LCGC: Learning from Consistency Gradient Conflicting for Class-Imbalanced Semi-Supervised Debiasing

Weiwei Xing[1], Yue Cheng [1], Hongzhu Yi [1], Xiaohui Gao[2], Xiang Wei [1] *, Xiaoyu Guo [1], Yuming Zhang [3], Xinyu Pang [4]

[1]School of Software Engineering, Beijing Jiaotong University
[2]School of Automation, Northwestern Polytechnical University
[3]School of Computing, Newcastle University
[4]School of Science, Chongqing University of Posts and Telecommunications
{wwxing, yuecheng, hongzhuyi, xiangwei, guoxiaoyu}@bjtu.edu.cn, gaitxh@foxmail.com
y.zhang361@newcastle.ac.uk, candypxy@163.com

Abstract

Classifiers often learn to be ***biased*** corresponding to the class-imbalanced dataset, especially under the semi-supervised learning (SSL) set. While previous work tries to appropriately re-balance the classifiers by subtracting a class-irrelevant image's logit, but lacks a firm theoretical basis. We theoretically analyze why exploiting a baseline image can refine pseudo-labels and prove that the black image is the best choice. We also indicated that as the training process deepens, the pseudo-labels before and after refinement become closer. Based on this observation, we propose a debiasing scheme dubbed **LCGC**, which Learning from Consistency Gradient Conflicting, by encouraging biased class predictions during training. We intentionally update the pseudo-labels whose gradient conflicts with the debiased logits, representing the optimization direction offered by the over-imbalanced classifier predictions. Then, we debiased the predictions by subtracting the baseline image logits during testing. Extensive experiments demonstrate that **LCGC** can significantly improve the prediction accuracy of existing CISSL models on public benchmarks.

Introduction

The predictions of an ideal unbiased classifier are attributed to class-relevant features of the input images. In real cases, however, such an ideal classifier can hardly be obtained due to the dataset selection biases and imbalances, the complexity of the real world, or other peculiarities (Bengio et al. 2020). That's why classifiers trained on a class-imbalanced set may fail to make trustworthy predictions, which is even more pronounced in semi-supervised learning (SSL) (Chen et al. 2023).

Essentially, when there is training with imbalanced data, the accuracy of the model drops mainly caused by the spurious correlation between the irrelevant features and category labels, i.e. the classifier output logits are biased attributes towards the majority class of class-dependent features. Taking the task of recognizing animal categories as an example, as shown in Figure 1, the sensitive map obtained after vanilla FixMatch gives certain attention to class-irrelevant background features, while the debias model directs its attention more towards class-relevant features. This results in the latter having less background noise.

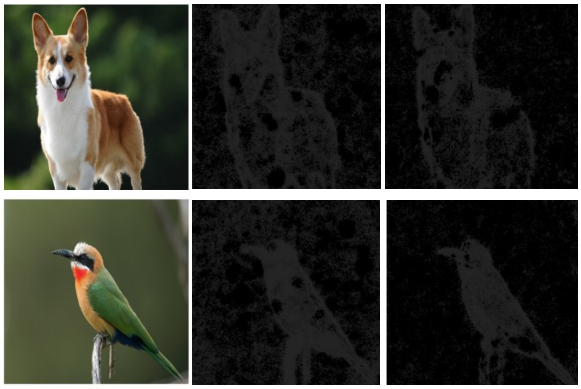


Figure 1: Visualization of sensitive maps produced by the FixMatch model and CDMAD at the image. Left-to-right: original input image, visualization of sensitive maps produced by the FixMatch, CDMAD. The lightness of the sensitive map indicates how much attention the models pay to a particular area of the input image. The sensitive maps obtained by CDMAD have less noise.

To address this issue, a few algorithms (Oh, Kim, and Kweon 2022; Wang et al. 2022; Schmutz, HUMBERT, and Mattei 2023; Lee and Kim 2024) have been proposed for classifier debiasing under class-imbalanced SSL (CISSL) settings. The basic idea of classifier debiasing is to regularize the response bias generated by the biased classification head to achieve the balance effect (Wei and Gan 2023; Wang et al. 2023a; Zhu et al. 2024). Such a training strategy alleviates the bias caused by the imbalanced dataset to a certain extent. However, they need to design complex network variants, such as adding classification heads (Lee, Shin, and Kim 2021), designing causal balancers (Wang et al. 2022; Li et al. 2024; Tao et al. 2024), etc. An immediate idea is whether there is a way to measure the bias of the classifier in a minimally expensive way and to perform post-hoc logit adjustment on the response bias generated by the biased classifi-

*Corresponding author.

cation head. Class-Distribution-Mismatch-Aware Debiasing (CDMAD) does this job well with a solid color image (Lee and Kim 2024). However, what we care about *is there any theoretical evidence to support that a solid color image can debias the classifier logits during training?*

In this paper, our theoretical analysis shows that the CDMAD enhances the balance of the base SSL model implicitly utilizing the integrated gradient flow. It is then natural to ask: *are there any further debiasing methods that can improve the performance of the classifier?* If so, we still expect the new proxy method to be amenable to providing insights for classifier debias and a wider spectrum of architectures. To this end, we propose a novel debiasing model **LCGC**, by **L**earning from **C**onsistency **G**radient **C**onflicting, to overcome the improper training bias from class-imbalanced datasets. In particular, we first update the pseudo-labels whose gradient conflicts with the baseline image refined logits, which is represented as the optimization direction offered by the over-imbalanced classifier predictions. Here, we employ Kullback-Leibler (KL) divergence to measure the consistency between the refinement and original pseudo-labels logits. Encouraging the conflict between them, we train an ***over-biased*** classifier neural network. Then, we expected to debias the predictions by subtraction the baseline image logits during testing.

Our work makes three major contributions. 1) We theoretically analyze that a baseline image enhances the base SSL model by implicitly utilizing the integrated gradient flow. 2) We propose a simple debiased scheme by learning consistency gradient conflicting. 3) We validated the effectiveness of LCGC on four benchmark datasets in both scenarios where the class distributions of the labeled and unlabeled sets either match or mismatch.

Related Work

Classifier Debias

For the classifier debiasing problem, the mainstream methods involve adjustments through re-weighting (Zhang et al. 2021) and re-sampling (Lee, Shin, and Kim 2021), as well as the lately prevalent logits adjustment (Wang et al. 2023b) technique. However, this approach might lead to difficulties and instability in optimization (Cao et al. 2019). To tackle this issue, LfF (Nam et al. 2020) training a pair of neural networks simultaneously that can effectively mitigate the bias in the network. DebiasPL (Wang et al. 2022) designs an adaptive debiasing module from a causal perspective. UniSSL (Huang et al. 2021) proposes class-sharing data detection and feature adaptation methods. DST (Chen et al. 2022) adopts a self-training debiased framework. CDMAD (Lee and Kim 2024) uses solid color images to correct classifier bias during training and inference. Based on this finding, we design the LCGC to train consistent gradients conflicting guided bias model from class-imbalanced data utilizing FixMatch (Sohn et al. 2020) and ReMixMatch (Berthelot et al. 2020).

