Appendix for the Paper Entitled "Rebalancing Using Estimated Class Distribution for Imbalanced Semi-Supervised Learning under Class Distribution Mismatch"

Taemin Park*, Hyuck Lee*, and Heeyoung Kim†

Department of Industrial and Systems Engineering, KAIST
Daejeon 34141, Republic of Korea
{ptm1001,dlgur0921,heeyoungkim}@kaist.ac.kr

A Further related works

A.1 Semi-supervised Learning

Semi-supervised learning (SSL) has gained significant attention because of its effectiveness in leveraging both labeled and unlabeled samples. Prominent approaches include entropy minimization and consistency regularization. Entropy minimization [17] aims to prevent decision boundaries from being placed in dense regions by minimizing the entropy of predictions for unlabeled data. This is typically achieved through pseudo-labeling [33]. Consistency regularization [44,49,53] encourages the classifier to make consistent predictions on perturbed data to ensure that decision boundaries lie in low-density regions. Popular SSL algorithms such as MixMatch [4], ReMixMatch [2], and FixMatch [51] combine pseudo-labeling with consistency regularization. Many recent SSL algorithms such as FlexMatch [65], MarginMatch [52], SoftMatch [9], HyperMatch [67], and FreeMatch [59] are built upon the framework of FixMatch [51] owing to its effectiveness and simplicity. Specifically, FlexMatch [65] employs curriculum pseudo-labeling, which considers the learning status and difficulties for each class. MarginMatch [52] masks out low-quality pseudo-labeled samples based on the training dynamics of unlabeled data during training. SoftMatch [9] utilizes a truncated Gaussian function to calculate the weight of each pseudo-label, overcoming the trade-off between the quantity and quality of pseudo-labels. HyperMatch [67] mitigates confirmation bias by introducing a relaxed contrastive loss. FreeMatch [59] utilizes a self-adaptive confidence threshold for each class and adds a self-adaptive fairness loss to FixMatch.

* Equal contribution
† Corresponding author

A.2 Class-imbalanced supervised Learning

Various studies have been proposed for class-imbalanced supervised Learning (CIL). Fundamental methods include re-sampling [8, 14, 41, 56] and re-weighting [12, 37, 40, 47] techniques. Re-sampling rebalances the class distribution of the training set by oversampling the minority classes or undersampling the majority classes. Re-weighting rebalances the gradients by assigning larger weights to the training losses for samples from minority classes. Another approach, LDAM [6], minimizes the generalization error bound by minimizing the label-distribution-aware-margin loss to better generalize minority classes. LA [43] adjusts the algorithm's predictions based on label frequency to minimize the balanced error. Decoupled learning [24] decouples the representation learning from the classifier training to learn both high-quality representations and a balanced classifier. Alshammari *et al.* [1] balance weights through network parameter regularization, and Parisot *et al.* [46] transfer knowledge from common-class features to tail-class features when they are semantically similar. Ma *et al.* [42] introduce the concept of semantic scale to capture broader aspects of imbalance. Additionally, multi-expert learning [5, 35, 57, 63, 66, 68], contrastive learning [11, 23, 39, 54], and knowledge distillation [21, 38] have been proposed for CIL. Even though these CIL techniques effectively mitigate bias in a supervised setting, they encounter challenges when dealing with datasets that contain unlabeled data because CIL techniques rely on sample labels to mitigate bias.

B Training losses of FixMatch and ReMixMatch

B.1 FixMatch

FixMatch [51] is trained by minimizing the classification loss L_{cls} and the consistency regularization loss L_{con}, defined as follows:

$$L_{cls} = \frac{1}{B_l} \sum_{b=1}^{B_l} \mathcal{H}(p_b^m, f(\alpha(x_b^m))), \tag{1}$$

$$L_{con} = \frac{1}{B_u} \sum_{b=1}^{B_u} \mathbb{1}(\max(q_b^m) \geq \epsilon) \mathcal{H}(\widehat{q_b^m}, f(\mathcal{A}(u_b^m))), \tag{2}$$

where $q_b^m = f(\alpha(u_b^m))$ and $\widehat{q_b^m}$ is the one-hot version of $\arg\max(q_b^m)$. The total training loss for FixMatch can be calculated as follows:

$$L_{fix} = L_{cls} + \lambda_u L_{con}, \tag{3}$$

where λ_u is a hyperparameter that represents the weight of the consistency regularization loss.

