DeCAB: Debiased Semi-supervised Learning for Imbalanced Open-Set Data

Xiaolin Huang[1], Mengke Li[2], Yang Lu[1(✉)], and Hanzi Wang[1]

[1] Fujian Key Laboratory of Sensing and Computing for Smart City, School of Informatics, Xiamen University, Xiamen, China
luyang@xmu.edu.cn

[2] Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen, China

Abstract. Semi-supervised learning (SSL) has received significant attention due to its ability to use limited labeled data and various unlabeled data to train models with high generalization performance. However, the assumption of a balanced class distribution in traditional SSL approaches limits a wide range of real applications, where the training data exhibits long-tailed distributions. As a consequence, the model is biased towards head classes and disregards tail classes, thereby leading to severe class-aware bias. Additionally, since the unlabeled data may contain out-of-distribution (OOD) samples without manual filtering, the model will be inclined to assign OOD samples to non-tail classes with high confidence, which further overwhelms the tail classes. To alleviate this class-aware bias, we propose an end-to-end semi-supervised method *De*bias *C*lass-*A*ware *B*ias (DeCAB). DeCAB introduces positive-pair scores for contrastive learning instead of positive-negative pairs based on unreliable pseudo-labels, avoiding false negative pairs negatively impacts the feature space. At the same time, DeCAB utilizes class-aware thresholds to select more tail samples and selective sample reweighting for feature learning, preventing OOD samples from being misclassified as head classes and accelerating the convergence speed of the model. Experimental results demonstrate that DeCAB is robust in various semi-supervised benchmarks and achieves state-of-the-art performance. Our code is temporarily available at https://github.com/xlhuang132/decab.

Keywords: Semi-supervised Learning · Imbalanced Learning · Contrastive Learning

This work was supported in part by the National Natural Science Foundation of China under Grants 62002302, 62306181; in part by the FuXiaQuan National Independent Innovation Demonstration Zone Collaborative Innovation Platform under Grant 3502ZCQXT2022008; in part by the China Fundamental Research Funds for the Central Universities under Grants 20720230038.

Q. Liu et al. (Eds.): PRCV 2023, LNCS 14433, pp. 104–119, 2024.
https://doi.org/10.1007/978-981-99-8546-3_9

1 Introduction

Deep supervised learning models have attracted significant interest from both industrial and academia owing to their exceptional performance. However, this outstanding performance is mainly attributed to the abundance of human-annotated data, which can be relatively expensive to obtain [8,9,15]. As a solution, semi-supervised learning (SSL) has emerged as a viable method for leveraging limited labeled data and copious amounts of unlabeled data to achieve models with high generalization performance [1,12,17,19,28].

In the standard SSL paradigm, the assumption is made that the distribution of target data is balanced across all classes, with an equal number of labeled and unlabeled samples for each class. However, it is noteworthy that the class distribution of naturally collected data may potentially exhibit a long-tailed characteristic [24]. In situations where the labeled data is imbalanced, the performance of conventional SSL methods can be adversely affected as the unlabeled data is also likely to be imbalanced. The imbalanced nature of unlabeled data primarily enhances the performance of head classes, while simultaneously exacerbating the performance degradation of the tail classes. This phenomenon is referred to as class-aware bias, which is a significant challenge in semi-supervised learning. As a result, researchers have shown a growing interest in developing imbalance-robust SSL models [6,14,21] to mitigate the effect of this bias.

In addition to the challenge of imbalanced data in SSL, another problem is the existence of out-of-distribution (OOD) samples in the unlabeled data [7]. Without label information and manual filtering, the unlabeled data may have a high probability of containing OOD samples that do not belong to the target distribution. Common SSL methods will treat OOD samples in the same manner as in-distribution (ID) samples, resulting in introducing noisy samples during training. To address this problem, several open-set SSL methods have been proposed [2,7,23]. To mitigate the negative impact of OOD data in SSL, a common strategy is to identify and then filter out or reduce the weight of these samples [7,26]. However, these methods tend to classify unlabeled tail samples as OOD data [20] under long-tailed scenarios, further exacerbating the class-aware bias and diminishing performance. Thus, developing robust SSL methods that can handle both class imbalance and OOD data is crucial for achieving high generalization performance in practical scenarios.