Class-Imbalanced Semi-Supervised Learning

This part of the research primarily leverages a fusion of CIL and SSL techniques, which allows for the full utilization of unlabeled data during training, while also providing them with balanced pseudo-labels. By assuming that the class distribution of the unlabeled set is known and the same as the labeled dataset. CReST (Wei et al. 2021) uses unlabeled samples predicted as the minority classes more frequently than those predicted as the majority classes for interactive self-training. CoSSL (Cai, Wang, and Hwang 2021) and BaCon (Feng et al. 2024) use an auxiliary classifier and train the classifier to be balanced. SMCLP (Du et al. 2024) utilizes a collaborative manner to exploit the correlations from labels and instances to overcome the imbalanced problem. Conversely, SAW (Lai et al. 2022) mitigates class imbalance using smoothed reweighting based on the number of pseudo-labels belonging to each class without prior knowledge of the class distribution of the unlabeled set.

Problem Setup and Preliminaries

Problem Setup

The problem of class-imbalanced semi-supervised learning aims to train a classifier involving labeled set $\mathcal{X} = \{(x^n, y^n) : n \in (1, \cdots, N)\}$ and unlabeled set $\mathcal{U} = \{(u^m) : m \in (1, \cdots, M)\}$, where $x^n \in \mathbb{R}^d$ and $y^n \in [C] = \{1, \cdots, C\}$ denote the n-th labeled sample and corresponding label, respectively, and $u^m \in \mathbb{R}^d$ denotes the m-th unlabeled sample. Here, the number of labeled and unlabeled samples of class c as N_c and M_c i.e., $\sum_{c=1}^{C} N_c = N$ and $\sum_{c=1}^{C} M_c = M$, where M_c is challenging to know in a realistic scenario. Formally, the ratio of the class imbalance of labeled and unlabeled sets can be represented as $\gamma_l = \frac{N_1}{N_c} \geq 1$ and $\gamma_u = \frac{M_1}{M_c} \geq 1$ under the class-imbalanced training set, where N_1 and M_1 are the number of samples in the largest labeled and unlabeled class. For each iteration of training, we sample minibatches $\mathcal{MX} = \{(x_b^m, y_b^m) : b \in (1, \cdots, B)\} \subset \mathcal{X}$ and $\mathcal{MU} = \{(u_b^m) : b \in (1, \cdots, \mu B)\} \subset \mathcal{U}$ from the training set, where B denotes the minibatch size and μ denotes the relative size of $\mathcal{MU}$ to $\mathcal{MX}$. The goal of a CISSL model is to learn a classifier $f_\theta : \mathbb{R}^d \to \{1, \cdots, C\}$ that effectively classifies samples in a test set $\mathcal{X}^{test} = \{(x_k^{test}, y_k^{test})\}$, where θ denotes parameters of base SSL algorithm. The output logits of f_θ on an input as $g_\theta(\cdot) \in \mathbb{R}^C$, i.e., $f_\theta(\cdot) = \arg\max_c g_\theta(\cdot)_c$, where $(\cdot)_c$ denotes the c-th element.

Backbone SSL Algorithm

In this paper, we consider exploiting FixMatch (Sohn et al. 2020) or ReMixMatch (Berthelot et al. 2020) as the backbone for the base CISSL problem. FixMatch first predicts the class probability of weakly augmented unlabeled data $\alpha(u_b^m)$ as $q_b = \text{Softmax}(g_\theta(\alpha(u_b^m))$ and then generates hard pseudo-label $\tilde{q}_b = \arg\max_c q_{b,c}$. Then the consistency regularization is calculated from $\tilde{q}_b$ and its strongly augmented $\mathcal{A}(u_b^m)$ version. ReMixMatch similarly produces q_b but additionally adopts distribution alignment and sharpening strategies, which also conducts MixUp regularization

and is self-supervised by rotating unlabeled samples (Gidaris, Singh, and Komodakis 2018). The training losses of FixMatch $\mathcal{L}_F$ and ReMixMatch $\mathcal{L}_R$ on $\mathcal{MX}$ and $\mathcal{MU}$ is given by:

$$\mathcal{L}_F(\mathcal{MX}, \mathcal{MU}, \tilde{q}, \tau; \theta) \quad (1)$$

$$\mathcal{L}_R(\mathcal{MX}, \mathcal{MU}, \bar{q}; \theta) \quad (2)$$

where τ denotes a predefined confidence threshold to improve the quality of the pseudo-labels. $\bar{q}$ denotes the sharpened pseudo-label after distribution alignments. Note that, in the absence of specifying FixMatch or ReMixMatch as the backbone for CISSL, we slightly abuse notation by uniformly using $\mathcal{L}$ to denote the training loss.

Class-Distribution-Mismatch-Aware Debiasing

We primarily explore the reason why class-distribution-mismatch-aware debiasing (CDMAD) is effective (Lee and Kim 2024). CDMAD first calculates the logits on a weakly augmented unlabeled sample $g_\theta(\alpha(u_b^m))$, and the logits on a baseline (solid color image) $g_\theta(\mathcal{I})$. Then CDMAD adjusts for the classifier's biased degree by simple subtraction of both the training and testing phase.

$$\textbf{\textit{Trainning phase:}}\ g_\theta^* = g_\theta(\alpha(u_b^m)) - g_\theta(\mathcal{I}) \quad (3)$$

$$\textbf{\textit{Testing phase:}}\ g_\theta^* = g_\theta(x_k^{test}) - g_\theta(\mathcal{I}) \quad (4)$$

where x_k^{logits} is test samples.

Integrated Gradients

Focus on the image classification problem and assume the classifier network $f_\theta : \mathbb{R}^d \to \{1, \cdots, C\}$ is differentiable almost everywhere and perfectly fits the dataset, i.e., there exist interpolating parameters θ^* such that $\forall x_n \in \mathcal{X} :< x_n, \theta^* >= y_n$. Let $x_n \in \mathbb{R}^d$ be the input, and $\mathcal{I} \in \mathbb{R}^d$ be the baseline image, then the integrated gradients are obtained by accumulating the straightline path from the $\mathcal{I}$ to the x_n (Sundararajan, Taly, and Yan 2017).

$$\sum_{i=1}^{d} \text{IntergratedGrads}_i(x_n) = f_\theta(x_n) - f_\theta(\mathcal{I}) \quad (5)$$

Why and How a Baseline Image Can Debias the Classifier Logits

In this section, we characterize the class bias of pseudo-label q_b, which is generated by FixMatch or ReMixMatch from g_θ. Note that the base SSL model parameters are updated according to the gradient flow:

$$\frac{d\theta}{dt} = -\nabla_\theta \mathcal{L}(\mathcal{MX}, \mathcal{MU}, q; \theta) \quad (6)$$

Previous work (Lee and Kim 2024) showed that the classifier's biased degree can be adjusted by a baseline image $\mathcal{I}$. We examine ***why and how a baseline image can debias the classifier logits***? We revisit an empirical phenomenon observed in CDMAD Section 4.4: After exploring a series of measures of baseline image debiasing, they find that the base CISSL framework can also be greatly improved by refinement of biased class predictions only during testing. As

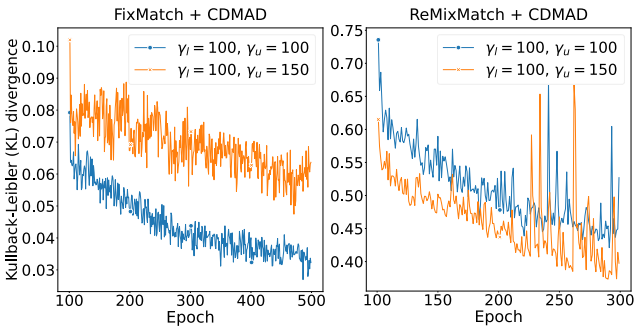


Figure 2: The plot of $\mathcal{L}_{kl}(\theta)$ under CIFAR10-LT dataset and different imbalance ratios γ_l, γ_u. Both columns show that refined by the baseline image, the KL divergence value continues to decrease, indicating that the logit output distribution of the model before and after refinement becomes closer.

severe class distribution mismatch between labeled and unlabeled sets, the resulting classifier's different biased degree toward each class. That is the logit distribution $P(g_\theta(\mathcal{I}))$ of a baseline image $\mathcal{I}$ through CISSL is imbalanced. We formulate the phenomena in the following assumption.

