國立臺灣大學電機資訊學院電信工程學研究所

碩士論文

Graduate Institute of Communication Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

偏差消除之資料增強學習於去偏差化聯邦學習

Bias-Eliminating Augmentation Learning for Debiased Federated Learning

許元譯

Yuan-Yi Hsu

指導教授：王鈺強 博士

Advisor: Yu-Chiang Frank Wang, Ph.D.

中華民國 112 年 07 月

July 2023

doi:10.6342/NTU202301252

國立臺灣大學碩士學位論文
口試委員會審定書

MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

偏差消除之資料增強學習於去偏差化聯邦學習

Bias-Eliminating Augmentation Learning for Debiased Federated Learning

本論文係 許元譯 (r10942094) 在國立臺灣大學電信工程學研究所完成之碩士學位論文，於民國 112 年 7 月 10 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Graduate Institute of Communication Engineering at National Taiwan University on July 10, 2023, has examined a Master's thesis entitled above presented by Yuan-Yi Hsu (r10942094) candidate and hereby certifies that it is worthy of acceptance.

口試委員 Oral examination committee:

王鈺強　　孫紹華

（指導教授 Advisor）

陳祝嵩

系主任/所長 Director: 周錫增

致謝

首先，我要感謝我的指導教授王鈺強老師，您給予我許多寶貴的指導和建議，讓我能夠順利完成我的碩士論文。您的耐心、理解和支持，讓我能夠克服研究過程中遇到的困難，並在學術上取得了新的突破。

接著，我要感謝我的父母，在我的求學過程中，您們一直給予我無條件的支持和鼓勵。您們從小培養我對於知識的渴望與熱情，以及用心的教育及栽培，都是我能完成碩士研究的最大根本。我也要感謝我的哥哥，雖然每次相處時都還是像小時候一樣又吵又鬧，但還是給予我很多成熟的建議。

最後，我要感謝我的同學和朋友，在我的學習和生活中，您們給我許多幫助和支持。特別感謝奐萱不斷給予我支持與鼓勵，讓我有信心克服萬難並完成研究。感謝同甘共苦的實驗室夥伴: 彥仰、宇軒和永玄，給予我非常多研究上的建議。感謝實驗室學長姐: 福恩、棋祥、如芸、萬泉、聖喻和志皓，給予我研究上的幫助以及生涯規劃上的建議。感謝大學好友: 子豪、崇恆、立楷、皓方、亭瑋、冠綸、則宇、有文、元皓、瑋儒，適時陪我放鬆，緩解做研究遇到困難時的壓力。感謝實驗室學弟: 凱博、啟斌、子庭和亦傑，不管在實驗室還是出遊時都帶來歡樂。

在此，我再次感謝所有幫助過我的人，最後祝大家一切順利。

2023.07.25 許元譯

doi:10.6342/NTU202301252

中文摘要

在訓練於具有偏見數據集上的學習模型往往會觀察到類別和不良特徵之間的相關性，導致模型性能下降。大多數現有的去偏差化學習模型是為集中式機器學習而設計的，無法直接應用於保護隱私的分散式設置，如在不同客戶端收集數據的聯邦學習。為了應對具有挑戰性的去偏差化聯邦學習任務，我們提出了一種新穎的聯邦學習框架，稱為偏差消除資料增強學習（FedBEAL），該框架學習使用偏差消除資料增強器（BEA）在每個客戶端生成特定於客戶端的偏差衝突樣本。由於事先不知道偏差類型或屬性，我們提出了一種獨特的學習策略，以共同訓練BEA和提出的聯邦學習框架。我們對具有各種偏差類型的數據集進行了廣泛的圖像分類實驗，以證實FedBEAL的有效性和可應用性，在去偏差化聯邦學習的性能上表現優於最先進的去偏差化方法和聯邦學習方法。

doi:10.6342/NTU202301252

Bias-Eliminating Augmentation Learning for Debiased Federated Learning

Yuan-Yi Hsu

Advisor: Yu-Chiang Frank Wang, Ph.D.

Graduate Institute of Communication Engineering
National Taiwan University
Taipei, Taiwan

July 2023

doi:10.6342/NTU202301252

Abstract

Learning models trained on biased datasets tend to observe correlations between categorical and undesirable features, which result in degraded performances. Most existing debiased learning models are designed for centralized machine learning, which cannot be directly applied to distributed settings like federated learning (FL), which collects data at distinct clients with privacy preserved. To tackle the challenging task of debiased federated learning, we present a novel FL framework of **B**ias-**E**liminating **A**ugmentation **L**earning (**FedBEAL**), which learns to deploy **B**ias-**E**liminating **A**ugmenters (**BEA**) for producing client-specific bias-conflicting samples at each client. Since the bias types or attributes are not known in advance, a unique learning strategy is presented to jointly train BEA with the proposed FL framework. Extensive image classification experiments on datasets with various bias types confirm the effectiveness and applicability of our FedBEAL, which performs favorably against state-of-the-art debiasing and FL methods for debiased FL.

doi:10.6342/NTU202301252

Contents

doi:10.6342/NTU202301252

doi:10.6342/NTU202301252

List of Figures

doi:10.6342/NTU202301252

doi:10.6342/NTU202301252

List of Tables

Chapter 1

Introduction

Deep neural networks have shown promising progress across different domains such as computer vision [1] and natural language processing [2]. Their successes are typically based on the collection of and training on data that properly describe the inherent distribution of the data of interest. However, in real-world scenarios, biased data [3] are often observed during data collection. Biased datasets [4, 5, 6] contain features that are highly correlated to class labels in the training dataset but not sufficiently describing the inherent semantic meaning. Training on such biased data thus result in degraded model generalization capability. Take Fig. 1.1 for example; when addressing the cat-dog classification task, training images collected by users might contain only orange cats and black dogs. Their color attributes are strongly correlated with the image labels during training, but such attributes are not necessarily relevant to the classification task during inference. As pointed out in [5, 6], deep neural networks trained with such biased data are more likely to make decisions based on *bias* attributes instead of semantic attributes. As a result, during inference, performances of the learned models would dramatically drop when observing *bias-conflicting* samples (i.e., data containing semantic and bias attributes that are rarely correlated in the training set).

To tackle the data bias problem, several works have been proposed to remove or alleviate data bias when training deep learning models [3, 7, 8, 9, 10, 11, 12, 13].

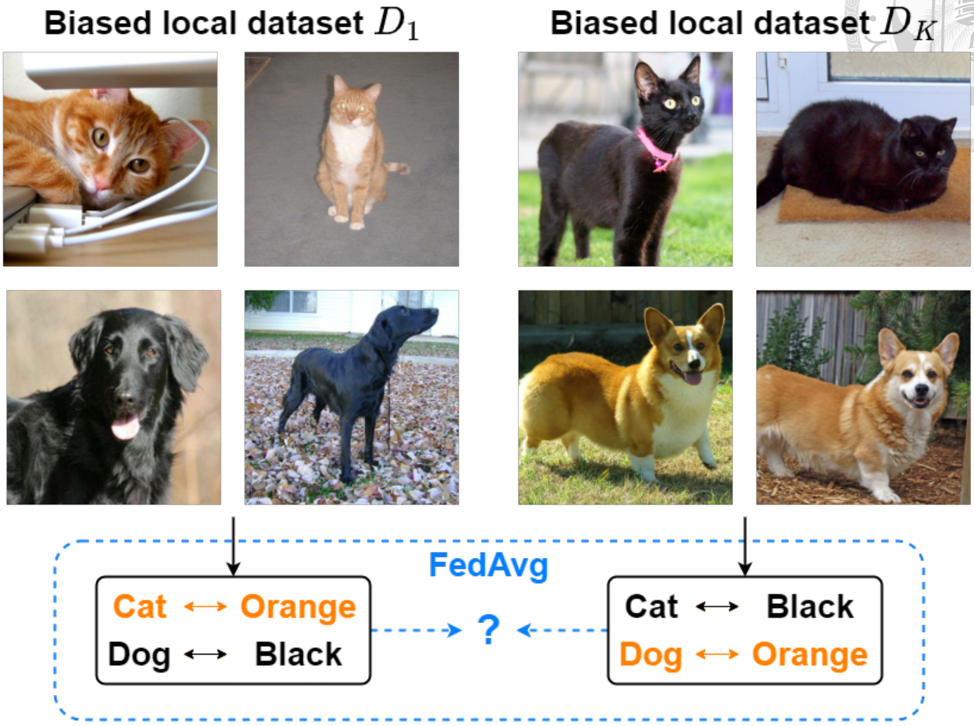


Figure 1.1: **Example of local data bias in FL.** When deploying FL to train a cat-dog classifier with image datasets collected by multiple pet owners, most of the local images are obtained with their pets with specific colors. Therefore, the models trained with each local dataset are likely to establish decision rules on biased attributes (e.g., fur color), which prevents the aggregated model from learning proper representation for classification.