This paper thereby investigates the problem of imbalanced SSL with OOD data, which is a more practical and general scenario in real-world applications. The long-tail distribution commonly leads to the erroneous classification of OOD samples as head samples, consequently intensifying the long-tail problem and perpetuating a detrimental loop. To alleviate this issue, we propose an end-to-end semi-supervised method named *De*bias *C*lass-*A*ware *B*ias (DeCAB). DeCAB leverages contrastive learning to mitigate the class-aware bias by introducing positive-pair scores to replace positive-negative pairs based on unreliable pseudo-labels, thereby avoiding the detrimental impact of false negative pairs on the model. In addition, DeCAB utilizes class-aware thresholds and selective sample reweighting to emphasize feature learning of unconfident unlabeled data.

Our main contributions can be summarized as follows:

- We study a novel problem of SSL on imbalanced open-set data and reveal the fact that the OOD samples exacerbate the long-tail problem in existing SSL methods.
- We propose a simple but effective semi-supervised method DeCAB to tackle more realistic scenarios, by evaluating the feature learning requirements of samples and sample pairs to facilitate feature training.
- DeCAB shows superior performance compared with various state-of-the-art SSL methods on more realistic scenarios, improving the performance of the tail classes while maintaining the performance of the head classes.

2 Related Work

2.1 General SSL Methods

General semi-supervised learning methods can be divided into three categories: pseudo-label-based methods [12,22], consistency regularization-based methods [10,18], and hybrid methods that combine the two [1,16,17]. Specifically, pseudo-label-based semi-supervised methods [12] use the self-training strategy to train the model with labeled data, then pseudo-labels unlabeled data for further training. Consistency regularization-based methods [10,18] encourage the model to output consistent results for the same input data with different forms of augmentation [3,4]. Hybrid methods [1,17] combine pseudo-label-based and consistency regularization-based methods to further improve performance. However, these methods both assume that the target data distribution is balanced and there are no OOD samples in unlabeled data. In more realistic scenarios, data often has two characteristics: 1) ID data follows a natural long-tailed distribution, and 2) unlabeled data may contain OOD data. These characteristics make existing methods vulnerable to distribution shifts and noisy data, thereby reducing performance. Therefore, further research and development of semi-supervised learning are needed to address the challenges in real-world scenarios.

2.2 Imbalanced SSL Methods

Significant progress has been made in developing imbalanced semi-supervised methods to address the first data characteristic. CReST [21] identifies more accurate tail samples and then performs probability sampling biased towards tail classes, leading to a significant improvement in tail class performance. DASO [14] combines similarity-based semantic pseudo-labels with linear classifier pseudo-labels in a self-adaptive manner, and utilizes semantic alignment loss to establish balanced feature representations, reducing biased predictions from the classifier. CoSSL [6] designs a tail class feature enhancement module to alleviate the unbalance problem.

2.3 Open-Set SSL Methods

To address the second data characteristic, open-set SSL has been proposed, which allows the model to identify and reject OOD data, thereby enhancing its robustness and performance on unseen data. OpenMatch [16] introduces the OVA classifier and proposes a soft consistency regularization loss to improve the smoothness of the ova classifier relative to the input transformation. There are currently many SSL methods resorting to contrastive learning to deal with open-set data. For example, CCSSL [23] uses class-aware contrastive learning to enhance the SSL model's ability to handle OOD samples, making the model more robust.