Assumption 1. *The SSL classifiers trained on the class-imbalanced datasets produced biased results:*

$$P(y_n = c|x_n) \propto \exp\left(g_{\theta,c}(x_n)\right)) \quad (7)$$

where $P(y_n = c|x_n) \neq 1/C$ for any $y_n \in [C]$.

For convenience, we define

$$g'_\theta(x_n) = g_\theta(x_n) - g_\theta(\mathcal{I}) \quad (8)$$

where $g_\theta(x_n)$ is the logit of x_n.

We then derive the conditional probability $P_b = P(y|g_\theta(x_n))$ and $P_a = P(y_n|g'_\theta(x_n))$ w.r.t. the outputs of $g_\theta(x_n)$ and $g'_\theta(x_n)$:

Lemma 1. *For a biased classifier trained on the class-imbalanced datasets, the basic CISSL model's logit $g_\theta(x_n)$ and its refinement $g'_\theta(x_n)$ have diverse predictions:*

$$(g_\theta(x_n) \perp g'_\theta(x_n))|y_n \quad (9)$$

We now state the main theorem regarding the class debias of the black baseline image below:

Theorem 1 (Integrated gradient flow for class debiasing)**.** *For a CISSL classifier network f_θ, if a baseline image (the baseline is preferably a black image) is exploited to refine the logit, i.e. $g'_\theta(\alpha(u_b^m)) = g_\theta(\alpha(u_b^m)) - g_\theta(\mathcal{I})$ during the training phase, the gradient flow $-\nabla_\theta \mathcal{L}(\mathcal{MX}, \mathcal{MU}, q; \theta)$ of f_θ contains a linear integrated gradients term $\mathbb{I}(\theta)$:*

$$\nabla_\theta \mathcal{L} = \mathbb{C}(\theta) + \mathbb{I}(\theta)$$
$$\mathbb{I}(\theta) = -\sum_b (\sum_{i=1}^{d} \text{IntergratedGrads}_i(u_b^m)) \nabla g_b \quad (10)$$

where $\mathbb{C}(\theta)$ is the term in the gradient flow other than the $\mathbb{I}(\theta)$ component. To derive the results, we first define some

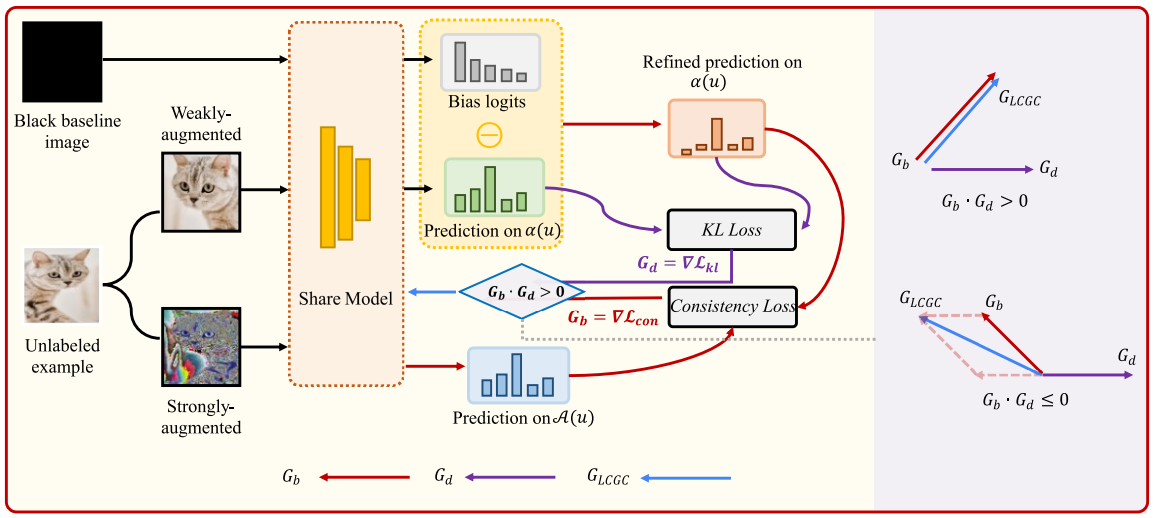


Figure 3: Pseudo-label refinement process using LCGC.

notations. Let $\mathcal{L}(\mathcal{MX}, \mathcal{MU}, q; \theta) = \mathcal{L}_{Con}(\mathcal{MU}, \tilde{q}; \theta) + \mathcal{L}_{Sup}(\mathcal{MX}; \theta)$ denote the loss function of backbone SSL algorithm on a minibatch for labeled set $\mathcal{MX}$ and a minibatch for unlabeled set $\mathcal{MU}$. Therefore, we have

$$\nabla_\theta \mathcal{L} = \nabla_\theta \mathcal{L}_{Con} + \nabla_\theta \mathcal{L}_{Sup} \tag{11}$$

Since the adjust of CDMAD only acts on the $\mathcal{L}_{Con}$ term in $\mathcal{L}$, we only need to study $\nabla_\theta \mathcal{L}_{Con}$. Consequently, we have $\mathcal{L}_{Con} = \frac{1}{\mu B} \sum_{u_b^m \in \mathcal{MU}} \mathbf{H}(P_\theta(y|\mathcal{A}(u_b^m)), \tilde{q}_b)$, where $\mathbf{H}$ is cross-entropy loss. It can be derived

$$\begin{aligned}
\nabla_\theta \mathcal{L}_{Con} &= \sum_b \frac{\partial \mathcal{H}}{\partial q_{\mathcal{A}b}} \frac{\partial q_{\mathcal{A}b}}{\partial \theta} \\
&= \sum_b q_{\mathcal{A}b} \frac{\partial q_{\mathcal{A}b}}{\partial \theta} - \sum_b \tilde{q}_b \frac{\partial q_{\mathcal{A}b}}{\partial \theta} \\
&= \sum_b q_{\mathcal{A}b} \frac{\partial q_{\mathcal{A}b}}{\partial \theta} - \sum_b (q_b - q_{\mathcal{I}}) \frac{\partial q_{\mathcal{A}b}}{\partial \theta} \\
&= \sum_b q_{\mathcal{A}b} \frac{\partial q_{\mathcal{A}b}}{\partial \theta} - \sum_b (g(u_b^m)_b - g(\mathcal{I})_b) \nabla g_b \\
&= -\sum_b (\sum_{i=1}^{d} \text{IntergratedGrads}_i(u_b^m)) \nabla g_b + \sum_b q_{\mathcal{A}b} \frac{\partial q_{\mathcal{A}b}}{\partial \theta}
\end{aligned} \tag{12}$$

Theorem 1 reveals during the training phase, the gradient flow is implicitly guided in the direction of the integrated gradient. With the attribute capability of integral gradient, the class-related features received more attention, and then the bias of the classifier was also reduced. Naturally, the best choice for the baseline image is the same as the baseline for the integrated gradient, which is ***black***.

