×	63.2±0.7	**48.5±2.8**	**4.8±2.2**	15.7±0.3	**10.3±1.2**

Table 4: Debiasing results on BFFHQ dataset. *Ratio* represents the ratio of bias-conflicting samples in the training set.

Ratio(%)	Metrics	Vanilla	LfF	JTT	DebiAN	BPA	Echoes
0.5	Avg	72.8	73.4	**76.0**	74.7	74.7	71.7
	Worst	35.6	33.2	40.8	40.0	37.2	**48.8**
	Bias	52.6	50.8	47.0	49.4	47.8	**38.2**
1.0	Avg	74.4	73.5	**76.2**	74.5	74.9	70.9
	Worst	40.4	30.0	41.2	39.6	45.2	**52.0**
	Bias	41.6	50.6	45.2	47.4	44.2	**19.8**
2.0	Avg	78.5	77.2	**80.4**	78.2	79.9	73.5
	Worst	50.0	48.4	55.6	55.6	54.8	**60.0**
	Bias	40.2	40.4	36.8	38.4	37.4	**21.0**

Table 5: The bias-aligned accuracy and bias-conflicting accuracy on the test sets of CelebA and UrbanCars.

Dataset	Accuracy	ERM	GCE	Ours
CelebA	Bias-aligned↑	98.3	98.1	**98.4**
	Bias-conflicting↓	68.5	68.4	**65.5**
UrbanCars	Bias-aligned↑	**93.5**	91.7	91.9
	Bias-conflicting↓	49.6	49.9	**45.3**

(a) CelebA   (b) UrbanCars

Figure 7: The classification error rate of our biased model on the training data of CelebA and UrbanCars.

dataset, i.e., 8000 unbalanced images for training and 1000 balanced images for testing.

BFFHQ (Biased FFHQ) [13] is constructed based on the real-world face dataset FFHQ [9], where the label is age and the biased feature is gender.

Baselines. To evaluate the effectiveness of our method in debiasing, we compare it with the prior methods including vanilla network, GroupDRO [21], LfF [19], JTT [17], DebiAN [16], BPA [22]. Vanilla denotes the classification model trained only with the original cross-entropy (CE) loss, without any debiasing strategies. Group-DRO explicitly leverages the bias labels (e.g., the gender labels in CelebA) during the training phase, while others require no prior knowledge of the biases. LfF and JTT are evaluated on the single-bias datasets only, while DebiAN and BPA are evaluated on the multi-bias datasets in their respective works.

Points of comparison. We aim to answer two main questions: (1) How does the classification performance of *Echoes* compare with other methods? (2) How does the fairness of *Echoes* compare to other methods? To answer the first question, we report the average group accuracy and the worst group accuracy. To answer the second question, we check whether the model predictions meet the fairness criterion *bias gap*, which indicates the drop in model accuracy when the group shifts. For example, the *gender gap* in CelebA is as follows:

$$gender\ gap = \frac{|ACC_{GaAa} - ACC_{GcAa}| + |ACC_{GaAc} - ACC_{GcAc}|}{2}$$

(10)

Implementation details. In the experiments, we use ResNet-18 [6] for all the datasets. For the training, we set the batch size as 256 and the learning rate as 3e-4. We train the models for 100 epochs with the Adam [11] optimizers in the experiments. We set $t_{error} = 0.3$ for UrbanCars, $t_{error} = 0.5$ for CelebA and BFFHQ, and $\alpha = 0.5, \lambda = 1000$ for all datasets. Since there is no prior knowledge of the biases for model selection, we report the accuracy of the last epoch. All experimental results have averaged over three independent trials.

4.2 Main Result

Debiasing with multi-bias. Table 2 and Table 3 show the quantitative experimental results on CelebA and UrbanCars, respectively. On CelebA, our method achieves the best worst group accuracy and average bias gap on all three tasks. The average group accuracy of our method is relatively low, but the worst group accuracy is high, which indicates that the accuracy of different groups is close, and this also reflects the debiasing performance of our method. On UrbanCars, our method also achieves the best worst group accuracy and average bias gap which shows the effectiveness of our method in handling multi-bias.

Table 6: The comparison of the results of our method with and without class balancing on CelebA.