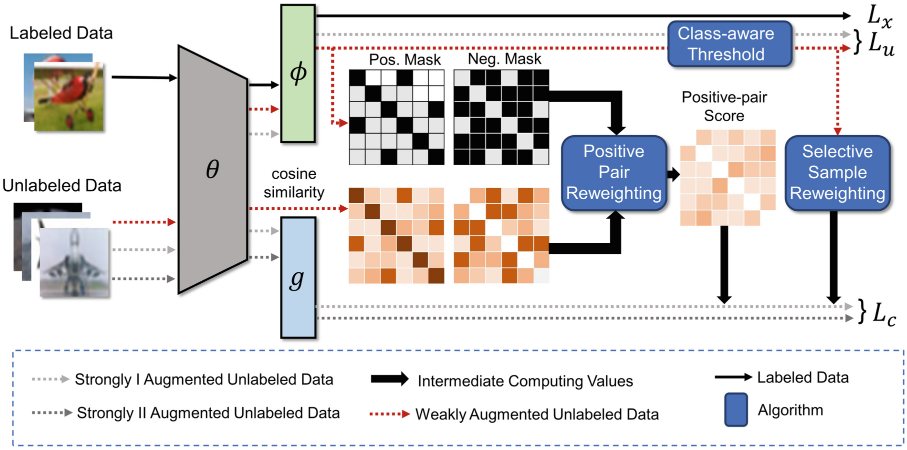

Labeled Data
Unlabeled Data
θ
ϕ
g
cosine similarity
Pos. Mask
Neg. Mask
Positive Pair Reweighting
Positive-pair Score
Class-aware Threshold
Selective Sample Reweighting
L_x
L_u
L_c
Strongly I Augmented Unlabeled Data
Strongly II Augmented Unlabeled Data
Intermediate Computing Values
Weakly Augmented Unlabeled Data
Labeled Data
Algorithm

Fig. 1. Overview of the proposed DeCAB. Unlabeled data is filtered through class-aware thresholds to obtain pseudo-labels, and the selected samples undergo feature learning needs evaluation. Meanwhile, for each sample pair, positive-pair scores are computed to conduct weighted contrastive learning.

3 Proposed Method

To address the issue of class-aware bias, we propose Debias Class-Aware Bias (DeCAB), which is an end-to-end semi-supervised method that incorporates contrastive information. It consists of three core components, that is, class-aware threshold, selective sample reweighting, and positive-pair reweighting, with positive-pair reweighting being the core part. The overall framework of DeCAB is shown in Fig. 1.

3.1 Problem Setting and Notations

In a setting of SSL, we have labeled data $\mathcal{D}_L$ and unlabeled data $\mathcal{D}_U$. Each sample x_i^l in the labeled data $\mathcal{D}_L = \{(x_i^l, y_i^l)\}_{i=1}^N$ is associated with a label $y_i^l \in \{1, ..., C\}$. C is the total number of known classes. $N = \sum_{i=1}^C N_i$ is the total number of labeled data and N_i is the number of samples in class i. We assume the classes are imbalanced and sorted in non-ascending order, i.e., $N_i \geq N_j$ for $i < j$. The imbalance factor of a dataset is measured by the imbalance factor $IF = N_1/N_C$. Unlabeled data $\mathcal{D}_U = \{(x_i^u)\}_{i=1}^M$ contains M samples without annotation. We assume that unlabeled data consists of both ID data $\mathcal{D}_I$ and OOD data $\mathcal{D}_O$, i.e., $\mathcal{D}_U = \mathcal{D}_I \cup \mathcal{D}_O$. The samples in $\mathcal{D}_I$ follow the same class distribution as $\mathcal{D}_L$, and therefore IF is the same for both $\mathcal{D}_L$ and $\mathcal{D}_I$. Samples in $\mathcal{D}_O$ belong to classes other than the known C classes. For unlabeled data, we use one weak augmentation ($a(x^u)$) and two strong augmentations ($A_1(x^u)$, $A_2(x^u)$). We aim to learn a model to effectively learn $\mathcal{D}_L$ and $\mathcal{D}_U$ to generalize well under a class-balanced test criterion.