Furthermore, we found an interesting phenomenon that the distributions of pseudo labels before and after refinement tend to be consistent, as the training process deepens. We employ KL divergence to measure the consistency of this distribution, as shown in Figure 2. This observation leads us to the following corollary:

Corollary 1.1. *Under conditions of Theorem 1, if a black image is exploited for the refinement of pseudo-labels during training and if the gradient flow solution $\theta(\infty)$ satisfies that $f_{\theta(\infty)}(x_b) = y_b$, then gradient flow converges to a minimizer of the Kullback-Leibler (KL) divergence $\mathcal{L}_{kl}(\theta)$ between refinement and original pseudo-labels logits.*

$$\theta(\infty) = \arg\min_\theta \mathcal{L}_{kl}(\theta), \ s.t. \ f_{\theta(\infty)}(x_b) = y_b \tag{13}$$

where

$$\mathcal{L}_{kl} = -\sum_b g_\theta(x_b, \alpha(u_b^m)) \log \frac{g'_\theta(x_b, \alpha(u_b^m))}{g_\theta(x_b, \alpha(u_b^m))} \tag{14}$$

We now summarize the internal mechanism of the classifier debias with a black baseline image under the CISSL framework as follows. **1)** According to Sundararajan, Taly, and Yan, for a pre-trained classifier network f_θ, integrated gradients are better at reflecting class-related features of the input image. Since the CDMAD is implicitly trained in the direction of the linear integrated gradients, more label-irrelevant features can be discarded to achieve better classification results on SSL. **2)** According to Corollary 1.1, a cross-entropy minimization predictor in which the refinement process implicitly regulates the KL divergence between the logits. Therefore, as the KL divergence becomes smaller, the consistency of logits before and after debias increases, which further regularizes the model space and improves the generalization ability of a model.

LCGC: A Consistency Gradient Conflicting Guided Debiasing Method

From the analysis above, we further design a novel debiasing method that can improve the performance of the classifier under CISSL. In particular, our algorithm trains a classifier neural network f_θ as follows: 1) intentionally training the model f_θ to be ***over-biased*** on the unlabeled training samples that exhibit inconsistency before and after logit debiasing and 2) using a black baseline image to refine of biased class predictions during testing. Figure 3 illustrates the pseudolabel refinement process.

Training a Consistency Conflicting Guided Model

Since we hope that the classifier can produce over-biased results, we need to make the distribution of pseudo-labels before and after refinement more different. Motivated by the success of regularization dropout (liang et al. 2021), we compare the debias logits g'_θ with the bias logits g_θ to regularize the gradient direction. Specifically, we obtain the ***biased consistency direction*** G_b by calculating the consistency regularization loss $\mathcal{L}_{Con}$ between the weakly augmented prediction $p(\alpha(u_b^m))$ and the strongly augmented prediction $p(\mathcal{A}(u_b^m))$ as follows:

$$\mathcal{L}_{Con} = -\sum_b p(\mathcal{A}(u_b^m)) \log p(\alpha(u_b^m)) \tag{15}$$

the ***debiased direction*** G_d based on the KL divergence according to Eq. (14).

We denote the gradients of $\mathcal{L}_{Con}$ and $\mathcal{L}_{kl}$ as $G_b = \nabla_u \mathcal{L}_{Con}(u)$ and $G_d = \nabla_u \mathcal{L}_{kl}(u)$, respectively. The relations between G_b and G_d are two-fold. **1)** Their angle is smaller than $90°$, which indicates that the optimization direction of debias logits does not conflict with bias logits. In this case, we safely set the updated gradient direction G_{LCGC} as G_b. **2)** Their angle is larger than $90°$, which indicates that debias logits conflict with bias logits. That is, optimizing the classifier following G_b will lead to an ***over-biased*** classifier. In this case, we project the G_b to the horizontal direction of G_d then add it to G_b to obtain an over-biased direction to optimize the classification model. Although this will slightly increase the KL loss, the post-adjustment logits by the black baseline image will be enhanced. Our LCGC strategy is mathematically formulated as:

$$G_{LCGC} = \begin{cases} G_b & \text{if } G_d \cdot G_b \geq 0 \\ G_b + \lambda \cdot \frac{G_d \cdot G_b}{\|G_d\|^2} G_d & \text{otherwise} \end{cases} \tag{16}$$

Instead of updating the parameter θ using G_b, we optimize the θ using G_{LCGC}, which encourages the gradient direction from biased classifier. We further introduce λ in Eq. (16) to generalize the formulation, which can flexibly control the strength of debias logits guidance in applications. In particular, $\lambda = 1$ denotes projecting G_b to the ***over-biased*** direction, while setting $\lambda = 0$ makes LCGC degenerate to CDMAD, *i.e.*, CDMAD is a special case of our strategy. See more details on the selection of parameter λ in Section 5.3.

Refinement of Biased Logits During Inference

While we train a biased model as described earlier, we must refine biased logit during inference on x_k^{test}, for $k = 1, \cdots, K$. Then, the logits for test samples:

$$g_\theta^*(x_k^{test}) = g'_\theta(x_k^{test}) - g_\theta(\mathcal{I}) \tag{17}$$

where $g_\theta^*(x_k^{test})$ is the finally prediction of x_k^{test}.

Experiments

Experimental Setup

We implemented a series of experiments on CIFAR-10-LT, CIFAR-100-LT (Cui et al. 2019), SVHN-LT (Miyato et al. 2018) and STL-100-LT (Kim et al. 2020a) following the settings of previous studies (Fan et al. 2022; Lee and Kim 2024). To assess classification performance across these datasets, we employed balanced accuracy (bACC) (Huang et al. 2016) and geometric mean (GM) (Kubat, Matwin et al. 1997) as metrics for CIFAR-10-LT, SVHN-LT and STL-10-LT. For CIFAR-100-LT, evaluation was conducted using only the bACC metric. Each experiment was conducted three times on RTX 3090 GPUs to ensure reproducibility, and we report both the average and standard error of the performance measures.

Experimental Results

Table 1 presents a detailed comparison of bACC and GM across various algorithms on CIFAR-10-LT under conditions where γ_u is known and matched to γ_l. It is observed that SSL algorithms like FixMatch and ReMixMatch demonstrate marked improvements in classification performance over the traditional approach. Despite their improved performance, they fall short of the efficiencies achieved by CISSL algorithms, emphasizing the necessity of directly addressing class imbalance issues within SSL frameworks. Notably, CISSL algorithms integrating the LCGC approach consistently outperform other methodologies. This strategy markedly improves performance across varying class imbalance ratios, as evidenced by superior bACC and GM scores.