B.2 ReMixMatch

Let $\mathcal{MB_X}'$ and $\mathcal{MB_U}'$ be the minibatches of labeled data and unlabeled data generated through mixup with $\mathcal{MB_X}$ and $\mathcal{MB_U}$ by blending augmented inputs and labels (or pseudo-labels) of two samples through convex combination. $Mix(x_b^m)$ and $Mix(p_b^m)$ represent the bth input in $\mathcal{MB_X}'$ and its corresponding one-hot label, respectively. Similarly, $Mix(u_b^m)$ and $Mix(\overline{q_b^m})$ represent the bth input in $\mathcal{MB_U}'$ and its corresponding pseudo-label. With the mixed inputs, ReMixMatch is trained by minimizing the mixup regularization loss L_{mix}, calculated as follows:

$$\begin{aligned} L_{mix} = & \frac{1}{|\mathcal{MB_X}'|} \sum_{x_b^m \in \mathcal{MB_X}'} \mathcal{H}(Mix(p_b^m), f(Mix(x_b^m))) \\ & + \frac{\lambda_u}{|\mathcal{MB_U}'|} \sum_{u_b^m \in \mathcal{MB_U}'} \mathcal{H}(Mix(\overline{q_b^m}), f(Mix(u_b^m))), \end{aligned} \tag{4}$$

where λ_u is a hyperparameter that represents the weight of the unlabeled part of the mixup regularization loss. In addition to L_{mix}, ReMixMatch also minimizes the consistency regularization loss L_{con}, calculated as follows:

$$L_{con} = \frac{1}{B_u} \sum_{b=1}^{B_u} \mathcal{H}(\overline{q_b^m}, f(\mathcal{A}(u_b^m))). \tag{5}$$

Finally, ReMixMatch minimizes the rotation loss L_{rot}, which is calculated as follows:

$$L_{rot} = \frac{1}{B_u} \sum_{b=1}^{B_u} \mathcal{H}(r, f_r(rot(\mathcal{A}(u_b^m)))), \tag{6}$$

where $rot(\mathcal{A}(u_b^m))$ denotes $\mathcal{A}(u_b^m)$ rotated r degree, and f_r denotes a classifier for predicting the rotated angle. The total training loss for ReMixMatch can be calculated as follows:

$$L_{remix} = L_{mix} + \hat{\lambda_u} L_{con} + \lambda_r L_{rot}, \tag{7}$$

where $\hat{\lambda_u}$ and λ_r are hyperparameters representing the weight of the consistency regularization loss and rotation loss, respectively.

C Computational complexity of RECD

To verify that estimating the unknown class distribution of unlabeled data and employing AAFM incur negligible computational cost compared to the original ABC, we measured the FLoating point OPerations per Second (FLOPS) for training FixMatch+ABC, FixMatch+RECD without AAFM, FixMatch+RECD, ReMixMatch+ABC, ReMixMatch+RECD without AAFM, and ReMixMatch+RECD. We conducted experiments on CIFAR-10-LT using an RTX 3090, and Tab. 1 summarizes the results. We can observe that estimating the unknown class distribution of unlabeled data and employing AAFM add negligible computational cost.

Table 1: FLOPS for the training of FixMatch+ABC, FixMatch+RECD without AAFM, FixMatch+RECD, ReMixMatch+ABC, ReMixMatch+RECD without AAFM, and ReMixMatch+RECD on CIFAR-10-LT

CIFAR-10-LT	
Algorithm	iteration/sec
FixMatch+ABC	20.01
FixMatch+RECD without AAFM	19.75
FixMatch+RECD	19.47
ReMixMatch+ABC	8.08
ReMixMatch+RECD without AAFM	8.06
ReMixMatch+RECD	8.01

D Pseudo code

Algorithm 1 presents the pseudo-code of the training process for RECD.

E Additional descriptions for the experimental setup

E.1 Detailed descriptions for datasets

We conducted experiments on four benchmark datasets: CIFAR-10-LT, CIFAR-100-LT [13], STL-10-LT [27], and Small-ImageNet-127 [15].

CIFAR-10-LT and CIFAR-100-LT are synthetically generated from CIFAR-10 and CIFAR-100 [29], respectively, by discarding training samples from minority classes. The numbers of labeled and unlabeled samples belonging to the kth class, denoted as N_k and M_k, are calculated as $N_k = N_1 \times {\gamma_l}^{-\frac{k-1}{K-1}}$ and $M_k = M_1 \times {\gamma_u}^{-\frac{k-1}{K-1}}$, where K denotes the total number of classes. For CIFAR-10-LT, we set $N_1 = 1500$ and $M_1 = 3000$. We conducted experiments under the setting where $\gamma = \gamma_l = \gamma_u$, with γ set to 50, 100, and 150. In addition, we conducted experiments under the setting where $\gamma_l \neq \gamma_u$, with γ_l set to 100 and γ_u set to 1, 50, and 150. For CIFAR-100-LT, we set $N_1 = 150$ and $M_1 = 300$. We conducted experiments under the setting where $\gamma = \gamma_l = \gamma_u$, with γ set to 20 and 50.

STL-10-LT is synthetically generated from STL-10 [10] by discarding labeled training samples from minority classes. The numbers of labeled samples belonging to the kth class, denoted as N_k, are calculated as $N_k = N_1 \times {\gamma_l}^{-\frac{k-1}{K-1}}$, where K denotes the total number of classes. STL-10-LT has 100,000 unlabeled samples, and the class distribution of the unlabeled set is unknown. We set $N_1 = 450$ and used all unlabeled samples for training. We conducted experiments on STL-10-LT with γ_l set to 10 and 20.