3.2 Class-Aware Threshold

For unlabeled data, we compute the corresponding consistent loss as FixMatch [17]. The class-aware bias can lead to lower confidence of samples in the tail classes, resulting in a few tail class samples being selected if using a uniform high threshold for filtering. Therefore, we set different thresholds for each class based on sample size to filter more lower-confidence tail samples. The threshold formula is as follows:

$$\tau_c = 0.5 + N_c/N_1 \times (\tau_0 - 0.5), \tag{1}$$

where τ_0 denotes base threshold, and $c \in \{1, 2, 3, \ldots, C\}$. For selected data, we compute the predicted class distribution of the weakly augmented version $a(x_i^u)$ of the unlabeled data point x_i^u: $q_i = f(a(x_i^u))$, and use $\hat{y}_i = \arg\max(q_i)$ as its pseudo-label. Then, we train the model to produce predicted class distribution $\tilde{q} = f(A_1(x_i^u))$ of its strongly augmented [4] version $A_1(x_i^u)$ and constrain it to be consistent with the pseudo-labels $\hat{y}$ by the following loss item:

$$L_u = \frac{1}{\mu B} \sum_{i=1}^{\mu B} \mathbb{1}(\max(q_i) > \tau_{\hat{y}_i}) H(\hat{y}_i, f(A_1(x_i^u))), \tag{2}$$

where $\tau_{\hat{y}_i}$ is the threshold related to class $\hat{y}_i$, μ is a hyper-parameter determining the relative ratio between the mini-batch size of labeled data, and that of unlabeled data, B is the batch size of mini-batch.

By using this mechanism, more tail-class samples are selected explicitly. Moreover, as is shown in Fig. 3, more OOD samples tended to be classified as non-head classes, implicitly mitigating the problem of exacerbated class perception bias due to OOD data. Therefore, the class-aware threshold can alleviate class-aware bias in both explicit and implicit ways.

3.3 Selective Sample Reweighting

For the samples that are filtered by the class-aware threshold, we think that the potential value of each sample in feature learning is not the same. In SSL methods, typically the same weight is assigned for feature learning on each sample. However, in situations with severe class-aware bias, the model may excessively prioritize learning confident head class samples and ignore unconfident tail class samples, resulting in low efficiency in feature learning. To overcome this problem, we utilize a sample reweighting mechanism that emphasizes the feature learning of unconfident samples. Specifically, the weight for feature contrastive learning of sample i is calculated as:

$$s_i = \mathbb{1}(\max(q_i) > \tau_{\hat{y}_i})(1 - \max(q_i)). \tag{3}$$

This mechanism, in combination with the class-aware threshold, can make the model focus more on the feature learning of less confident tail-class samples in the early stage of training, thus accelerating the convergence speed of the model.

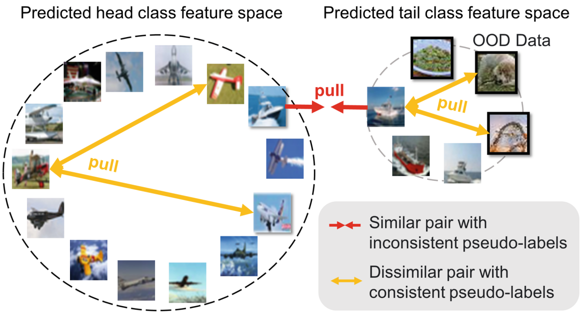

Predicted head class feature space
Predicted tail class feature space
OOD Data
pull
pull
pull
Similar pair with inconsistent pseudo-labels
Dissimilar pair with consistent pseudo-labels

Fig. 2. A schematic shows how positive-pair scores work. The class-aware threshold tends to classify OOD samples as tail classes, and by using positive-pair scores, it pulls tail classes away from similar head classes while correcting misclassified tail samples.