CIFAR-10-LT ($\gamma = \gamma_l = \gamma_u$, γ_u is assumed to be known)			
Algorithm	$\gamma = 50$	$\gamma = 100$	$\gamma = 150$
Vanilla	65.2±0.1/61.1±0.1	58.8±0.1/58.2±0.1	55.6±0.4/44.0±1.0
Re-sampling	64.3±0.5/60.6±0.7	55.8±0.5/45.1±0.3	52.2±0.1/38.2±1.5
LDAM-DRW	68.9±0.1/67.0±0.1	62.8±0.2/58.9±0.6	57.9±0.2/50.4±0.3
cRT	67.8±0.1/66.3±0.2	63.2±0.5/59.9±0.4	59.3±0.1/54.6±0.7
FixMatch	79.2±0.3/77.8±0.4	71.5±0.7/66.8±1.5	68.4±0.2/59.9±0.4
/+DARP+cRT	85.8±0.4/85.6±0.6	82.4±0.3/81.8±0.2	79.6±0.4/78.9±0.4
/+CReST+LA	85.6±0.4/81.9±0.5	81.2±0.7/74.5±1.0	71.9±2.2/64.4±1.8
/+ABC	85.6±0.3/85.2±0.3	81.1±1.1/80.3±1.3	77.3±1.3/75.6±1.7
/+CoSSL	86.8±0.3/86.6±0.3	83.2±0.5/82.7±0.6	80.3±0.6/79.6±0.6
/+SAW+LA	86.2±0.2/83.9±0.4	80.7±0.2/77.5±0.2	73.7±0.1/71.2±0.2
/+Adsh	83.4±0.1/ -	76.5±0.4/ -	71.5±0.3/ -
/+DebiasPL	- / -	80.6±0.5/ -	- / -
/+UDAL	86.5±0.3/ -	81.4±0.4/ -	77.9±0.3/ -
/+L2AC	- / -	82.1±0.6/81.5±0.6	77.6±0.5/75.8±0.7
/+CDMAD	87.3±0.1/87.0±0.2	83.6±0.5/83.1±0.6	80.8±0.9/79.9±1.1
/+LCGC	**87.3**±0.0/**87.1**±0.1	**84.9**±0.1/**84.6**±0.2	**82.4**±0.0/**81.9**±0.1
ReMixMatch	81.5±0.3/80.2±0.3	73.8±0.4/69.5±0.8	69.9±0.5/62.5±0.4
/+DARP+cRT	87.3±0.6/87.0±0.1	83.5±0.1/83.1±0.1	79.7±0.5/78.9±0.5
/+CReST+LA	84.2±0.1/ -	81.3±0.3/ -	79.2±0.3/ -
/+ABC	87.9±0.5/87.6±0.5	84.5±0.3/84.1±0.4	80.5±1.2/79.5±1.4
/+CoSSL	87.7±0.2/87.6±0.3	84.1±0.6/83.7±0.7	81.3±0.8/80.5±0.8
/+SAW+cRT	87.6±0.2/87.4±0.3	85.4±0.3/83.9±0.2	79.9±0.2/79.9±0.1
/+CDMAD	88.3±0.4/88.1±0.4	85.5±0.5/85.3±0.4	82.5±0.2/82.0±0.3
/+LCGC	**88.7**±0.1/**88.5**±0.1	**85.7**±0.4/**85.4**±0.4	**82.8**±0.3/**82.4**±0.4

Table 1: Comparison of bACC/GM on CIFAR-10-LT under $\gamma = \gamma_l = \gamma_u$ (γ_u is assumed to be known).

Table 2 presents a comparative analysis of bACC and GM metrics on CIFAR-10-LT datasets under conditions where

the class distribution of labeled and unlabeled sets are mismatched ($\gamma_l \neq \gamma_u$). The results highlight the robustness and efficacy of the LCGC approach in addressing class distribution mismatches in semi-supervised learning scenarios. For the CIFAR-10-LT dataset, where γ_l is fixed at 100 and γ_u varies, the performance of FixMatch combined with LCGC is notably superior. We outperform the second-best by an absolute accuracy of 1.7% at an imbalance ratio $\gamma_u = 150$ with FixMatch, showcasing the ability of LCGC to mitigate biases introduced by class distribution mismatches effectively. Note that we employed ReMixMatch* (Kim et al. 2020b) to improve baseline algorithms, significantly improving the classification performance. However, LCGC still achieved better results in real-world scenarios. This makes LCGC more effective and scalable where the class distribution of unlabeled data is unknown and difficult to estimate.

CIFAR-10-LT($\gamma_l = 100$, γ_u is assumed to be unknown)			
Algorithm	$\gamma_u = 1$	$\gamma_u = 50$	$\gamma_u = 150$
FixMatch	68.9±2.0/42.8±8.1	73.9±0.3/70.5±0.5	69.6±0.6/62.6±1.1
/+DARP	85.4±0.6/85.0±0.7	77.3±0.2/75.5±0.2	72.9±0.2/69.5±0.2
/+DAPR+LA	86.6±1.1/86.2±1.2	82.3±0.3/81.5±0.3	78.9±0.2/77.7±0.1
/+DARP+cRT	87.0±0.7/86.8±0.7	82.7±0.2/82.3±0.3	80.7±0.4/80.2±0.6
/+ABC	82.7±0.5/81.9±0.7	82.7±0.6/82.0±0.8	78.4±0.9/77.2±1.1
/+SAW	81.2±0.7/80.2±0.9	79.8±0.3/79.1±0.3	74.5±1.0/72.5±1.4
/+SAW+LA	84.5±0.7/84.1±0.3	82.9±0.4/82.6±0.4	79.1±0.8/78.6±0.9
/+SAW+cRT	84.6±0.2/84.4±0.3	81.6±0.4/81.3±0.3	77.6±0.4/77.1±0.4
/+CDMAD	87.5±0.5/87.1±0.5	85.7±0.4/85.3±0.4	82.3±0.2/81.8±0.3
/+LCGC	**88.2**±0.4/**87.8**±0.4	**85.9**±0.4/**85.4**±0.4	**84.0**±0.2/**83.7**±0.2
ReMixMatch	48.3±0.1/19.5±0.9	75.1±0.4/71.9±0.8	72.5±0.1/68.2±0.3
ReMixMatch*	85.0±1.4/84.3±1.6	77.0±0.1/74.7±0.0	72.8±0.1/68.8±0.2
/+DARP	86.9±0.1/86.4±0.2	77.4±0.2/75.0±0.3	73.2±0.1/69.2±0.3
/+DARP+LA	81.8±0.5/80.9±0.4	83.9±0.4/83.4±0.5	81.1±0.2/80.3±0.3
/+DARP+cRT	88.7±0.3/88.5±0.3	83.5±0.5/83.1±0.5	80.9±0.3/80.3±0.3
/+ABC	76.4±5.3/74.8±6.1	85.2±0.2/84.7±0.3	80.4±0.4/80.0±0.4
/+SAW	87.0±0.8/86.4±0.9	80.6±1.6/79.2±2.2	77.6±0.8/76.0±0.9
/+SAW+LA	74.2±1.5/65.1±2.4	84.8±1.1/82.4±2.3	81.3±2.4/80.9±2.5
/+SAW+cRT	88.8±0.8/88.6±0.8	84.5±0.8/83.6±1.3	82.4±0.1/82.0±0.1
/+CDMAD	89.9±0.5/89.6±0.5	86.9±0.2/86.7±0.2	83.1±0.5/82.7±0.5
/+LCGC	**90.1**±0.5/**89.6**±0.4	**87.0**±0.1/**86.8**±0.1	**83.9**±0.3/**83.6**±0.3

Table 2: Comparison of bACC/GM on CIFAR-10-LT under $\gamma_l \neq \gamma_u$ (γ_u is assumed to be unknown). ReMixMatch* denotes ReMixMatch with the estimated class distribution of the unlabeled set.