Task	Metrics	w/o class balancing	w/ class balancing
Smiling	Avg	67.5	**69.5**
	Worst	54.1	**58.7**
Narrow Eyes	Avg	60.3	**62.1**
	Worst	41.0	**48.3**
Arched Eyebrows	Avg	56.7	**58.0**
	Worst	35.2	**35.5**

Debiasing with single-bias. Table 4 shows the results on BFFHQ [13]. The bias is the accuracy drop from bias-aligned accuracy to bias-conflicting accuracy. The results show that our method also outperforms baselines in debiasing under single-bias setting.

4.3 The Effectiveness of Our Biased Model

For the biased model, we want it to (i) not overfit to the bias-conflicting samples in the training set, and (ii) fully learn the biased features in order to pass information to the target model. Fig.7 and Table 5 show the advantages of our bias model compared to ERM model and GCE model.

As show in Fig.7, the error rate of our model on the bias-conflicting samples of the training set is consistently high, i.e., our biased model does not overfit to the bias-conflicting samples. As show in Table 5, our biased model achieves the lowest bias-conflicting accuracy on both test sets of CelebA and Urbancars and the highest bias-aligned accuracy on CelebA, which indicates that the model sufficiently learns the biased features rather than the target features.

4.4 Ablation Study

Class-level balancing. Table 6 shows the comparison of the results of *Echoes* with and without class-level balancing on CelebA. Both the average gourp and worst group accuracy of the model decrease on all tasks when without class-level balancing, indicating the effectiveness of the class-level balancing.

Different values of α. We use $\alpha \in [0, 1]$ to control the degree of weight reduction of the samples when training the biased model. Fig.8 shows the results of different values of α. When α is close to 0, the model ignores too many samples, resulting in insufficient learning, so both worst group and average group accuracy are low. When α is close to 1, the model is similar to ERM, leading to low worst group accuracy. Therefore, taking α as 0.5 is appropriate.

5 RELATED WORK

Debiasing methods. Early works on debiasing assume some prior knowledge about the bias (i.e., supervised debiasing). Some methods require that each training sample is provided with a bias label (e.g., the backgrouds of the car images) [8, 21, 25]. Other methods use knowledge of the bias type in the datasets, such as color or texture in images, and this information is typically used to design custom architectures [1, 4]. For example, Geirhos et al. [4] use a style
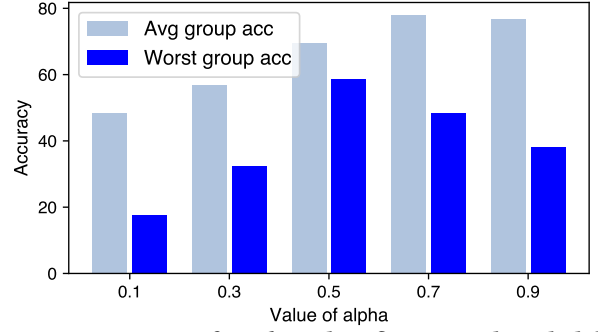
Figure 8: Accuracy of *Smiling* classification task with different values of alpha α on CelebA.

transfer network to generate StylizedImageNet, a stylized version of ImageNet. Models trained on such data can mitigate texture bias. Rencet works focus on the more realistic and challenging setting, where debiasing without bias information (i.e., unpervised debiasing) [13, 17, 19]. The works assume that biased features are "easy-to-learned" than robust ones and thus train an auxiliary model that intentionally relies primarily on the biased features. However, the existing proposed auxiliary models (e.g., ERM model [17] or GCE model [19]) overfit to the bias-conflicting samples of the train data which impairs the debiasing performance of the method. In contrast, our method constructes an echo chamber environment (i.e., reduces the weights of bias-conflicting samples) when training the auxiliary model, which mitigates the overfitting.

Echo chamber effect. The echo chamber effect is a common phenomenon in social media and recommender systems [2, 20]. Participants are exposed to homogenized views, which are repeatedly disseminated and diffused, eventually leading to the polarization of information and the formation of a closed system. For example, users in the recommendation system will constantly click on the items of interest and ignore the uninteresting ones, the recommendation system will optimize the recommendation results, and finally the items displayed will be of interest to the users. Although this can introduce exposure bias [10] in the recommender system, we use the echo chamber effect to amplify the bias in the training data so that the biased model can better learn biased features.