3.4 Positive-Pair Reweighting

Class-aware contrastive learning approaches [13, 27] have demonstrated remarkable performance in supervised learning tasks. However, in the context of imbalanced semi-supervised learning with OOD data, unreliable pseudo-labels can introduce false positive and false negative pairs for class-aware contrastive learning, potentially undermining the effectiveness of semantic feature learning. Therefore, we introduce positive-pair scores to address this issue, and Fig. 2 shows a schematic of how it works. We propose that the feature learning requirements of each sample pair may potentially differ, particularly in the semi-supervised scenario investigated in this paper. For sample pairs with consistent pseudo-labels but dissimilar features and sample pairs with inconsistent pseudo-labels but similar features, we argue they have more learning value, because the former larger feature differences contain more learnable information, and the latter may be potential positive sample pairs. Such a mechanism can avoid the negative impact of false negative pairs on the model.

Specifically, given a sample pair x_i^u and x_j^u, the positive-pair score is calculated as follows:

$$p_{ij} = \begin{cases} 1 - M_{ij}^p \cdot sim(\mathbf{v}_i, \mathbf{v}_j), \text{ if } \hat{y}_i = \hat{y}_j \\ 1 + M_{ij}^n \cdot sim(\mathbf{v}_i, \mathbf{v}_j), \text{ otherwise} \end{cases}, \tag{4}$$

where $\mathbf{v} = \theta(a(x))$ is the high-dimension feature of the weakly augmented version of x extracted by the feature extractor θ, $sim(\cdot, \cdot)$ is cosine similarity computation function, $\hat{y}_i$ and $\hat{y}_j$ denote the pseudo-labels of x_i^u and x_j^u respectively. M_{ij}^p and M_{ij}^n represent the positive-pair mask and negative-pair mask generated through pseudo-labeling (Pos. Mask and Neg. Mask in Fig. 2). If $\hat{y}_i = \hat{y}_j$, M_{ij}^p is set to 1, otherwise, 0, and M_{ij}^n follows the opposite logic.

Thus, for a given sample x_i^u, we calculate the contrastive learning loss of its strongly augmented views by the following formula:

$$L_{c_i} = \frac{1}{\sum_{j \neq i} p_{ij}} \sum_{j=1, j \neq i}^{2\mu B} p_{ij} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_j / T)}{\sum_{k=1, k \neq i}^{2\mu B} \exp(\mathbf{z}_i \cdot \mathbf{z}_k / T)}, \tag{5}$$

where T is the temperature scaling factor, $\mathbf{z}$ is the low-dimension feature projected by g after extracted from θ, and $\mathbf{z} = g(\theta(A_1(x))$ or $\mathbf{z} = g(\theta(A_2(x))$. In a mini-batch, we get the contrastive learning loss as follows:

$$L_c = \frac{1}{2\mu B} \sum_{i=1}^{2\mu B} s_i L_{c_i}, \tag{6}$$

where s_i is the corresponding sample weight obtained from its weak augmentation by Eq. (3). Although this mechanism has no explicit negative pairs, it does not pose a significant risk of feature collapse. This is because it is combined with the class-aware threshold and sample reweighting mechanisms, where the feature learning intensity is reduced for highly confident samples. During the early stage of training, DeCAB prioritizes the learning of features from less confident tail samples while mitigating the impact of false negative pairs by employing positive-pair scores. In the later stages of training, the model assigns increasingly high confidence to samples, leading to a gradual weakening of the corresponding feature learning, reducing the risk of feature collapse.

3.5 Overall Training Objective

For the labeled data, we adopt the cross-entropy loss to utilize supervised information by the following:

$$L_x = \frac{1}{B} \sum_{i=1}^{B} l(y_i, f(x_i)), \tag{7}$$

where $l(\cdot,\cdot)$ is the common cross-entropy loss. The overall training objective function is as follows:

$$L_{total} = L_x + \lambda_u L_u + \lambda_c L_c, \tag{8}$$

where λ_u and λ_c are hyper-parameters, indicating the weight of each loss item.