CIFAR-10-LT, $\gamma_l = 100$, $\gamma_u = 100$ (reversed)						
Algorithm	ABC	SAW	SAW+LA	SAW+cRT	CDMAD	LCGC
FixMatch+	69.5/66.8	72.3/68.7	74.1/72.0	75.5/73.9	77.1/75.4	**78.6/77.2**
ReMixMatch+	63.6/60.5	79.5/78.5	50.2/14.8	80.8/79.9	81.7/81.0	**83.6/82.4**

Table 3: Comparison of bACC/GM under $\gamma_l = \gamma_u = 100$ (reversed)

To examine the performance of our method in more imbalanced scenarios, experiments were also conducted under conditions where the class distribution of the unlabeled set was imbalanced in the opposite direction to the labeled set. As shown in Table 3, LCGC consistently outperforms other algorithms under these settings.

Table 4 presents a detailed comparison of bACC on CIFAR-100-LT across various algorithms under different class imbalance ratios (γ). The results clearly demonstrate the effectiveness of the proposed LCGC approach in improving classification performance under class-imbalanced settings with a large number of classes. When comparing FixMatch and ReMixMatch combined with various debiasing techniques, it is evident that LCGC consistently achieves the highest bACC scores. In scenarios where γ is set to 20, 50, and 100, FixMatch+LCGC and ReMixMatch+LCGC outperform other methods. Besides the best performance across settings, our method also improves performance for small imbalance ratios as well (1.0% higher than the second-best at imbalance ratio $\gamma = 20$ with FixMatch).

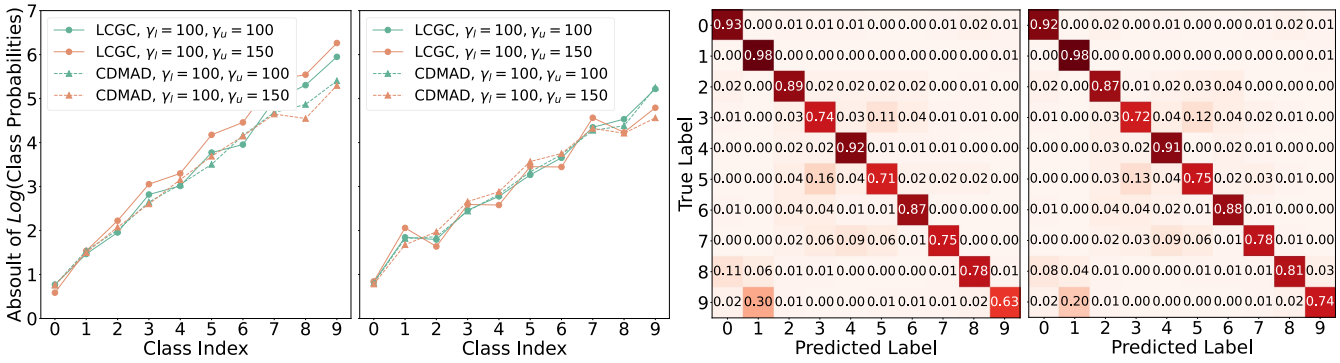
CIFAR-100-LT($\gamma = \gamma_l = \gamma_u$, γ_u is assumed to be known)			
Algorithm	$\gamma = 20$	$\gamma = 50$	$\gamma = 100$
FixMatch	49.6±0.8	42.1±0.3	37.6±0.5
FixMatch+DARP	50.8±0.8	43.1±0.5	38.3±0.5
FixMatch+DARP+cRT	51.4±0.7	44.9±0.5	40.4±0.8
FixMatch+CReST	51.8±0.7	44.9±0.5	40.1±0.7
FixMatch+CReST+LA	52.9±0.1	47.3±0.2	42.7±0.7
FixMatch+ABC	53.3±0.8	46.7±0.3	41.2±0.7
FixMatch+CoSSL	53.9±0.8	47.6±0.6	43.0±0.6
FixMatch+UDAL	-	48.0±0.6	43.7±0.4
FixMatch+CDMAD	54.3±0.4	48.8±0.8	44.1±0.3
FixMatch+LCGC	**55.3**±0.5	**49.3**±0.3	**44.8**±0.5
ReMixMatch	51.6±0.4	44.2±0.6	39.3±0.4
ReMixMatch+DARP	51.9±0.4	44.7±0.7	39.8±0.5
ReMixMatch+DARP+cRT	54.5±0.4	48.5±0.9	43.7±0.8
ReMixMatch+CReST	51.3±0.3	45.5±0.8	41.0±0.8
ReMixMatch+CReST+LA	51.9±0.6	46.6±1.1	41.7±0.7
ReMixMatch+ABC	55.6±0.4	47.9±0.1	42.2±0.1
ReMixMatch+CoSSL	55.8±0.6	48.9±0.6	44.1±0.6
ReMixMatch+CDMAD	57.0±0.3	**51.1**±0.5	44.9±0.4
ReMixMatch+LCGC	**57.3**±0.3	50.7±0.4	**45.9**±0.6

Table 4: Comparison of bACC on CIFAR-100-LT

	STL-10-LT		SVHN-LT
Algorithm	$\gamma_l = 10$	$\gamma_l = 20$	$\gamma_l = 100$
FixMatch	72.9±0.1/69.6±0.0	63.4±0.2/52.6±0.1	88.0±0.3/79.4±0.5
/+DARP	77.8±0.3/76.5±0.4	69.9±1.8/65.4±3.1	88.6±0.2/80.5±0.5
/+DAPR+LA	78.6±0.3/77.4±0.4	71.9±0.5/68.7±0.5	- / -
/+DARP+cRT	79.3±0.2/78.7±0.2	74.1±0.6/73.1±1.2	89.9±0.4/83.5±0.6
/+ABC	79.1±0.5/78.1±0.6	73.8±0.2/72.1±0.2	92.0±0.4/87.9±0.7
/+CDMAD	79.9±0.2/78.9±0.4	75.2±0.4/73.5±0.3	92.4±0.2/92.2±0.2
/+LCGC	**80.1**±0.4/**79.2**±0.3	**76.6**±0.3/**75.2**±0.3	**93.3**±0.2/**93.2**±0.2

Table 5: Comparison of bACC/GM on STL-10-LT and SVHN-LT (γ_u is Unknown)

Table 5 summarizes bACC and GM of the baseline algorithms on STL-10-LT and SVHN-LT. In the STL-10-LT



(a) FixMatch+CDMAD / +LCGC (b) ReMixMatch+CDMAD / +LCGC (c) FixMatch+CDMAD (d) FixMatch+LCGC

Figure 4: (a) and (b) present the class probabilities that take the absolute value of the log predicted on a black image using the proposed algorithm. (c) and (d) present the confusion matrices of the class predictions on test samples.

dataset, where γ_l varies, FixMatch combined with LCGC again demonstrates superior performance. With γ_l set to 10, FixMatch+LCGC achieves better results than other algorithms, and when γ_l is increased to 20, the performance remains consistently higher. In the SVHN-LT dataset, LCGC outperformed the baseline algorithms by a large margin. The effective use of unlabeled samples through gradient-guided pseudo-label refinement may allow the proposed algorithm to be suitable for CISSL on special scenarios datasets.