For example, Nam *et al.* [10] train an intentionally biased auxiliary model while enforcing the main model to go against the prejudice of the biased network. Lee *et al.* [11] utilize the aforementioned biased model to synthesize diverse bias-conflicting hidden features for learning debiased representations. Nevertheless, the above techniques are designed for centralized datasets. When performing distributed training of learning models, such methods might fail to generalize.

For distributed learning, federated learning (FL) [14] particularly considers data collection and training conducted at each client, with data privacy needing to be preserved. When considering privately distributed datasets, real-world FL applications

doi:10.6342/NTU202301252

are more likely to suffer data heterogeneity issues [15, 16, 17], i.e., data collected by different clients are not independent and identically distributed (IID). Recently, several works [18, 19, 20, 21, 22, 23, 24] propose to alleviate performance degradation caused by data heterogeneity. However, existing methods typically consider data heterogeneity in terms of label distribution skew [18, 19, 20, 21, 22] or domain discrepancy [23, 24] among clients. These FL methods are not designed to tackle potential data bias across different clients, leaving the debiased FL a challenging task to tackle.

To mitigate the local bias in **Fed**erated learning, we propose a novel FL scheme of **B**ias-**E**liminating **A**ugmentation **L**earning (**FedBEAL**). In FedBEAL, we learn a Bias-Eliminating Augmenter (BEA) for each client, with the goal of producing bias-conflicting samples. To identify and introduce the desirable semantic and bias attributes to the augmented samples, our FedBEAL uniquely adopts the global server model and each client model trained across iterations without prior knowledge of bias type or annotation. With the introduced augmenter and the produced bias-conflicting samples, debiased local updates can be performed at each client, followed by simple aggregation of such models for deriving the server model.

We now summarize the contributions of this work below:

- To the best of our knowledge, We are among the first to tackle the problem of debiased federated learning, in which local yet distinct biases exist at the client level.

- We present FedBEAL for debiased FL, which introduces Bias-Eliminating Augmenters (BEA) at each client with the goal of generating bias-conflicting samples to eliminate local data biases.

- Learning of BEA can be realized by utilizing the global server and local client models trained across iterations, which allows us to identify and embed desirable semantic and bias features for augmentation purposes.

doi:10.6342/NTU202301252

Chapter 2

Related Work

2.1 Debiasing in Centralized Machine Learning

With the presence of biased datasets, neural networks are prone to relying on simpler features (e.g., color information) and remaining invariant to other predictive complex features [5, 6] (e.g., semantic information), which limit the performances of the learned models. Several works [25, 7, 26, 9] propose debiasing techniques to improve the robustness of the model trained on such biased datasets. However, they either assume the bias type to be known (e.g., texture bias) in advanced [7] or require auxiliary annotations of the bias attributes (e.g., color information) for each sample [25, 26, 9]), which might not be practically available. To alleviate this concern, some research works [10, 11, 12, 27] focus on mitigating dataset biases without presuming bias categories or involving additional annotations. For instance, Nam *et al.* [10] train a biased model by repeatedly amplifying its prejudice. Based on the assumption that biased models fail on bias-conflicting samples, they further upweight the bias-conflicting samples so that a debiased model can be trained accordingly. Lee *et al.* [11] follow the above approach to debias the main model by disentangling the semantic and bias features. On the other hand, Hong *et al.* [12] apply contrastive learning [28, 29] to encourage intra-bias feature dissimilarities. Although the above methods have shown promising performances, they are mainly

doi:10.6342/NTU202301252

applicable to centralized learning schemes. For distributed learning like federated learning, these methods cannot be directly applied.

2.2 Federated Learning with Data Heterogeneity

Label distribution skew.. Under the heterogeneous label distribution, existing methods [18, 19, 20, 21, 22, 30, 31] focus on correcting client drift using global information. For example, FedProx [18] adds a regularization term to preserve consistency between local updates and the global model. SCAFFOLD [19] mitigates gradient dissimilarity using control variates. MOON [21] addresses non-IID problems by applying contrastive learning at the model level.

Distribution shift across clients.. As for feature distribution drift (also known as domain shift), previous FL works [23, 24] are designed to bridge the domain gap between different clients. For instance, FedBN [23] choose to fix the parameters for local Batch Normalization and do not synchronize them with the global model. As for FCCL [24], it views domain shift as a catastrophic forgetting problem and approaches it by using knowledge distillation techniques.

Debiased federated learning.. Recently, a number of FL works [32, 33, 34, 35, 36] are proposed to eliminate *local biases* from the training data. In [33, 34], such biases are referred to as label distribution skew. For example, [33] uses the term *local learning bias* to describe decision boundaries discrepancy among networks trained on heterogeneous data. As for [35, 36], additional efforts are made to take care of underprivileged or sensitive data subsets (e.g., racial, gender groups). For example, Ezzeldin *et al.* [36] propose a fairness-aware FL framework for preventing the trained model from being biased toward an underlying demographic group, aiming to produce a fair model across clients while maintaining high utility. It can be seen that we are among the first to address the learning task of *debiased federated learning*, in which undesirable correlations of bias attributes and class

doi:10.6342/NTU202301252

labels are observed at each client.

doi:10.6342/NTU202301252

Chapter 3

Proposed Method

3.1 Problem Definition and Method Overview

Problem definition. For the sake of completeness, we first define the problem setting and notations used in this paper. We assume that training image data are privately distributed in K clients $D = \{D_1, D_2, ..., D_K\}$, each containing a set of image-label pairs $D_k = \{(x, y) \mid P_k(X = x, Y = y)\}$. To formulate local data biases, we follow Hong *et al.* [12] and assume that images X can be decomposed into semantic attributes A_{sem} and hidden bias attributes A_{bias}. Note that A_{sem} is expected to describe categorical information, while A_{bias} contains undesirable features highly correlated with Y. As depicted in 1.1, we assume each client with disparate bias-label correlations (i.e., $\forall_{k \neq k'} P_k(Y|A_{bias}) \neq P_{k'}(Y|A_{bias})$). On the other hand, since this work focuses on mitigating local client bias instead of the bias of the global dataset D, we assume the union of all local training datasets shares the same bias distribution with the test dataset D_{test} (i.e., $P(Y|A_{bias}) = P_{test}(Y|A_{bias})$). With a total of T communication rounds, the goal of debiased FL is to derive a model f that satisfies

$$\arg\min_{f} \Sigma_{k=1}^{K} \frac{|D_k|}{|D|} \mathcal{L}_k(f), \tag{3.1}$$

where $\mathcal{L}_k(f) = \mathbb{E}_{(x,y)\sim D_k}[\ell(f(x), y)]$ represents the empirical loss of client k.

doi:10.6342/NTU202301252

Method overview. Based on FedAvg [14], our proposed Bias-Eliminating Augmentation Learning (**FedBEAL**) trains a network robust to data bias observed at each client. Similar to standard FL, training of FedBEAL requires alternative optimization between the two stages. More specifically, *debiased local update* is performed at the client side, and *global aggregation* is conducted at the server side. To address local bias problems, we uniquely propose to learn a **B**ias-**E**liminating **A**ugmenter (**BEA**) g_k for each client k. As depicted in 3.1, BEA is deployed to generate bias-conflicting samples and allows updates of each f_k. As for the global aggregation stage, each f_k will be uploaded to the server for producing a debiased global model f. We now detail our proposed learning scheme below.