Algorithm 1 Pseudo code of the training process for RECD

Require: Labeled set $\mathcal{X}$, unlabeled set $\mathcal{U}$

Ensure: Trained feature extractor ξ and trained auxiliary balanced classifier f_{abc}

Initialize $p_\theta(y_u)_k = \frac{N_k}{\sum N_k}, C_{lk} = 0, C_{uk} = 0$ for $k = 1, ..., K$

while Training **do**

  Generate minibatches $\mathcal{MB}_\mathcal{X}$, $\mathcal{MB}_\mathcal{U}$

  Augment the data points in minibatches: $\alpha(x_b^m)$, $\alpha(u_b^m)$, $\mathcal{A}_i(u_b^m)$, $i = 1, 2$

  $q_b^m = f_{abc}(\xi(\alpha(u_b^m)))$ for $b = 1, ..., B_u$

  $\widehat{q_b^m} = \arg\max(q_b^m)$ for $b = 1, ..., B_u$

  $p_\theta(y_u)_{batch} = \frac{1}{B_u} \sum_{b=1}^{B_u} q_b^m$

  $p_\theta(y_u) = \rho \times p_\theta(y_u) + (1 - \rho) \times p_\theta(y_u)_{batch}$

  if $\sum_{b=1}^{B_l} \mathbb{1}(y_b^m = k) > 0$ **then**

    $C_{lk}^{batch} = \frac{1}{\sum_{b=1}^{B_l} \mathbb{1}(y_b^m = k)} \sum_{b=1}^{B_l} f_{abc}(\alpha(x_b^m))_k \times \mathbb{1}(y_b^m = k)$ for $k = 1, ..., K$

    $C_{lk} = \omega \times C_{lk} + (1 - \omega) \times C_{lk}^{batch}$ for $k = 1, ..., K$

  end if

  if $\sum_{b=1}^{B_u} \mathbb{1}(\widehat{q_b^m} = k) > 0$ **then**

    $C_{uk}^{batch} = \frac{1}{\sum_{b=1}^{B_u} \mathbb{1}(\widehat{q_b^m} = k)} \sum_{b=1}^{B_u} f_{abc}(\alpha(u_b^m))_k \times \mathbb{1}(\widehat{q_b^m} = k)$ for $k = 1, ..., K$

    $C_{uk} = \omega \times C_{uk} + (1 - \omega) \times C_{uk}^{batch}$ for $k = 1, ..., K$

  end if

  $\tau_k^l = 1 + \eta \times \frac{N_k}{N_1} \times C_{lk}^2 \times \mathbb{1}(t > T_{warm})$ for $k = 1, ..., K$

  $\tau_k^u = 1 + \eta \times \frac{p_\theta(y_u)_k}{max(p_\theta(y_u))} \times C_{uk}^2 \times \mathbb{1}(t > T_{warm})$ for $k = 1, ..., K$

  Generate 0/1 mask $M(x_b^m) = \mathcal{B}(\frac{N_K}{N_{y_b^m}})$ for $b = 1, ..., B_l$

  Generate 0/1 mask $M(u_b^m) = \mathcal{B}(\frac{min(p_\theta(y_u))}{p_\theta(y_u)_{\widehat{q_b^m}}})$ for $b = 1, ..., B_u$

  L_{cls} : Classification loss for ABC with $(\xi(\alpha(x_b^m)) \times \tau_{y_b^m}^l, y_b^m, M(x_b^m))$

  L_{con} : Regularization loss for ABC with $(\xi(\mathcal{A}_i(y_b^m)) \times \tau_{\widehat{q_b^m}}^u, q_b^m, M(u_b^m))$

  L_{back} : Backbone loss with $(\mathcal{MB}_\mathcal{X}, \tau^l, \mathcal{MB}_\mathcal{U}, \tau^u, q_b^m)$

  $L_{total} = L_{back} + L_{cls} + L_{con}$, $\Delta\theta \propto \nabla_\theta L_{total}$, $\quad \theta \leftarrow \theta + \Delta\theta$

end while

Small-ImageNet-127 is a down-sampled version of ImageNet-127 [20], which is created by consolidating the 1,000 classes from ImageNet [48] into 127 classes based on WordNet hierarchy. Small-ImageNet-127 has 1,281,167 training samples with an imbalance ratio of approximately 286. Following Fan *et al*. [15], we used 10% of the samples in each class as labeled samples, while the remaining were used as unlabeled samples. The samples were down-sampled to sizes of 32×32 and 64×64. We conducted experiments only on Small-ImageNet-127 using FixMatch, similar to [15,60]. The class distribution of the test set of Small-ImageNet-127 is also imbalanced.

E.2 Detailed descriptions for training setups and baseline algorithms

We used a Wide ResNet-28-2 [64] as the Deep CNN for experiments on CIFAR-10-LT, CIFAR-100-LT, and STL-10-LT, while a ResNet-50 [19] was used for experiments on Small-ImageNet-127. For training the algorithm, we used the Adam optimizer [28] with the learning rate set to 1.5×10^{-3} for FixMatch and 2×10^{-3} for ReMixMatch. The network parameters were updated using EMA at each iteration, with a decay factor of 0.999. In the case of RECD, AAFM was applied after the first 20 epochs to ensure stable representation learning at the beginning. Each experiment, excluding those on Small-ImageNet-127, was repeated three times with varying random seeds. We reported the average and standard deviation of the balanced accuracy (bACC) and geometric mean (GM) from these repeated experiments.