4 Experimental Results

4.1 Experimental Settings

We compare DeCAB with several related SSL methods including general SSL methods FixMatch [17], MixMatch [1], OpenMatch [16], and imbalanced SSL method CReST [21] and DASO [14].

Imbalanced Dataset with OOD Data. The data used in the experiments consists of three subsets for the purpose of training, validation, and testing. **CIFAR-10/100** [11] are used as ID datasets, which are commonly adopted in the SSL literature [17]. We denote N_1 and M_1 as the number of head class samples in labeled data and unlabeled ID data, $N_i = (IF)^{-\frac{i-1}{C-1}} \cdot N_1$ and the same for M_i, while $N_1 = 1,500$, $M_1 = 3,000$ for CIFAR-10, $N_1 = 150$, $M_1 = 300$ for CIFAR-100. The testing sets of **Tiny ImageNet** (TIN) [5] and **LSUN** [25] are used as the OOD dataset. We mix these two OOD datasets into the unlabeled ID data and train the model on the mixed dataset.

Implementation Details. We employ Wide ResNet-28-2 for CIFAR-10 and Wide ResNet-28-8 for CIFAR-100 as backbone architecture, respectively. The standard training is performed for a total of 250,000 iterations, and validation is conducted every 500 iterations. The labeled data batch size is set to 64 for CIFAR-10, and 16 for CIFAR-100, while the batch size for unlabeled data is twice that, with μ set to 2. τ_0 in Eq. (1) is set to 0.95 for all experiments. The temperature scaling factor T in L_c is set to 0.007. λ_u and λ_c in Eq. (8) are set to 1.0 and 0.2 respectively for all experiments. We utilize the SGD optimizer with a basic learning rate of 0.03, momentum of 0.9, and weight decay of 1e-4. For experimental reproducibility, all experiments fix the random seed to 7.

Evaluation Criteria. In all experiments, the average top-1 accuracy (%) of each class is used for performance evaluation [17]. Additionally, we split the categories of the ID dataset into three groups (Head, Medium, and Tail) according to the class size, with the number of categories per group {3, 3, 4}, and {30, 35, 35} for CIFAR-10 and CIFAR-100, respectively.

4.2 Numerical Comparison

Experiments on the CIFAR-10 Dataset. We conduct a comparative analysis of model performance across different settings of CIFAR-10 with two OOD datasets (TIN and LSUN). As presented in Table 1, DeCAB exhibits remarkable

Table 1. Comparisons of group average accuracy with SSL methods on CIFAR-10 with two different OOD datasets (TIN and LSUN). The best results are shown in bold and the second-best ones are underlined.

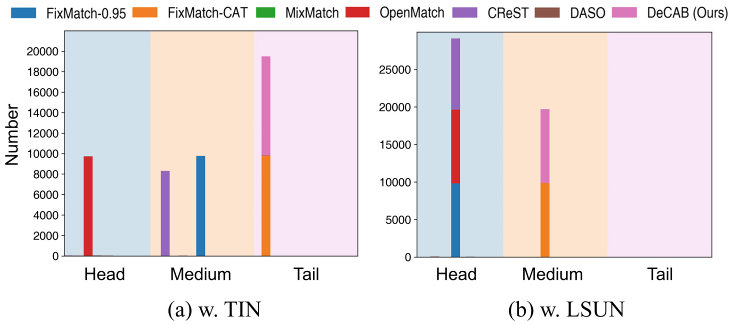
Method	CIFAR-10-LT (IF = 100, TIN)				CIFAR-10-LT (IF = 100, LSUN)			
	Head	Medium	Tail	Avg. Acc	Head	Medium	Tail	Avg. Acc
General SSL Methods								
FixMatch	91.70	**77.10**	51.70	71.32	**94.20**	73.17	52.33	71.14
MixMatch	92.63	66.50	43.72	65.23	91.40	66.13	39.95	63.24
OpenMatch	92.80	70.53	40.42	65.17	91.77	64.37	41.87	63.59
Imbalanced SSL Methods								
CReST	92.13	72.97	51.35	70.07	91.57	70.07	49.98	68.48
DASO	91.67	76.47	57.67	73.51	92.83	72.17	56.10	71.94
DeCAB (Ours)	**94.10**	74.03	**62.42**	**75.41**	88.87	**74.30**	**60.60**	**73.19**