3.2 Bias-Eliminating Augmenter

To eliminate the local bias in FL, we propose to deploy Bias-Eliminating Augmenters at each client. Since the bias information is unknown, how to design BEA for creating bias-conflicting samples within each local client would be challenging. With local image data and their class labels observed, we now explain how our BEA can be learned in an FL scheme.

3.2.1 Design and architecture

As depicted in 3.2, for each client k, we randomly sample two samples x^i and x^j with *distinct* labels from the local dataset D_k. Inspired by recent mixed sample data augmentation (MSDA) techniques [37, 38, 39, 40, 41, 42, 43], we produce the mixed bias-conflicting sample $\tilde{x}$ by utilizing U-Net as the backbone, with a modulator $M \in [0,1]^{H \times W \times 3}$ deployed. With the concatenation of x^i and x^j as the input to BEA, the output $\tilde{x}$ can be expressed as:

$$\tilde{x} = M \odot x^i + (1 - M) \odot x^j, \tag{3.2}$$

where $\odot$ indicates the element-wise multiplication, and we have $\tilde{y} = y^i$ for the manipulated output (i.e., the class label of $\tilde{x}$ is identical to that of x^i).

For $\tilde{x}$ being a bias-conflicting example, it would be desirable for $\tilde{x}$ to share the same semantic attribute with x^i (i.e., $\tilde{a}_{sem}$ of $\tilde{x}$ to be closed to a^i_{sem} of x^i), while sharing the same bias attribute with x^j (i.e., $\tilde{a}_{bias}$ of $\tilde{x}$ to be closed to a^j_{bias} of x^j). Once such bias-conflicting samples are obtained, one can train the associated client model and update the global model accordingly, which is expected to produce debiased representations.

3.2.2 Learning of BEA

Without prior knowledge of bias types, providing guidance to train the BEA would not be straightforward. In order to have BEA identify desirable intrinsic semantic and inherent bias attributes for manipulating bias-conflicting samples, we propose a unique learning scheme utilizing the global server model f^t and local client model f_k^{t-1}.

Extracting semantic attributes via unbiased global prediction.. For a bias-conflicting sample $\tilde{x}$, its semantic attribute $\tilde{a}_{sem}$ is expected to be similar to a^i_{sem} of x^i. In FL, since the global server model f^t is derived by global aggregation, f^t can be considered relatively unbiased when compared to the local model f_k^{t-1} produced at the previous iteration. Thus, it would be desirable for $\tilde{a}_{sem}$ and a^i_{sem} to exhibit large similarity, which can be derived from the difference between the predictions of $\tilde{x}$ and x^i derived from the global model f^t. To be precise, the loss function for encouraging such semantic attribute consistency is defined as:

$$\mathcal{L} = d_{KL}(f^t(\tilde{x}), f^t(x^i)), \tag{3.3}$$

where d_{KL} calculates the KL divergence between the predictions using f^t.

Producing bias attributes via biased local prediction.. On the other hand, for a bias-conflicting sample $\tilde{x}$, its bias attribute $\tilde{a}_{bias}$ is expected to be similar to a^j_{bias} of x^j, which is sampled from an instance from a different category (as described in 3.2.1).

To identify and relate such bias attributes, we take the local client model f_k^{t-1} as the guidance. Note that, compared to the aggregated server model, client models produced at prior iterations are considered to be affected more by local biased data, which is more likely to predict the output $f_k^{t-1}(x)$ based on its hidden bias attributes. Therefore, we define the similarity between the bias attributes $\tilde{a}_{bias}$ and a^j_{bias}, which is now calculated and guided by the difference between the predictions of $\tilde{x}$ and x^i using f_k^{t-1}. Specifically, we minimize:

$$\mathcal{L}_k = d_{KL}(f_k^{t-1}(\tilde{x}), f_k^{t-1}(x^j)), \tag{3.4}$$

where $(t-1)$ denotes the training round.

From the above design and derivation, we have the objective for training BEA as:

$$\mathcal{L}_{total} = \mathcal{L} + \mathcal{L}_k. \tag{3.5}$$

As depicted in 3.2, via minimization of $\mathcal{L}$, BEA will be optimized so that the semantic attribute $\tilde{a}_{sem}$ of $\tilde{x}$ will be updated and be close to a^i_{sem} of x^i. On the other hand, minimizing $\mathcal{L}_k$ encourages the bias attribute $\tilde{a}_{bias}$ of $\tilde{x}$ to be updated as a^j_{bias} of x^j. In other words, optimization of BEA would encourage the generated samples whose semantic and bias attributes are extracted from training data of distinct classes.

While our BEA can be viewed as performing mixed-sample data augmentation, existing MSDA methods [37, 38, 39, 40, 41, 42, 43] are only designed to produce handcrafted augmentation outputs, which may not necessarily to be bias-conflicting. For example, spatial location-based augmentations (e.g., CutMix [37], FMix [38]) only fuse two images by replacing a region of one image with that from another, alleviating only high-level bias (e.g., background bias [10]). Style-based augmentations [40, 39, 43] are only capable of alleviating low-level biases by mixing style and content from distinct images. As verified in 4, learning of BEA would be desirable for debiased FL.

doi:10.6342/NTU202301252

Algorithm 1: Training of FedBEAL

```
**Input:** $T$, $T_w$, $K$, $D = \{D_1, D_2, ..., D_k\}$, $p$, $g_1, g_2, ..., g_K$, $f^0$, local epochs $E_g$ and $E_f$,
        learning rate $\eta_g$ and $\eta_f$
for $t = 0, 1, ..., T-1$ do
    for $k = 1, 2, ..., K$ in parallel do
        if $t \geq T_w$ then
            TrainBEA($f^t, f_k^{t-1}$)
        end
        $f_k^t \leftarrow$ LocalUpdate($f^t$)
        $f^{t+1} \leftarrow \Sigma_{k=1}^{K} \frac{|D_k|}{|D|} f_k^t$
    end
end
**Output:** return $f^T$
TrainBEA($f^t, f_k^{t-1}$)
for $e = 1, 2, ..., E_g$ do
    for $(x^i,\ x^j)$ of $D_k$ do
        $\tilde{x} \leftarrow g_k(x^i, x^j)$
        $L \leftarrow d_{KL}(f^t(\tilde{x}), f^t(x^i))$
        $L_k \leftarrow d_{KL}(f_k^{t-1}(\tilde{x}), f_k^{t-1}(x^j))$
        $L_{total} \leftarrow L + L_k$
        $g_k \leftarrow g_k - \eta_g \nabla L_{total}$
    end
end
LocalUpdate($f^t$)
$f_k^t \leftarrow f^t$
for $e = 1, 2, ..., E_f$ do
    for $(x^i,\ x^j,\ y^i,\ y^j)$ of $D_k$ do
        if $t \geq T_w$ and $Uniform(0,1) < p$ then
            $\tilde{x},\ \tilde{y} \leftarrow g_k(x^i, x^j),\ y^i$
            $L_{cls} \leftarrow CrossEntropy(f_k^t(\tilde{x}), \tilde{y})$
        end
        else
            $L_{cls} \leftarrow CrossEntropy(f_k^t(x^i), y^i)$
        end
        $f_k^t \leftarrow f_k^t - \eta_f \nabla L_{cls}$
    end
end
return $f_k^t$
```