6 CONCLUTION

We presented *Echoes*, a method for unsupervised debiasing that jointly trains a biased model and a target model in an echo chamber environment. The echo chamber is constructed by continuously reducing the weights of samples misclassified by the biased model under the "easy-to-learned" hypothesis. Experiments show that *Echoes* outperforms baselines both on multi-biased and single-biased benchmarks.

REFERENCES

[1] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. 2020. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*. PMLR, 528–539.

[2] Yingqiang Ge, Shuya Zhao, Honglu Zhou, Changhua Pei, Fei Sun, Wenwu Ou, and Yongfeng Zhang. 2020. Understanding echo chambers in e-commerce recommender systems. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 2261–2270.

[3] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence* 2, 11 (2020), 665–673.

[4] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. 2019. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. https://openreview.net/forum?id=Bygh9j09KX

[5] Agrim Gupta, Piotr Dollar, and Ross Girshick. 2019. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5356–5364.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[7] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. 2021. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8340–8349.

[8] Youngkyu Hong and Eunho Yang. 2021. Unbiased classification through bias-contrastive and bias-balanced learning. *Advances in Neural Information Processing Systems* 34 (2021), 26449–26461.

[9] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4401–4410.

[10] Sami Khenissi and Olfa Nasraoui. 2020. Modeling and counteracting exposure bias in recommender systems. *arXiv preprint arXiv:2001.04832* (2020).

[11] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980

[12] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*. 554–561.

[13] Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. 2021. Learning debiased representation via disentangled feature augmentation. *Advances in Neural Information Processing Systems* 34 (2021), 25123–25133.

[14] Jungsoo Lee, Jeonghoon Park, Daeyoung Kim, Juyoung Lee, Edward Choi, and Jaegul Choo. 2023. BiasEnsemble: Revisiting the Importance of Amplifying Bias for Debiasing. (2023).

[15] Zhiheng Li, Ivan Evtimov, Albert Gordo, Caner Hazirbas, Tal Hassner, Cristian Canton Ferrer, Chenliang Xu, and Mark Ibrahim. 2023. A Whac-A-Mole Dilemma: Shortcuts Come in Multiples Where Mitigating One Amplifies Others. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. https://arxiv.org/abs/2212.04825

[16] Zhiheng Li, Anthony Hoogs, and Chenliang Xu. 2022. Discover and Mitigate Unknown Biases with Debiasing Alternate Networks. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIII*. Springer, 270–288.

[17] Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. 2021. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*. PMLR, 6781–6792.

[18] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

[19] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. 2020. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems* 33 (2020), 20673–20684.

[20] Tien T Nguyen, Pik-Mai Hui, F Maxwell Harper, Loren Terveen, and Joseph A Konstan. 2014. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *Proceedings of the 23rd international conference on World wide web*. 677–686.

[21] Shiori Sagawa*, Pang Wei Koh*, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=ryxGuJrFvS

[22] Seonguk Seo, Joon-Young Lee, and Bohyung Han. 2022. Unsupervised learning of debiased representations with pseudo-attributes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16742–16751.

[23] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. 2020. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems* 33 (2020), 9573–9585.

[24] Anique Tahir, Lu Cheng, Ruocheng Guo, and Huan Liu. 2022. Distributional Shift Adaptation using Domain-Specific Features. In *2022 IEEE International Conference on Big Data (Big Data)*. IEEE, 5593–5597.

[25] Enzo Tartaglione, Carlo Alberto Barbano, and Marco Grangetto. 2021. End: Entangling and disentangling deep representations for bias correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 13508–13517.

[26] Damien Teney, Ehsan Abbasnejad, Simon Lucey, and Anton Van den Hengel. 2022. Evading the simplicity bias: Training a diverse set of models discovers solutions with superior ood generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16761–16772.

[27] Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. 2007. On early stopping in gradient descent learning. *Constructive Approximation* 26, 2 (2007), 289–315.

[28] Zhilu Zhang and Mert R. Sabuncu. 2018. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (Eds.). 8792–8802. https://proceedings.neurips.cc/paper/2018/hash/f2925f97bc13ad2852a7a551802feea0-Abstract.html

[29] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence* 40, 6 (2017), 1452–1464.