For CIFAR-100-LT, we set the weight decay as 0.06 for FixMatch and 0.08 for ReMixMatch to mitigate overfittiing, given the scarcity of samples in each class. For other datasets, we set the weight decay as 0.04 for FixMatch and 0.06 for ReMixMatch when $N + M < 3 \times 10^4$, while we set the weight decay as 0.01 for both FixMatch and ReMixMatch when $N + M \geq 3 \times 10^4$, because a large number of training samples generally lowers the risk of overfitting.

For the experiments using FixMatch, we set the hyperparameter η of AAFM as 1.5 when $N + M < 3 \times 10^4$, while we set η as 0.5 when $N + M \geq 3 \times 10^4$. We trained the algorithm for 500 epochs, with each epoch consisting of 500 iterations. The labeled minibatch size was set to 32, and the unlabeled minibatch size was set to 64. However, for Small-ImageNet-127, we set the labeled minibatch size to 64. We set the confidence threshold ϵ to 0 to utilize the entire unlabeled samples.

For the experiments using ReMixMatch, we set η of AAFM to 1 when $\gamma = 50$ on CIFAR-10-LT, and to 1.5 for other cases when $N + M < 3 \times 10^4$, while we set η to 0.3 when $N + M \geq 3 \times 10^4$. We trained the algorithm for 500 epochs, with both labeled and unlabeled minibatch sizes set to 64. We did not apply the distribution alignment technique of ReMixMatch, as aligning the predictions for unlabeled samples along the class distribution of the labeled set may degrade classification performance when the class distribution of the unlabeled set differs from that of the labeled set. Instead, we incorporated a classification loss for weakly augmented labeled samples into the training loss for ReMixMatch.

We used NVIDIA RTX 2080 Ti, NVIDIA RTX 3090, and NVIDIA TITAN V for the GPU server, and PyTorch versions 1.7.1, 1.8.1, and 1.11.0 for the deep learning library. Classification performance of the proposed algorithm was compared with the following baseline algorithms: **vanilla algorithm** that trains a deep CNN with a cross-entropy loss, **CIL algorithms**, such as Resampling [22], LDAM-DRW [7], and cRT [25], which mitigate class imbalance, **SSL algorithms** such as FixMatch [50] and ReMixMatch [3], which utilize both labeled and unlabeled samples for training, and **CISSL algorithms**, such as DARP [26], CReST [61], ABC [34], CoSSL [16], SAW [31], Adsh [18], DebiasPL [58], UDAL [32], and L2AC [55], which utilize both labeled and unlabeled datasets to mitigate class imbalance. The source codes for implementing the proposed algorithm are provided in the supplementary material.

E.3 Hyperparameters for RECD and AAFM

ρ and ω serve as hyperparameters for EMA in RECD and AAFM, respectively. We set ρ as 0.99 and ω as either 0.9 or 1. For each iteration, ω was set as 0.9 if a minibatch contained samples from class k, and 1 otherwise. In other words, we updated EMA confidence when a minibatch contained corresponding samples for each class. We set ω to be smaller compared to ρ to quickly reflect the algorithm's learning status during the update of EMA. The hyperparameter t represents the power of EMA confidence, and we set it to 2 for all experiments.

F Integrating RECD with DARP

To demonstrate that RECD can be integrated with other CISSL algorithms beyond ABC, we conducted experiments with $\text{RECD}_{\text{DARP}}$, which integrates RECD with DARP [27]. Specifically, DARP refines pseudo-labels by aligning the predictions for unlabeled data along the target distribution, which is the class distribution of the unlabeled set estimated based on the class predictions on the labeled set. $\text{RECD}_{\text{DARP}}$ replaces the target distribution for pseudo-label refinement with $p_\theta(y_u)$, which is the class distribution of the unlabeled set estimated by RECD. In addition, $\text{RECD}_{\text{DARP}}$ applies AAFM to mitigate class imbalance in the feature map. We conducted experiments on CIFAR-10-LT under $\gamma_l = 100$ and γ_u=1. The experimental results are summarized in Tab. 2. $\text{RECD}_{\text{DARP}}$ outperformed DARP, DARP+LA [43], and DARP+cRT [24]. This may be attributed to RECD's superior estimation of the unknown class distribution of unlabeled data compared to DARP, coupled with AAFM's effective mitigation of imbalance in the feature map.