3.3 Training of FedBEAL

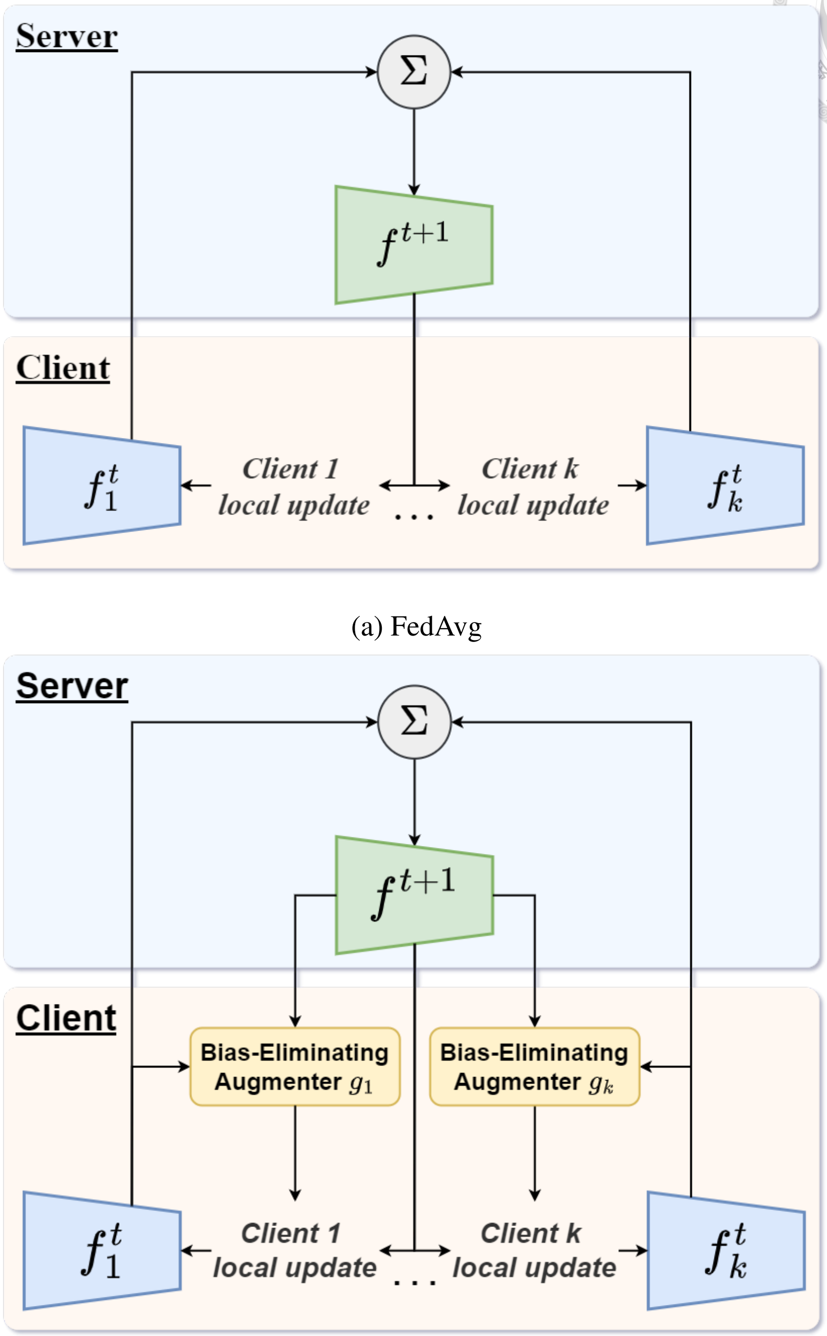
Debiased local update.. After BEA is learned and deployed at each client k, we perform debias local updates by training each local model f_k^t using additionally produced bias-conflicting data pairs (i.e., $\tilde{x}$ and $\tilde{y}$). To further improve the robustness of our framework, we follow [37] to consider several techniques at this local update stage. That is, we define $p \in [0, 1]$ as the probability of augmenting each data batch to control the degree of debiasing. Moreover, we define the warm-up round T_w (i.e., BEA is activated after round T_w) to avoid undesirable augmentation outputs harmful to local training happening in the beginning stage. With bias-conflicting data and the introduced learning techniques, we are able to enforce the local model to be better guided by the semantic information while suppressing the bias.

Global aggregation.. For each training iteration, once the debiased local updates are performed, we then collect and aggregate the learned weights of each local model (weighted by the size of the corresponding local dataset [14]). To be more specific, the global model for the next round f^{t+1} can be calculated as follows:

$$f^{t+1} = \Sigma_{k=1}^{K} \frac{|D_k|}{|D|} f_k^t. \tag{3.6}$$

With the convergence of the overall training process, the final global model can be applied to perform classification on unbiased test data. The pseudo-code of our complete FedBEAL framework is summarized in 34.

doi:10.6342/NTU202301252



(a) FedAvg

(b) FedBEAL

Figure 3.1: **Comparisons between (a) FedAvg and (b) FedBEAL.** Our FedBEAL learns Bias-Eliminating Augmenters (BEA) to produce bias-conflicting samples at each client, allowing the learned model to produce improved debiased representations.

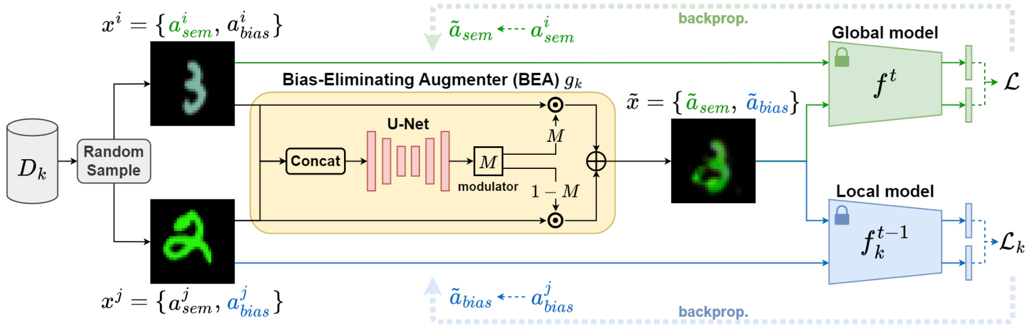

$x^i = \{a^i_{sem}, a^i_{bias}\}$
$\tilde{a}_{sem} \leftarrow a^i_{sem}$
backprop.
Global model
f^t
$\mathcal{L}$
Bias-Eliminating Augmenter (BEA) g_k
$\tilde{x} = \{\tilde{a}_{sem}, \tilde{a}_{bias}\}$
U-Net
M
D_k
Random Sample
Concat
M
modulator
$1 - M$
Local model
f^{t-1}_k
$\mathcal{L}_k$
$x^j = \{a^j_{sem}, a^j_{bias}\}$
$\tilde{a}_{bias} \leftarrow a^j_{bias}$
backprop.

Figure 3.2: **Design and learning of Bias-Eliminating Augmenter.** Given two randomly selected images x^i and x^j at client k, the Bias-Eliminating Augmenter (BEA) learns to produce a bias-conflicting sample $\tilde{x}$. That is, the semantic attribute a_{sem} of $\tilde{x}$ is expected to be close to that of x^i, while the bias attribute a_{bias} of $\tilde{x}$ would be extracted from x^j. Note that f^t and f^{t-1}_k denote the server and client models learned at t-th and $(t-1)$-th iterations, respectively.