Small-ImageNet-127 ($\gamma = \gamma_l = \gamma_u$, γ_u is assumed to be known)		
Algorithm	32×32	64×64
FixMatch	29.7	42.3
FixMatch+DARP	30.5	42.5
FixMatch+DARP+cRT	39.7	51.0
FixMatch+CReST	32.5	44.7
FixMatch+CReST+LA	40.9	55.9
FixMatch+ABC	46.9	56.1
FixMatch+CoSSL	43.7	53.8
FixMatch+CDMAD	48.4	59.3
FixMatch+LCGC	49.0	59.8

Table 6: Comparison of bACC/GM on Small-ImageNet-127 (size 32×32 and 64×64, γ_u is assumed to be known)

Table 6 summarizes bACC of the baseline algorithms on Small-ImageNet-127. LCGC achieves the best performance for image sizes 32 and 64, respectively. This result highlights the effectiveness of our method in addressing the challenges of CISSL on large-scale and complex datasets.

Qualitative Analyses

In Figure 4 (a) and (b), we observe the class probability distributions under various configurations for FixMatch and ReMixMatch algorithms enhanced with CDMAD and LCGC debiasing techniques. Notably, in Figure 4 (a), LCGC shows significantly higher probabilities for the head classes and lower probabilities for the tail classes than CDMAD, indicating that it is more sensitive to the bias caused by class imbalance. Therefore, in the testing phase, using the logit of the baseline image to refine the over-biased logits can achieve better results than CDMAD. Figure 4 (c) and

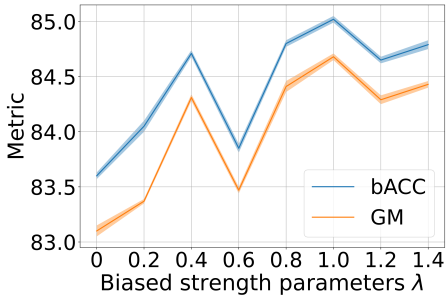


Figure 5: Line chart of validation bACC and GM for the CIFAR-10-LT ($\gamma_l = \gamma_u = 100$) dataset across a range of hyperparameter λ.

(d) display the confusion matrices of the class predictions on the test set of CIFAR-10 using FixMatch+CDMAD and FixMatch+LCGC trained on CIFAR-10LT under $\gamma_l = 100$ and $\gamma_u = 150$. Notably, the LCGC algorithm demonstrates improved classification accuracies across several classes, particularly in handling minority class samples more effectively. We conducted an ablation study to examine the impact of each component of LCGC using CIFAR-10-LT with $\gamma_l = 100$ and $\gamma_u = 150$, as summarized in Table 7. The results indicate that removing the component for refining pseudo-labels results in a substantial performance drop. Similarly, excluding LCGC during the test phase also negatively impacts performance. Moreover, utilizing a confidence threshold ($\tau = 0.95$) and applying distribution alignment techniques yield slightly lower performance compared to the full LCGC implementation.

Sensitivity Analysis

We hold out the CIFAR-10-LT ($\gamma_l = \gamma_u = 100$) as a validation set, and we report the bACC and GM across a range of parameter values in Figure 5. Note that we choose hyperparameters ranging from 0-1.4 with an interval of 0.2. We select $\lambda = 1.0$ as they correspond to the smallest values.

Ablation study ($\gamma_l = 100$, $\gamma_u = 150$)	**bACC/GM**		**bACC/GM**
FixMatch+LCGC	**84.0/83.7**	**ReMixMatch+LCGC**	**83.9/83.6**
Without LCGC for refining pseudo-labels	81.3/80.4	Without LCGC for refining pseudo-labels	83.6/83.0
Without LCGC for test phase	75.9/73.2	Without LCGC for test phase	78.4/76.5
With the use of confidence threshold $\tau = 0.95$	83.0/82.6	With the use of distribution alignment technique	82.1/81.4

Table 7: Ablation study for the proposed algorithm on CIFAR-10-LT under $\gamma_l = 100$ and $\gamma_u = 150$

FixMatch+LCGC	CIFAR-10-LT	
Input	$\gamma_l = \gamma_u = 100$	$\gamma_l = 100$, $\gamma_u = 150$
Red	83.6/83.2	82.1/81.6
Green	83.7/83.4	82.3/81.9
Blue	84.6/84.3	83.0/82.6
Gray	84.5/84.1	83.9/83.4
White	84.9/84.6	84.0/83.6
Black	**84.9/84.6**	**84.0/83.7**

Table 8: Experiments with the replacement of $\mathcal{I}$

Ablation Study

We also conducted experiments to evaluate the effectiveness of replacing the solid color image $\mathcal{I}$ with various color inputs in the LCGC debiasing process using CIFAR-10-LT with $\gamma_l = \gamma_u = 100$ and $\gamma_l = 100$, $\gamma_u = 150$, as summarized in Tab 8. Notably, the use of a black image consistently yielded the highest performance across both $\gamma_l = \gamma_u = 100$ and $\gamma_l = 100$, $\gamma_u = 150$ settings. The experiments with other color inputs, such as red, green, and blue, showed a slight reduction in performance compared to black. This is due to the Batch Normalization (BN) layer in the classifier backbone: solid color images yield uniform or near-uniform outputs after convolution, which the BN layer standardizes to zero mean and unit variance. As a result, solid black and white images perform similarly (though slight numerical differences exist before rounding), and other solid color images exhibit similar effects. Conversely, noise images produce random pixel distributions that vary across regions, leading to inconsistent outputs after BN standardization and thus poorer performance. From the analysis, it is evident that the original solid color image, particularly black, plays a critical role in the debiasing process of LCGC.

Conclusion

We proposed the LCGC algorithm, a novel approach designed to leverage consistency gradient conflicting for improved class representation. We give a theoretical basis for using a baseline image to debias and further integrate gradient guidance information to further improve the debias model. Our comprehensive experiments on benchmark datasets demonstrated that LCGC improves classification accuracy. It is particularly noteworthy that our method is more effective under the class imbalance setting, and the effect becomes better as the imbalance ratio increases.

Acknowledgments

This work was supported by the Beijing Natural Science Foundation under Grant (No.L231005), and by the National Key Research and Development Program of China under Grant (No.2024YFB3312200). Yue Cheng would like to thank Jiuqian Dai for the useful discussion.