G Comparing RECD with DASO and ACR

Because the classification performances of DASO [45] and ACR [62] were measured under slightly different experimental settings than ours, we did not directly

Table 2: Comparison of bACC/GM on CIFAR-10-LT under $\gamma_l = 100$ and $\gamma_u = 1$

CIFAR-10-LT	
Algorithm	$\gamma_l = 100, \gamma_u = 1$
FixMatch+DARP [27]	85.4/ 85.0
FixMatch+DARP+LA [27,43]	86.6/ 86.2
FixMatch+DARP+cRT [24,27]	87.0/ 86.8
FixMatch+RECD$_{\mathbf{DARP}}$	**88.4/ 88.2**

compare the classification performance of DASO and ACR reported in their respective papers with ours. Instead, we measured their classification performances under identical settings to ours and compared the results. RECD demonstrated outstanding performance compared to DASO and ACR, as presented in Tab. 3, Tab. 4, Tab. 5, and Tab. 6. The experimental results in Tab. 5 reveal that LA degrades the classification performance of DASO when the class distribution of the unlabeled set severely mismatches that of the labeled set. This is presumably because LA only considers the class distribution of labeled data. These results indicate that considering the unknown class distribution of the unlabeled set is crucial to appropriately mitigate class imbalance in CISSL. RECD exhibited notable performance in various settings by effectively estimating the class distribution of the unlabeled set.

Table 3: Comparison of bACC/GM on CIFAR-10-LT under $\gamma = \gamma_l = \gamma_u$

CIFAR-10-LT ($\gamma = \gamma_l = \gamma_u$)			
Algorithm	$\gamma = 50$	$\gamma = 100$	$\gamma = 150$
FixMatch+DASO [45]	81.8/ 81.0	75.7/ 74.0	72.0/ 68.9
FixMatch+DASO+LA [43,45]	84.1/ 83.7	79.4/ 78.8	76.5/ 75.5
FixMatch+RECD	**87.3/ 87.2**	**84.0/ 83.6**	**80.6/ 79.7**
ReMixMatch+DASO [45]	82.5/ 81.9	76.0/ 73.9	70.8/ 66.5
ReMixMatch+DASO+LA [43,45]	85.9/ 85.7	82.8/ 82.4	79.0/ 78.4
ReMixMatch+RECD	**88.1/ 87.9**	**85.4/ 85.2**	**82.5/ 82.1**

Table 4: Comparison of bACC/GM on CIFAR-100-LT under $\gamma = \gamma_l = \gamma_u$

CIFAR-100-LT ($\gamma = \gamma_l = \gamma_u$)		
Algorithm	$r = 20$	$r = 50$
FixMatch+DASO [45]	45.8	39.2
FixMatch+DASO+LA [43,45]	46.2	39.9
FixMatch+RECD	**54.6**	**47.8**
ReMixMatch+DASO [45]	51.5	43.0
ReMixMatch+DASO+LA [43,45]	52.8	45.5
ReMixMatch+RECD	**55.9**	**49.5**

Table 5: Comparison of bACC/GM on CIFAR-10-LT and STL-10-LT under $\gamma_l \neq \gamma_u$.

	CIFAR-10-LT ($\gamma_l = 100$)			STL-10-LT (γ_u = Unknown)	
Algorithm	$\gamma_u = 1$	$\gamma_u = 50$	$\gamma_u = 150$	$\gamma_l = 10$	$\gamma_l = 20$
FixMatch+DASO	86.4/ 86.0	79.1/ 78.2	74.2/ 71.6	68.4/ 65.3	62.1/ 58.9
FixMatch+DASO+LA	86.2/ 85.8	81.7/ 81.2	78.0/ 77.0	68.9/ 66.3	66.0/ 64.6
FixMatch+RECD	**90.2/ 90.0**	**85.6/ 85.3**	**82.3/ 81.8**	**81.4/ 80.6**	**79.0/ 78.1**
ReMixMatch+DASO	89.6/ 89.3	79.6/ 77.8	72.3/ 69.0	75.1/ 73.6	66.8/ 61.8
ReMixMatch+DASO+LA	80.6/ 77.7	84.8/ 84.5	79.7/ 79.2	78.1/ 77.3	75.3/ 74.0
ReMixMatch+RECD	**90.3/ 90.2**	**86.8/ 86.6**	**83.9/ 83.7**	**84.9/ 84.4**	**82.5/ 81.7**

Table 6: Comparison of bACC/GM on CIFAR-10-LT under $\gamma = \gamma_l = \gamma_u = 100$ and $\gamma_l = 100, \gamma_u = 1$

Algorithm/ CIFAR-10-LT	$\gamma_l = \gamma_u = 100$	$\gamma_l = 100, \gamma_u = 1$
FixMatch+ACR [62]	81.8/ 81.4	85.6/ 85.3
FixMatch+RECD	**84.0/ 83.6**	**90.2/ 90.0**

H Fine-grained experimental results of RECD

To demonstrate that RECD effectively improves classification performance on minority classes compared to baseline algorithms, we grouped the first three classes of CIFAR-10-LT as "many", the subsequent four classes as "medium", and the last three classes as "few". We then measured the classification performance for each group. We conducted experiments using FixMatch/ReMixMatch, FixMatch/ReMixMatch+SAW [30], FixMatch/ReMixMatch+SAW+cRT [24], and FixMatch/ReMixMatch+RECD under the setting $\gamma_l = 100$ and $\gamma_u = 1$. From Tab. 7, we can observe that RECD achieves the highest classification performance for the "few" group, implying that RECD effectively mitigates class imbalance.