Chapter 4

Experiments

4.1 Datasets and Implementation Details

Datasets.. To evaluate the effectiveness and applicability of our learning scheme in different types of bias, we consider three commonly used biased datasets, including Colored MNIST [44] (with color bias), Corrupted CIFAR-10 [45] (with corruption bias), and Collage CIFAR-10 [27] (with collaged images as bias). Colored MNIST contains images of hand-written digits colorized with different colors. Corrupted CIFAR-10 includes images applied with random corruptions (e.g., noises, blurring, brightness/contrast adjustment). In Collage CIFAR-10, a sample is combined with four images originating from four different datasets, including MNIST [46], Fashion MNIST [47] and SVHN [48] that jointly serve as bias attributes, and CIFAR-10 [49] as the semantic information. As noted in 3.1, we distribute the training set to K clients, where K is set to 10 across all our experiments. To quantify the severity of local bias in training data, we further define the ratio for the amount of biased local data β.

Implementation details.. For Colored MNIST, Corrupted CIFAR-10, and Collage CIFAR-10, input images are resized to 28×28, 32×32, and 64×64 pixels. For simplicity, we use LeNet [46] as the classifier f for Colored MNIST and

doi:10.6342/NTU202301252

ResNet-18 [1] for Corrupted CIFAR-10 and Collage CIFAR-10. A U-Net [50] with the encoder of ResNet-18 is adopted as the augmenter g. The communication round T is set to 100. For each round, each client train their g and f sequentially for 5 epochs using the SGD optimizer, with the batch size of 64, the learning rate of 0.01, the momentum of 0.9, and the weight decay of 0.00001. We implement our model using PyTorch, and conduct training on a single NVIDIA 3090 GPU with 24GB memory.

4.2 Quantitative Evaluation

4.2.1 Comparisons to debiasing and FL methods

We first compare proposed learning scheme with existing centralized debiasing [10, 12, 11] and heterogeneous federated learning [14, 18, 19, 21, 23] methods. In our experiments, SOLO and FedAvg [14] are viewed as baselines. The former only performs local training without global averaging of client models, while the latter is the fundamental framework for all the other methods reported in this section. Note that we report the mean accuracy of each local model in SOLO. As shown in 4.1, we evaluate state-of-the-art methods with the three datasets with β set from 0.99 to 0.999. From the upper half of 4.1, we applied existing debiasing methods designed for centralized machine learning [10, 12, 11] to debias local update at each client. For example, SoftCon [12] enabled each client to preserve intra-bias features dissimilarities to debias the model, which improved the results of Colored MNIST with β of 0.999 by 13.72%. On the other hand, from the bottom half of 4.1, existing FL approaches tackled data heterogeneity by preserving the consistency between the local and global models. It can be observed that such constraints were not sufficient to mitigate severe local bias and only slightly improved the performance (e.g., FedProx [18] improved the accuracy by 0.4% on Colored MNIST with β of 0.999). Instead, our FedBEAL performed favorably against the above methods on all datasets (e.g., accuracy improvements

doi:10.6342/NTU202301252

of 19.32% on Colored MNIST with β of 0.999). These quantitative comparisons verify that our proposed FL approach removes local biases across different clients for improved classification performances.

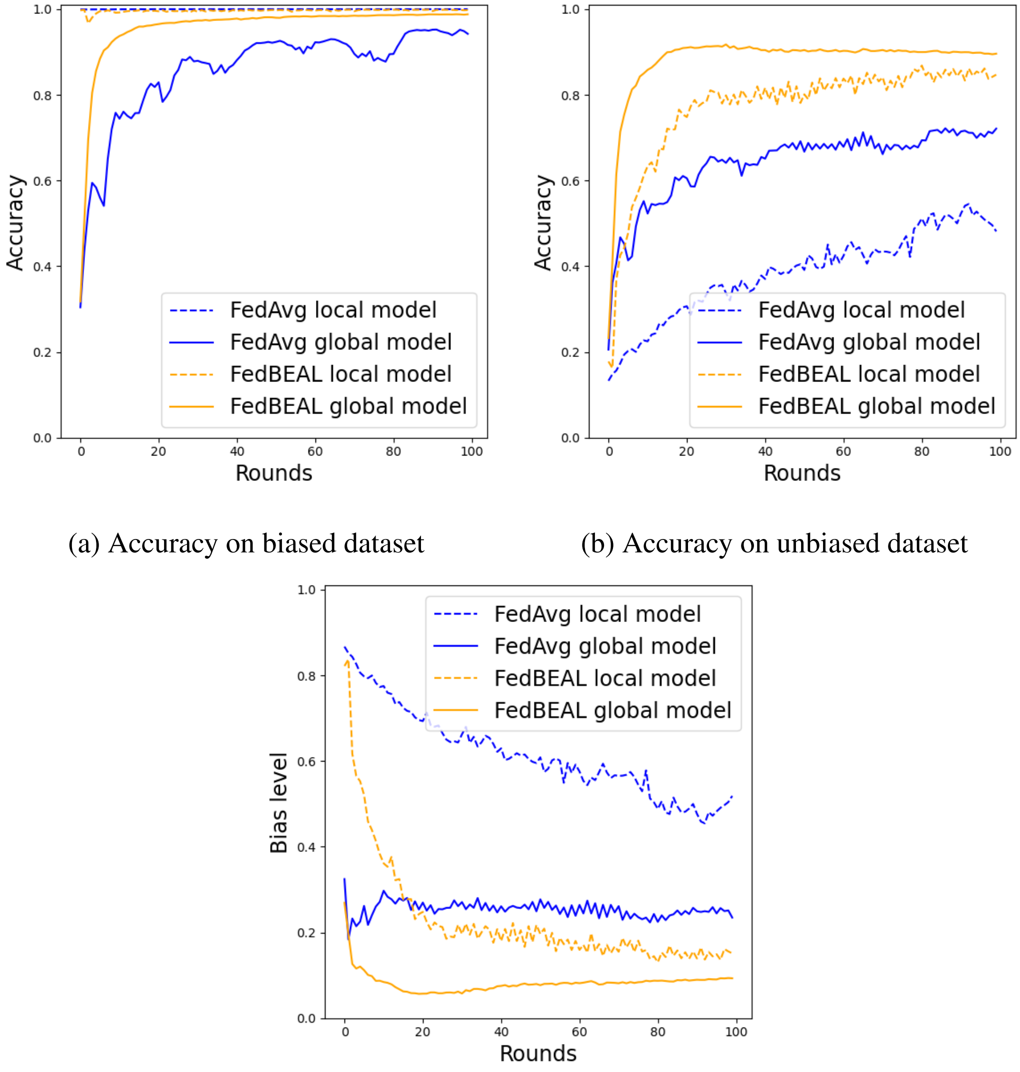
4.2.2 Comparisons to MSDA methods

To further verify the effectiveness of our augmentation scheme, we further compare our method with state-of-the-art mixed sample data augmentation algorithms [51, 37, 40]. Existing handcrafted MSDA methods are generally designed to handle particular types of bias and cannot easily generalize to bias types not defined in advance. As shown in 4.2, MixStyle [40] benefited low-level biases (e.g., color or corruption bias) by transferring style information of the images and improved the accuracies from 5.99% to 26.53% on Colored MNIST and Corrupted CIFAR-10. However, such augmentations was not able to mitigate high-level biases (e.g., background bias [10]), as the performance of MixStyle dropped from 2.82% to 3.18% on Collage CIFAR-10. On the other hand, CutMix significantly improved the accuracy by 27.07% on Collage CIFAR-10 with β of 0.999 since the cut-and-paste operation efficiently removed high-level regional bias. However, it failed to handle low-level color and corruption biases in Colored MNIST and Corrupted CIFAR-10 and degraded the performance from 7.64% to 13.12%. Compared to such MSDA methods, our approach learns to find the optimal BEA and thus exhibits more robust debiasing effects on various bias types. As shown in the last column of 4.2, our method performed favorably compared to MSDA approaches and achieved improved accuracy by 14.53%, indicating the robustness and generalization capability of our augmentation scheme to different bias types.