References

Bengio, Y.; Deleu, T.; Rahaman, N.; Ke, N. R.; Lachapelle, S.; Bilaniuk, O.; Goyal, A.; and Pal, C. 2020. A Meta-Transfer Objective for Learning to Disentangle Causal Mechanisms. In *International Conference on Learning Representations*.

Berthelot, D.; Carlini, N.; Cubuk, E. D.; Kurakin, A.; Sohn, K.; Zhang, H.; and Raffel, C. 2020. ReMixMatch: Semi-Supervised Learning with Distribution Matching and Augmentation Anchoring. In *International Conference on Learning Representations*.

Cai, J.; Wang, Y.; and Hwang, J.-N. 2021. ACE: Ally Complementary Experts for Solving Long-Tailed Recognition in One-Shot. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 112–121.

Cao, K.; Wei, C.; Gaidon, A.; Arechiga, N.; and Ma, T. 2019. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Chen, B.; Jiang, J.; Wang, X.; Wan, P.; Wang, J.; and Long, M. 2022. Debiased Self-Training for Semi-Supervised Learning. In *Advances in Neural Information Processing Systems*, volume 35, 32424–32437.

Chen, H.; Tao, R.; Fan, Y.; Wang, Y.; Wang, J.; Schiele, B.; Xie, X.; Raj, B.; and Savvides, M. 2023. SoftMatch: Addressing the Quantity-Quality Tradeoff in Semi-supervised Learning. In *The Eleventh International Conference on Learning Representations*.

Cui, Y.; Jia, M.; Lin, T.-Y.; Song, Y.; and Belongie, S. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9268–9277.

Du, G.; Zhang, J.; Zhang, N.; Wu, H.; Wu, P.; and Li, S. 2024. Semi-supervised imbalanced multi-label classification with label propagation. *Pattern Recognition*, 150: 110358.

Fan, Y.; Dai, D.; Kukleva, A.; and Schiele, B. 2022. Cossl: Co-learning of representation and classifier for imbalanced

semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14574–14584.

Feng, Q.; Xie, L.; Fang, S.; and Lin, T. 2024. BaCon: Boosting Imbalanced Semi-supervised Learning via Balanced Feature-Level Contrastive Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(11): 11970–11978.

Gidaris, S.; Singh, P.; and Komodakis, N. 2018. Unsupervised Representation Learning by Predicting Image Rotations. In *International Conference on Learning Representations*.

Huang, C.; Li, Y.; Loy, C. C.; and Tang, X. 2016. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5375–5384.

Huang, Z.; Xue, C.; Han, B.; Yang, J.; and Gong, C. 2021. Universal Semi-Supervised Learning. In *Advances in Neural Information Processing Systems*.

Kim, J.; Hur, Y.; Park, S.; Yang, E.; Hwang, S. J.; and Shin, J. 2020a. Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning. volume 33, 14567–14579.

Kim, J.; Hur, Y.; Park, S.; Yang, E.; Hwang, S. J.; and Shin, J. 2020b. Distribution Aligning Refinery of Pseudo-label for Imbalanced Semi-supervised Learning. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 14567–14579. Curran Associates, Inc.

Kubat, M.; Matwin, S.; et al. 1997. Addressing the curse of imbalanced training sets: one-sided selection. In *Icml*, volume 97, 179. Citeseer.

Lai, Z.; Wang, C.; Gunawan, H.; Cheung, S.-C. S.; and Chuah, C.-N. 2022. Smoothed Adaptive Weighting for Imbalanced Semi-Supervised Learning: Improve Reliability Against Unknown Distribution Data. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, 11828–11843.

Lee, H.; and Kim, H. 2024. CDMAD: Class-Distribution-Mismatch-Aware Debiasing for Class-Imbalanced Semi-Supervised Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 23891–23900.

Lee, H.; Shin, S.; and Kim, H. 2021. ABC: Auxiliary Balanced Classifier for Class-imbalanced Semi-supervised Learning. In *Advances in Neural Information Processing Systems*, volume 34, 7082–7094.

Li, L.; Tao, B.; Han, L.; Zhan, D.-c.; and Ye, H.-j. 2024. Twice Class Bias Correction for Imbalanced Semi-supervised Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(12): 13563–13571.

liang, x.; Wu, L.; Li, J.; Wang, Y.; Meng, Q.; Qin, T.; Chen, W.; Zhang, M.; and Liu, T.-Y. 2021. R-Drop: Regularized Dropout for Neural Networks. In *Advances in Neural Information Processing Systems*, volume 34, 10890–10905.

Miyato, T.; Maeda, S.-i.; Koyama, M.; and Ishii, S. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8): 1979–1993.

Nam, J.; Cha, H.; Ahn, S.; Lee, J.; and Shin, J. 2020. Learning from Failure: De-biasing Classifier from Biased Classifier. In *Advances in Neural Information Processing Systems*, volume 33, 20673–20684.

Oh, Y.; Kim, D.-J.; and Kweon, I. S. 2022. DASO: Distribution-Aware Semantics-Oriented Pseudo-Label for Imbalanced Semi-Supervised Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9786–9796.

Schmutz, H.; HUMBERT, O.; and Mattei, P.-A. 2023. Don't fear the unlabelled: safe semi-supervised learning via debiasing. In *The Eleventh International Conference on Learning Representations*.

Sohn, K.; Berthelot, D.; Carlini, N.; Zhang, Z.; Zhang, H.; Raffel, C. A.; Cubuk, E. D.; Kurakin, A.; and Li, C.-L. 2020. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In *Advances in Neural Information Processing Systems*, 596–608.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic Attribution for Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning*, 3319–3328.

Tao, B.; Li, L.; Li, X.-C.; and Zhan, D.-C. 2024. CLAF: Contrastive Learning with Augmented Features for Imbalanced Semi-Supervised Learning. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7185–7189.

Wang, R.; Jia, X.; Wang, Q.; Wu, Y.; and Meng, D. 2023a. Imbalanced Semi-supervised Learning with Bias Adaptive Classifier. In *The Eleventh International Conference on Learning Representations*.

Wang, X.; Wu, Z.; Lian, L.; and Yu, S. X. 2022. Debiased Learning From Naturally Imbalanced Pseudo-Labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14647–14657.

Wang, Z.; Xu, Q.; Yang, Z.; He, Y.; Cao, X.; and Huang, Q. 2023b. A Unified Generalization Analysis of Re-Weighting and Logit-Adjustment for Imbalanced Learning. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 48417–48430. Curran Associates, Inc.

Wei, C.; Sohn, K.; Mellina, C.; Yuille, A.; and Yang, F. 2021. Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10857–10866.

Wei, T.; and Gan, K. 2023. Towards Realistic Long-Tailed Semi-Supervised Learning: Consistency Is All You Need. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3469–3478.

Zhang, X.; Cui, P.; Xu, R.; Zhou, L.; He, Y.; and Shen, Z. 2021. Deep Stable Learning for Out-of-Distribution Generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5372–5382.

Zhu, G.; Liu, X.; Wu, X.; Tang, S.; Tang, C.; Niu, J.; and Su, H. 2024. Estimating before Debiasing: A Bayesian Approach to Detaching Prior Bias in Federated Semi-Supervised Learning. *arXiv preprint arXiv:2405.19789*.