Table 7: Fine-grained classification performance on CIFAR-10-LT ($\gamma_l = 100$, $\gamma_u = 1$)

CIFAR-10-LT ($\gamma_l = 100, \gamma_u = 1$)				
Algorithm	Overall	Many	Medium	Few
FixMatch [51]	70.2	96.3	77.7	34.0
FixMatch+SAW [30]	81.2	95.6	82.9	64.5
FixMatch+SAW+cRT [24, 30]	84.6	87.8	85.5	80.2
FixMatch+RECD	90.2	93.2	87.5	90.7
ReMixMatch [2]	65.4	96.6	70.8	27.0
ReMixMatch+SAW [30]	87.0	96.8	86.4	78.0
ReMixMatch+SAW+cRT [24, 30]	88.8	94.5	87.8	84.4
ReMixMatch+RECD	90.3	95.0	87.8	89.0

I Results when using FreeMatch as backbone SSL algorithm

To verify that RECD can be effectively combined with recent SSL algorithms, we conducted experiments using FreeMatch [59] as the backbone SSL algorithm. From Tab. 8, we can observe that RECD outperformed the compared algorithms, verifying that RECD can be effectively combined with recent SSL algorithms.

Table 8: Comparison of bACC/GM on CIFAR-10-LT when using FreeMatch as the backbone SSL algorithm

CIFAR-10-LT ($\gamma_l = 100$)		
Algorithm	$\gamma_u = 100$	$\gamma_u = 1$
FreeMatch [59]	74.0/ 71.9	73.2/ 69.3
FreeMatch+SAW+cRT [24,30]	82.8/ 82.3	86.4/ 86.2
FreeMatch+CoSSL [15]	81.7/ 81.3	87.9/ 87.6
FreeMatch+RECD	**83.8/ 83.5**	**90.8/ 90.7**

J Experimental results with fewer labeled samples

We conducted experiments by reducing the ratio of labeled to unlabeled samples, setting $N_1 = 500$ and $M_1 = 4000$ on CIFAR-10-LT, and $N_1 = 150$ on STL-10-LT. As shown in the table below, even with a reduced ratio of labeled to unlabeled samples, RECD outperformed the baseline algorithms in both scenarios where the class distributions of the labeled and unlabeled sets match or mismatch.

Table 9: Comparison of bACC/GM on CIFAR-10-LT with fewer labeled samples

	CIFAR-10-LT ($\gamma_l = 100$)		STL-10-LT (γ_u unknown)	
Algorithm	$\gamma_u = 100$	$\gamma_u = 1$	$\gamma_l = 10$	$\gamma_l = 20$
FixMatch	67.6 / 54.4	62.9 / 44.4	66.0 / 62.5	56.3 / 45.5
/+DARP+cRT	79.4 / 78.9	79.3 / 77.6	72.0 / 70.7	68.9 / 66.3
/+CoSSL	76.3 / 74.9	82.0 / 81.1	73.2 / 71.0	67.0 / 66.7
/+RECD	**80.3/ 79.7**	**87.1/ 86.9**	**75.2/ 74.0**	**72.3/ 70.8**

K Performance comparison between naive feature multiplier and AAFM

We argued that in scenarios where the class distributions of the labeled and unlabeled sets significantly mismatch, the naive feature multiplier fails to adequately mitigate class imbalance in the feature map, whereas AAFM effectively mitigates the class imbalance by considering the unknown class distribution of the unlabeled set. To verify this argument, we compared the classification performance of FixMatch+RECD with the naive feature multiplier and FixMatch+RECD with AAFM on CIFAR-10-LT. In Tab. 10, FixMatch+RECD with AAFM outperformed FixMatch+RECD with the naive feature multiplier in both cases where the class distributions of the labeled and unlabeled sets match and mismatch.

Table 10: Comparison of the naive feature multiplier and AAFM

CIFAR-10-LT ($\gamma_l = 100$)	$\gamma_u = 1$	$\gamma_u = 100$	$\gamma_u = 150$
FixMatch+RECD with Naive feature multiplier [36]	88.9	83.7	80.4
FixMatch+RECD with AAFM	**90.2**	**84.0**	**82.3**