4.2.3 Debiasing server and client models

As indicated in 3.2.2, the design and learning objectives for our BEA are based on the assumption that local models are relatively biased compared to the global aggregated one at each iteration. To verify this assumption, we quantitatively

doi:10.6342/NTU202301252



(a) Accuracy on biased dataset

(b) Accuracy on unbiased dataset

(c) Bias level

Figure 4.1: **Learning curve comparisons of global/local models from FedAvg and FedBEAL on Colored MNIST.** (a) and (b) show accuracies for biased and unbiased datasets, and (c) compares the bias level $\mathcal{S}$ (see 4.1). Note that the local model with FedBEAL shows improved debiased performance, and its global model also exhibits improved unbias ability over FedAvg.

doi:10.6342/NTU202301252

compare the *bias level* of the global and local models in FedAvg and FedBEAL on the Colored MNIST dataset. Given a biased dataset D_k from client k and an unbiased testing dataset D_{test}, we first define the bias level $\mathcal{S}$ of the local model f_k^t and the global model f^t as follows:

$$\mathcal{S} = 1 - \frac{Acc_{unbias}}{Acc_{bias}}, \tag{4.1}$$

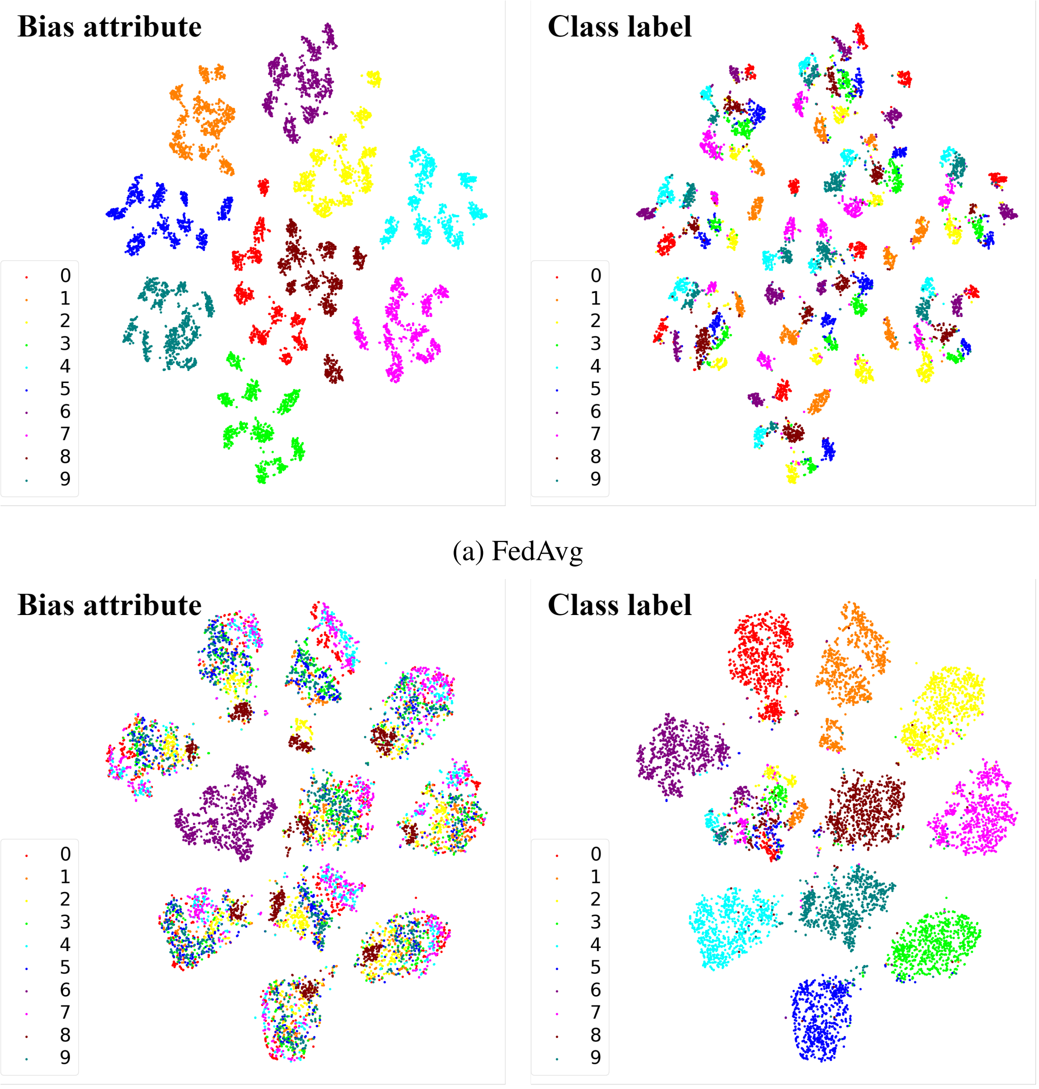
where Acc_{bias} and Acc_{unbias} are the accuracies evaluated on D_k and D_{test}, respectively. In other words, the model is biased (i.e., S is higher) if the model achieves high accuracy on the biased dataset while performing relatively unfavorable on the unbiased dataset.

Based on the above setting, we train our model on Colored MNIST with the bias ratio β of 0.999. As illustrated in 4.1c, while the local model of FedBEAL was relatively biased compared to the global model (see orange curves), we were able to gradually debias such models for improved performances when comparing to FedAvg. The above results support the design and learning scheme for the proposed BEA.

4.3 Qualitative Evaluation

Representation visualization.. We now qualitatively assess the ability of FedBEAL to derive semantic-aware and debiased feature representations. As shown in 4.2, we apply t-SNE [52] to compare the hidden representation derived by the global model of FedAvg and our approach on Colored MNIST. In 4.2a, we see that features extracted by FedAvg were grouped according to *bias* attributes and were not properly separated with respect to the class labels. In contrast, features derived by our model remained relatively uncorrelated in terms of the bias attributes, and the separation between different class clusters was more significant. The above observation indicates that our proposed bias-eliminating augmentation learning allows the derivation of discriminative and debiased features.

doi:10.6342/NTU202301252



(a) FedAvg

(b) FedBEAL

Figure 4.2: **t-SNE comparisons between FedAvg and FedBEAL on Colored MNIST.** Data points in the left column are colorized based on the bias attributes (i.e., color), while those in the right column are colorized based on the class labels.