Table 2. Comparisons with SSL methods on CIFAR-100-LT under two different settings with OOD datasets (TIN and LSUN). The best results are shown in bold and the second-best ones are underlined.

Method	CIFAR-100-LT (IF = 100, TIN)				CIFAR-100-LT (IF = 100, LSUN)			
	Head	Medium	Tail	Avg. Acc.	Head	Medium	Tail	Avg. Acc
General SSL Methods								
FixMatch	67.13	36.69	6.37	35.21	66.97	**39.03**	5.89	35.81
MixMatch	61.20	32.14	5.77	31.63	60.20	29.94	6.63	30.86
OpenMatch	65.53	31.03	6.00	32.62	65.77	31.60	5.03	32.55
Imbalanced SSL Methods								
CReST	59.60	30.17	5.97	30.53	61.57	32.40	6.09	31.94
DASO	**71.70**	36.83	7.06	**36.87**	**71.13**	36.94	7.29	36.82
DeCAB (Ours)	66.33	**40.71**	**7.17**	36.66	68.23	38.46	**9.40**	**37.22**

performance and surpasses other methods in overall performance. Specifically, DeCAB outperforms MixMatch and OpenMatch by nearly 10% in overall accuracy when subjected to TIN or LSUN as OOD data. In the case of CIFAR-10-LT (IF=100, TIN), it surpasses FixMatch by almost 5%, CReST by 4%, and DASO by 2% in overall accuracy. In the group average accuracy, DeCAB outperforms MixMatch and OpenMatch by 20%, FixMatch and CReST by 10%, and DASO by approximately 4% in tail classes. From the results presented in the table, we can see that DeCAB achieves a more robust performance than other methods and is more friendly to non-head classes, particularly the tail classes.

Experiments on the CIFAR-100 Dataset. The experimental results on CIFAR-100 are represented on Table 2. It can be seen that DeCAB shows better performance on medium and tail classes compared to other methods, while there is a slight decrease in performance on head classes compared with FixMatch, which we think is acceptable. In the case of CIFAR-100-LT (IF=100, LSUN), DeCAB exhibits superior performance over FixMatch by 3.51%, MixMatch by 2.77%, OpenMatch by 4.37%, CReST by 3.31%, and DASO by 2.11% on the tail class. In the case of CIFAR-100-LT (IF=100, TIN), DeCAB is still more friendly to non-head classes, exhibiting a competitive advantage in terms of overall performance.



(a) w. TIN

(b) w. LSUN

Fig. 3. Comparison of different methods on the CIFAR-100-LT(IF=100, TIN) dataset in terms of the number of OOD misclassification with high confidence (0.95). FixMatch-CAT is a variation of the FixMatch algorithm that utilizes class-aware thresholds.

4.3 Analysis on Impact of OOD Data

In order to understand the role of OOD in the process of training, Fig. 3 depicts a comparison of various methods on the CIFAR-100-LT (IF=100, TIN) dataset in the number of OOD misclassification with high confidence (above 0.95). FixMatch-CAT is a variation of the FixMatch algorithm that utilizes class-aware thresholds. As shown in the figure, OpenMatch, CReST, and FixMatch tend to misclassify OOD samples as non-tail classes, and MixMatch and DASO demonstrate robustness to OOD data and avoid high-confidence misclassification, while FixMatch-CAT and DeCAB prefer to misclassify OOD data as tail classes. It is obvious that the utilization of class-aware thresholds tends to misclassify OOD data into non-head classes with high confidence, which implicitly mitigates the exacerbation of the class-aware bias caused by OOD data. In addition, although MixMatch and DASO did not misclassify OOD samples into ID classes with high confidence, their performance on tail classes is still inferior to that of DeCAB. This observation highlights that the OOD data in DeCAB serves as a beneficial source of information.