References

1. Alshammari, S., Wang, Y.X., Ramanan, D., Kong, S.: Long-tailed recognition via weight balancing. In: CVPR. pp. 6897–6907 (2022)
2. Berthelot, D., Carlini, N., an Alex Kurakin, E.D.C., Sohn, K., Zhang, H., Raffel, C.: Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In: ICLR (2020)
3. Berthelot, D., Carlini, N., Cubuk, E.D., Kurakin, A., Sohn, K., Zhang, H., Raffel, C.: Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In: International Conference on Learning Representations (2020)
4. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.: Mixmatch: A holistic approach to semi-supervised learning. In: NeurIPS (2019)
5. Cai, J., Wang, Y., Hwang, J.N.: Ace: Ally complementary experts for solving long-tailed recognition in one-shot. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 112–121 (2021)
6. Cao, K., Wei, C., Gaidon, A., Arechiga, N., Ma, T.: Learning imbalanced datasets with label-distribution-aware margin loss. arXiv preprint arXiv:1906.07413v2 (2019)
7. Cao, K., Wei, C., Gaidon, A., Arechiga, N., Ma, T.: Learning imbalanced datasets with label-distribution-aware margin loss. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 32. Curran Associates, Inc. (2019)
8. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research pp. 321–357 (2002)
9. Chen, H., Tao, R., Fan, Y., Wang, Y., Wang, J., Schiele, B., Xie, X., Raj, B., Savvides, M.: Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. In: ICLR (2023)
10. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: In Proceedings of the fourteenth international conference on artificial intelligence and statistics (AISTATS) (2011)
11. Cui, J., Zhong, Z., Liu, S., Yu, B., Jia, J.: Parametric contrastive learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 715–724 (2021)
12. Cui, Y., Jia, M., Lin, T.Y., Song, Y., Belongie, S.: Class-balanced loss based on effective number of samples. In: CVPR. pp. 9268–9277 (2019)
13. Cui, Y., Jia, M., Lin, T.Y., Song, Y., Belongie, S.: Class-balanced loss based on effective number of samples. In: CVPR. pp. 9268–9277 (2019)
14. Estabrooks, A., Jo, T., Japkowicz, N.: A multiple resampling method for learning from imbalanced data sets. Computational Intelligence **20**(1), 18–36 (2004)
15. Fan, Y., Dai, D., Kukleva, A., Schiele, B.: Cossl: Co-learning of representation and classifier for imbalanced semi-supervised learning. In: CVPR. pp. 14574–14584 (2022)
16. Fan, Y., Dai, D., Kukleva, A., Schiele, B.: 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. pp. 14574–14584 (2022)
17. Grandvalet, Y., Bengio, Y.: Semi-supervised learning by entropy minimization. In: NeurIPS (2004)

18. Guo, L.Z., Li, Y.F.: Class-imbalanced semi-supervised learning with adaptive thresholding. In: International Conference on Machine Learning. pp. 8082–8094. PMLR (2022)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
20. Huh, M., Agrawal, P., Efros, A.A.: What makes imagenet good for transfer learning? arXiv preprint arXiv:1608.08614 (2016)
21. Iscen, A., Araujo, A., Gong, B., Schmid, C.: Class-balanced distillation for long-tailed visual recognition. arXiv preprint arXiv:2104.05279 (2021)
22. JAPKOWICZ, N.: The class imbalance problem: Significance and strategies. In: Proc. 2000 International Conference on Artificial Intelligence. vol. 1, pp. 111–117 (2000)
23. Kang, B., Li, Y., Xie, S., Yuan, Z., Feng, J.: Exploring balanced feature spaces for representation learning. In: ICLR (2021)
24. Kang, B., Xie, S., Rohrbach, M., Yan, Z., Gordo, A., Feng, J., Kalantidis, Y.: Decoupling representation and classifier for long-tailed recognition. In: ICLR (2020)
25. Kang, B., Xie, S., Rohrbach, M., Yan, Z., Gordo, A., Feng, J., Kalantidis, Y.: Decoupling representation and classifier for long-tailed recognition. In: International Conference on Learning Representations (2020)
26. Kim, J., Hur, Y., Park, S., Yang, E., Hwang, S.J., Shin, J.: Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 14567–14579. Curran Associates, Inc. (2020)
27. Kim, J., Hur, Y., Park, S., Yang, E., Hwang, S., Shin, J.: Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning. In: NeurIPS (2020)
28. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
29. Krizhevsky, A.: Learning multiple layers of features from tiny images. Technical report, Department of Computer Science, University of Toronto (2009)
30. Lai, Z., Wang, C., Gunawan, H., Cheung, S.C., Chuah, C.N.: Smoothed adaptive weighting for imbalanced semi-supervised learning: Improve reliability against unknown distribution data. In: ICML. vol. 162, pp. 11828–11843 (2022)
31. Lai, Z., Wang, C., Gunawan, H., Cheung, S.C.S., Chuah, C.N.: Smoothed adaptive weighting for imbalanced semi-supervised learning: Improve reliability against unknown distribution data. In: International Conference on Machine Learning. pp. 11828–11843. PMLR (2022)
32. Lazarow, J., Sohn, K., Lee, C.Y., Li, C.L., Zhang, Z., Pfister, T.: Unifying distribution alignment as a loss for imbalanced semi-supervised learning. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 5644–5653 (2023)
33. Lee, D.H.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: In Workshop on challenges in representation learning (ICML) (2013)
34. Lee, H., Shin, S., Kim, H.: Abc: Auxiliary balanced classifier for class-imbalanced semi-supervised learning. Advances in Neural Information Processing Systems **34**, 7082–7094 (2021)
35. Li, B., Han, Z., Li, H., Fu, H., Zhang, C.: Trustworthy long-tailed classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6970–6979 (2022)