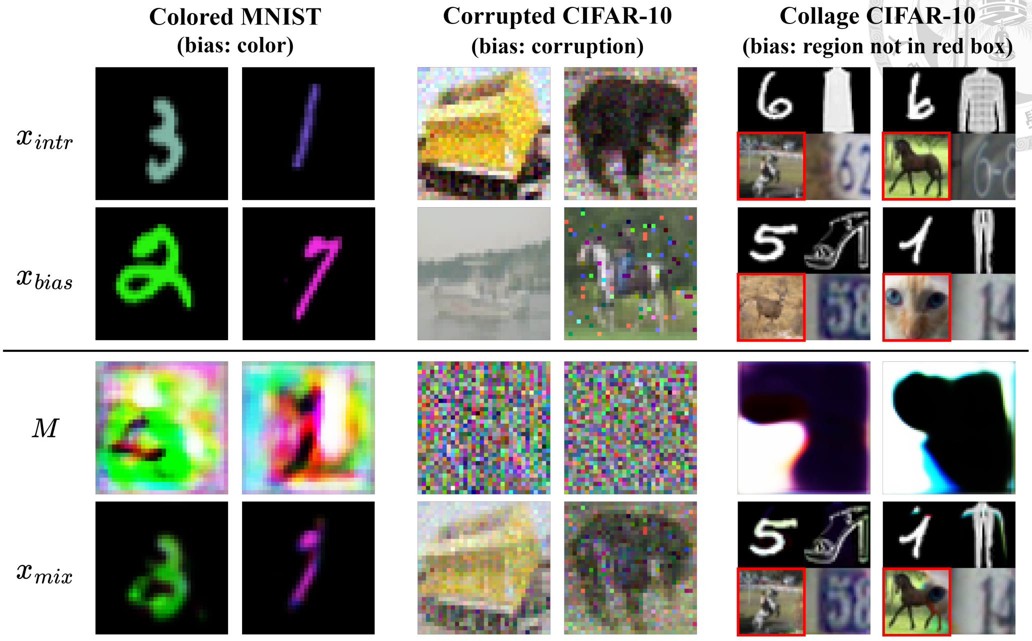
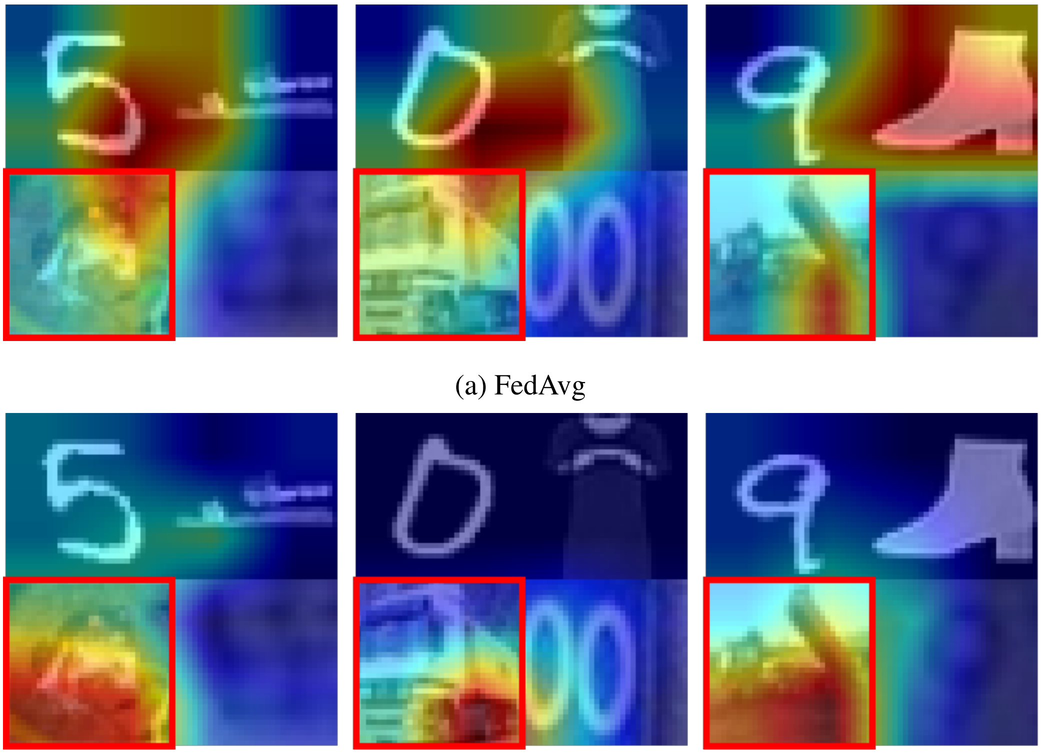


Figure 4.3: **Visualization of images produced by BEA.** Based on the mask learned by the BEA modulator M, the augmented bias-conflicting $\tilde{x}$ can be seen as the mixture of the content from x^i and the bias from x^j.

Visualization of augmented bias-conflicting samples.. We now show example augmented images $\tilde{x}$ produced by our method, which is expected to preserve the categorical information of x^i and impose the bias from x^j. In 4.3, we first see example images for Colored MNIST, and we observe $\tilde{x}$ obtained the digit color from x^j while preserving the original digit shape as of x^i. From the second image set of Corrupted CIFAR-10, $\tilde{x}$ inherited the high chromatic *impluse noise* from x^j while still maintaining semantically recognizable foreground objects. As for Collage CIFAR-10, our modulators M successfully captured the unbiased bottom-left image region for augmentation. From the examples, we confirm that our proposed BEA is capable of capturing inherent dataset bias while preserving desirable semantic attributes for augmenting bias-conflicting samples.

Grad-CAM visualization.. Grad-CAM [53] is commonly used to visually explain how deep learning models make classification decisions. To verify the effec-



(a) FedAvg

(b) FedBEAL

Figure 4.4: **Grad-CAM comparisons between FedAvg and FedBEAL on Collage CIFAR-10.** Compared to FedAvg results in (a), the attention map for FedBEAL in (b) better identify the object region of interest for classification (in red rectangles).

doi:10.6342/NTU202301252

tiveness of the proposed learning scheme, we consider FedAvg and our proposed method on the Collage CIFAR-10 dataset with β of 0.99, and we apply Grad-CAM to interpret the trained global models during classification (see 4.4). From 4.4a, we see that the global model trained with FedAvg attended to ambiguous or irrelevant image regions, implying the lack of ability to indicate regions with proper semantic features for classification. In 4.4a), we see that the global model trained by our proposed FedBEAL attended image regions on the augmented samples, which are correlated to the categorical information of interest. This also explains the reason why our FedBEAL is able to achieve satisfactory performances on debaised FL tasks.

doi:10.6342/NTU202301252

Dataset	Colored MNIST		Corrupted CIFAR-10		Collage CIFAR-10	
Bias ratio β	0.99	0.999	0.99	0.999	0.99	0.999
Baselines						
SOLO	46.90	14.46	16.80	13.19	12.28	10.58
FedAvg [14]	93.90	72.67	49.03	40.28	52.93	36.91
Centralized Debiasing Methods						
LfF [10]	87.64	55.27	53.47	42.25	46.53	26.96
SoftCon [12]	96.75	86.39	55.38	47.61	54.19	42.98
Lee *et al.* [11]	90.28	61.35	54.86	45.90	41.02	22.58
Data Heterogeneous Federated Learning						
FedProx [18]	94.51	73.07	44.06	34.01	41.87	25.94
SCAFFOLD [19]	95.01	68.41	41.73	34.35	38.37	33.85
MOON [21]	93.33	69.37	36.79	26.06	34.71	19.97
FedBN [23]	N/A	N/A	48.46	36.52	46.51	32.53
Ours	**98.58**	**91.99**	**59.18**	**49.09**	**69.53**	**64.53**

Table 4.1: **Comparisons to SOTA federated learning and debiasing algorithms. Bold** denotes the best result, while underline denotes the second best. Note that in Colored MNIST, FedBN is not applicable due to disregard of Batch Normalization layers.

doi:10.6342/NTU202301252

Dataset	Colored MNIST		Corrupted CIFAR-10		Collage CIFAR-10		Avg.
Bias ratio β	0.99	0.999	0.99	0.999	0.99	0.999	
FedAvg [14]	93.90	72.67	49.03	40.28	52.93	36.91	57.62
Mixup [51]	91.38	74.76	53.98	40.85	50.13	37.65	58.13
CutMix [37]	82.73	59.55	41.39	31.69	**71.26**	63.98	58.43
MixStyle [40]	**99.13**	**99.20**	58.99	46.27	49.75	34.09	64.57
Ours	98.58	91.99	**59.18**	**49.09**	69.53	**64.53**	**72.15**

Table 4.2: **Comparisons to MSDA methods for debiased FL. Bold** denotes the best result, while underline denotes the second best.

doi:10.6342/NTU202301252

Chapter 5

Conclusion

In this paper, we addressed the challenging problem of debiased FL and proposed FedBEAL for mitigating local biases. By introducing and learning Bias-Eliminating Augmenters at each client, bias-conflicting samples can be automatically learned. The learning of BEA can be simply utilized by the global server and local client models obtained during the training progress, and thus no prior knowledge of bias type or annotation would be required. We conducted extensive experiments, including comparisons to state-of-the-art debiasing, FL, and MSDA methods, and visualization of augmented images, which quantitatively and qualitatively confirmed the effectiveness and robustness of our proposed approach in discovering and solving unknown dataset bias in federated learning schemes.

doi:10.6342/NTU202301252

Reference

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. 1, 16

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018. 1

[3] B. Kim, H. Kim, K. Kim, S. Kim, and J. Kim, "Learning not to learn: Training deep neural networks with biased data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9012–9020. 1

[4] K. Karkkainen and J. Joo, "Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, January 2021, pp. 1548–1558. 1

[5] H. Shah, K. Tamuly, A. Raghunathan, P. Jain, and P. Netrapalli, "The pitfalls of simplicity bias in neural networks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9573–9585, 2020. 1, 4

[6] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann, "Shortcut learning in deep neural networks," *Nature Machine Intelligence*, vol. 2, no. 11, pp. 665–673, 2020. 1, 4

doi:10.6342/NTU202301252

[7] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness." in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Bygh9j09KX 1, 4

[8] Y. Li and N. Vasconcelos, "Repair: Removing representation bias by dataset resampling," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9572–9581. 1

[9] H. Bahng, S. Chun, S. Yun, J. Choo, and S. J. Oh, "Learning de-biased representations with biased representations," in *International Conference on Machine Learning*. PMLR, 2020, pp. 528–539. 1, 4

[10] J. Nam, H. Cha, S. Ahn, J. Lee, and J. Shin, "Learning from failure: De-biasing classifier from biased classifier," *Advances in Neural Information Processing Systems*, vol. 33, pp. 20 673–20 684, 2020. 1, 2, 4, 10, 16, 17, 24

[11] J. Lee, E. Kim, J. Lee, J. Lee, and J. Choo, "Learning debiased representation via disentangled feature augmentation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 25 123–25 133, 2021. 1, 2, 4, 16, 24

[12] Y. Hong and E. Yang, "Unbiased classification through bias-contrastive and bias-balanced learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 26 449–26 461, 2021. 1, 4, 7, 16, 24

[13] S. Sagawa*, P. W. Koh*, T. B. Hashimoto, and P. Liang, "Distributionally robust neural networks," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=ryxGuJrFvS 1

[14] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data,"

in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282. 2, 8, 12, 16, 24, 25

[15] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, “Advances and open problems in federated learning,” *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021. 3

[16] Q. Li, Y. Diao, Q. Chen, and B. He, “Federated learning on non-iid data silos: An experimental study,” in *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2022, pp. 965–978. 3

[17] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, “Federated learning with non-iid data,” *arXiv preprint arXiv:1806.00582*, 2018. 3

[18] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated optimization in heterogeneous networks,” *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020. 3, 5, 16, 24

[19] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, “Scaffold: Stochastic controlled averaging for federated learning,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 5132–5143. 3, 5, 16, 24

[20] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, “Tackling the objective inconsistency problem in heterogeneous federated optimization,” *Advances in neural information processing systems*, vol. 33, pp. 7611–7623, 2020. 3, 5

[21] Q. Li, B. He, and D. Song, “Model-contrastive federated learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 713–10 722. 3, 5, 16, 24

[22] M. Luo, F. Chen, D. Hu, Y. Zhang, J. Liang, and J. Feng, “No fear of heterogeneity: Classifier calibration for federated learning with non-iid data,”

doi:10.6342/NTU202301252

Advances in Neural Information Processing Systems, vol. 34, pp. 5972–5984, 2021. 3, 5

[23] X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou, "Fedbn: Federated learning on non-iid features via local batch normalization," *arXiv preprint arXiv:2102.07623*, 2021. 3, 5, 16, 24

[24] W. Huang, M. Ye, and B. Du, "Learn from others and be yourself in heterogeneous federated learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 143–10 153. 3, 5

[25] H. Wang, Z. He, Z. C. Lipton, and E. P. Xing, "Learning robust representations by projecting superficial statistics out," *arXiv preprint arXiv:1903.06256*, 2019. 4

[26] E. Tartaglione, C. A. Barbano, and M. Grangetto, "End: Entangling and disentangling deep representations for bias correction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 13 508–13 517. 4

[27] D. Teney, E. Abbasnejad, S. Lucey, and A. van den Hengel, "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*, 2022, pp. 16 761–16 772. 4, 15

[28] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738. 4

[29] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 661–18 673, 2020. 4

doi:10.6342/NTU202301252

[30] Z. Zhu, J. Hong, and J. Zhou, “Data-free knowledge distillation for heterogeneous federated learning,” in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 12 878–12 889. [Online]. Available: https://proceedings.mlr.press/v139/zhu21b.html 5

[31] T. Lin, L. Kong, S. U. Stich, and M. Jaggi, “Ensemble distillation for robust model fusion in federated learning,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 2351–2363, 2020. 5

[32] D. A. E. Acar, Y. Zhao, R. Zhu, R. Matas, M. Mattina, P. Whatmough, and V. Saligrama, “Debiasing model updates for improving personalized federated training,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 21–31. 5

[33] Anonymous, “Learning to aggregate: A parameterized aggregator to debias aggregation for cross-device federated learning,” in *Submitted to The Eleventh International Conference on Learning Representations*, 2023, under review. [Online]. Available: https://openreview.net/forum?id=IQM-3_Tzldw 5

[34] Y. Guo, X. Tang, and T. Lin, “Feddebias: Reducing the local learning bias improves federated learning on heterogeneous data,” 2023. [Online]. Available: https://openreview.net/forum?id=m_thN8e6qrF 5

[35] A. Abay, Y. Zhou, N. Baracaldo, S. Rajamoni, E. Chuba, and H. Ludwig, “Mitigating bias in federated learning,” *arXiv preprint arXiv:2012.02447*, 2020. 5

[36] Y. H. Ezzeldin, S. Yan, C. He, E. Ferrara, and S. Avestimehr, “Fairfed: Enabling group fairness in federated learning,” *arXiv preprint arXiv:2110.00857*, 2021. 5

doi:10.6342/NTU202301252

[37] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6023–6032. 8, 10, 12, 17, 25

[38] E. Harris, A. Marcu, M. Painter, M. Niranjan, A. Prügel-Bennett, and J. Hare, "Fmix: Enhancing mixed sample data augmentation," *arXiv preprint arXiv:2002.12047*, 2020. 8, 10

[39] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "Augmix: A simple data processing method to improve robustness and uncertainty," *arXiv preprint arXiv:1912.02781*, 2019. 8, 10

[40] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with mixstyle," *arXiv preprint arXiv:2104.02008*, 2021. 8, 10, 17, 25

[41] V. Olsson, W. Tranheden, J. Pinto, and L. Svensson, "Classmix: Segmentation-based data augmentation for semi-supervised learning," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1369–1378. 8, 10

[42] A. Dabouei, S. Soleymani, F. Taherkhani, and N. M. Nasrabadi, "Supermix: Supervising the mixing data augmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 794–13 803. 8, 10

[43] M. Hong, J. Choi, and G. Kim, "Stylemix: Separating content and style for enhanced data augmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 862–14 870. 8, 10

[44] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019. 15

doi:10.6342/NTU202301252

[45] D. Hendrycks and T. G. Dietterich, "Benchmarking neural network robustness to common corruptions and surface variations," *arXiv preprint arXiv:1807.01697*, 2018. 15

[46] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. 15

[47] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017. 15

[48] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. [Online]. Available: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf 15

[49] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Toronto, Ontario, Tech. Rep. 0, 2009. 15

[50] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241. 16

[51] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017. 17, 25

[52] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008. 19

[53] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based local-

doi:10.6342/NTU202301252

ization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626. 21

doi:10.6342/NTU202301252