36. Li, J., Meng, Z., Shi, D., Song, R., Diao, X., Wang, J., , Xu, H.: Fcc: Feature clusters compression for long-tailed visual recognition. In: CVPR. pp. 24080–24089 (2023)
37. Li, K., Kong, X., Lu, Z., Wenyin, L., Yin, J.: Boosting weighted elm for imbalanced learning. Neurocomputing **128**, 15–21 (2014)
38. Li, T., Wang, L., Wu, G.: Self supervision to distillation for long-tailed visual recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 630–639 (2021)
39. Li, T., Cao, P., Yuan, Y., Fan, L., Yang, Y., Feris, R.S., Indyk, P., Katabi, D.: Targeted supervised contrastive learning for long-tailed recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6918–6928 (2022)
40. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object detection. In: CVPR. pp. 2980–2988 (2017)
41. Liu, X.Y., Wu, J., Zhou, Z.H.: Exploratory undersampling for class-imbalance learning. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **39**(2), 539–550 (2008)
42. Ma, Y., Jiao, L., Liu, F., Li, Y., Yang, S., Liu, X.: Delving into semantic scale imbalance. In: ICLR (2023)
43. Menon, A.K., Jayasumana, S., Rawat, A.S., Jain, H., Veit, A., Kumar, S.: Long-tail learning via logit adjustment. In: ICLR (2021)
44. Miyato, T., ichi Maeda, S., Koyama, M., Ishii, S.: Virtual adversarial training: a regularization method for supervised and semi-supervised learning. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(8), 1979–1993 (2018)
45. Oh, Y., Kim, D.J., Kweon, I.S.: Daso: Distribution-aware semantics-oriented pseudo-label for imbalanced semi-supervised learning. In: CVPR. pp. 9786–9796 (2022)
46. Parisot, S., Esperanca, P.M., McDonagh, S., Madarasz, T.J., Yang, Y., Li, Z.: Long-tail recognition via compositional knowledge transfer. In: CVPR. pp. 6939–6948 (2022)
47. Ren, M., Zeng, W., Yang, B., Urtasun, R.: Learning to reweight examples for robust deep learning. In: ICML. vol. 80, pp. 4334–4343 (2018)
48. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
49. Sajjadi, M., Javanmardi, M., Tasdizen, T.: Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In: NeurIPS (2016)
50. Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C.A., Cubuk, E.D., Kurakin, A., Li, C.L.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. Advances in Neural Information Processing Systems **33** (2020)
51. Sohn, K., Berthelot, D., Li, C.L., Zhang, Z., Carlini, N., Cubuk, E.D., Kurakin, A., Zhang, H., Raffel, C.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In: NeurIPS (2020)
52. Sosea, T., Caragea, C.: Marginmatch: Improving semi-supervised learning with pseudo-margins. In: CVPR. pp. 15773–15782 (2023)
53. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: NeurIPS. pp. 1195–1204 (2017)
54. Wang, P., Han, K., Wei, X.S., Zhang, L., Wang, L.: Contrastive learning based hybrid networks for long-tailed image classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 943–952 (2021)

55. Wang, R., Jia, X., Wang, Q., Wu, Y., Meng, D.: Imbalanced semi-supervised learning with bias adaptive classifier. In: The Eleventh International Conference on Learning Representations (2023)
56. Wang, S., Minku, L.L., Yao, X.: Resampling-based ensemble methods for online class imbalance learning. IEEE Transactions on Knowledge and Data Engineering **27**(5), 1356–1368 (2015)
57. Wang, X., Lian, L., Miao, Z., Liu, Z., Yu., S.X.: Long-tailed recognition by routing diverse distribution-aware experts. In: ICLR (2021)
58. Wang, X., Wu, Z., Lian, L., Yu, S.X.: Debiased learning from naturally imbalanced pseudo-labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14647–14657 (2022)
59. Wang, Y., Chen, H., Heng, Q., Hou, W., Fan, Y., Wu, Z., Wang, J., Savvides, M., Shinozaki, T., Raj, B., Schiele, B., Xie, X.: Freematch: Self-adaptive thresholding for semi-supervised learning. In: ICLR (2023)
60. Wei, C., Sohn, K., Mellina, C., Yuille, A., Yang, F.: Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In: CVPR. pp. 10857–10866 (2021)
61. Wei, C., Sohn, K., Mellina, C., Yuille, A., Yang, F.: Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. arXiv preprint arXiv:2102.09559 (2021)
62. Wei, T., Gan, K.: Towards realistic long-tailed semi-supervised learning: Consistency is all you need. In: CVPR. pp. 3469–3478 (2023)
63. Xiang, L., Ding, G., Han, J.: Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In: European Conference on Computer Vision. pp. 247–263. Springer (2020)
64. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: In Proceedings of the British Machine Vision Conference (BMVC) (2016)
65. Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., Shinozaki, T.: Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In: NeurIPS (2021)
66. Zhang, Y., Hooi, B., Hong, L., Feng, J.: Self-supervised aggregation of diverse experts for test-agnostic long-tailed recognition. Advances in Neural Information Processing Systems **35**, 34077–34090 (2022)
67. Zhou, B., Lu, J., Liu, K., Xu, Y., Cheng, Z., Niu, Y.: Hypermatch: Noise-tolerant semi-supervised learning via relaxed contrastive constraint. In: CVPR. pp. 24017–24026 (2023)
68. Zhou, B., Cui, Q., Wei, X.S., Chen, Z.M.: Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In: CVPR. pp. 9719–9728 (2020)