Table 3. Results of ablation experiments on CIFAR-10-LT under the setting of $IF = 100$ with Tiny ImageNet as OOD data. CAT, SSR, and PPR denote class-aware threshold, selective sample reweighting, and positive-pair reweighting, respectively. Bold values are the best and underlined values come next.

ID	CAT	SSR	PPR	Head	Medium	Tail	Acc.(%)
0	No contrastive loss			91.70	77.10	51.70	71.32
1	Class-aware contrastive loss			**94.17**	71.43	55.85	72.02
2	✓	–	–	90.77	68.30	**65.45**	<u>73.90</u>
3	–	✓	–	92.37	72.77	53.28	70.85
4	–	–	✓	93.17	73.37	55.53	72.17
5	✓	✓	–	91.60	**74.77**	53.92	71.48
6	✓	–	✓	91.20	72.10	58.45	72.37
7	✓	✓	✓	<u>94.10</u>	<u>74.03</u>	<u>62.42</u>	**75.41**

4.4 Ablation Experiments

We present ablation experiments on three major components of DeCAB, namely class-aware threshold (CAT), selective sample reweighting (SSR), and positive-pair reweighting (PPR). The results are presented in Table 3. **Experiment ID-0** refers to the base model FixMatch and **Experiment ID-1** utilizes class-aware contrastive loss with pseudo-labels, where the model performs best on the head group but poor on the tail group. The results of **Experiment ID-2**, **Experiment ID-3** and **Experiment ID-4** demonstrate that individually employing each module fails to yield significant performance improvements to the model. Comparing the results of **Experiments ID-5**, **ID-6**, and **ID-7**, it is apparent that PPR contributes the most to the model's performance improvement in DeCAB. However, solely relying on PPR may not be sufficient as the model's performance is still suboptimal. The overall results of ablation experiments reveal that the individual components of DeCAB do not independently result in a notable improvement in model performance. However, when integrated into a cohesive framework, the three modules effectively tackle the difficulties posed by imbalanced semi-supervised learning with OOD data.

5 Conclusion

In this paper, we proposed an end-to-end method DeCAB to alleviate serious class-aware bias under an imbalanced open-set scenario. DeCAB introduces positive-pair scores instead of positive-negative pairs in contrastive learning to avoid the detrimental effect of unreliable pseudo-labels. Moreover, by integrating class-aware thresholds, selective sample reweighting, and positive-pair scores, the model can focus on learning features of less confident tail class samples in the early stage and gradually reduce sample feature learning in later stages to avoid feature collapse, thereby improving the performance of tail classes and enhancing generalization performance. Overall, our method provides a simple but effective approach to address the challenge of feature learning in a semi-supervised environment and has the potential to advance state-of-the-art techniques in various machine learning tasks.

Appendix

A Analysis of the Effect of OOD Data to SSL Methods

To reveal the fact that the OOD samples exacerbate the long-tail problem in existing SSL methods, we conduct a quick experiment. In detail, we consider a scenario in which the labeled data is characterized by a long-tailed distribution, with a small number of classes containing a disproportionate number of samples, while the vast majority of classes have a limited number of samples. In addition, the unlabeled data comprises both ID and OOD samples. The unlabeled ID samples follow the same class distribution and a similar long-tailed distribution to the labeled data, while the OOD samples do not belong to any of the ID classes. We conduct a quick experiment to demonstrate that the performance of existing SSL methods deteriorates when confronted with OOD data. To evaluate the model performance under various scenarios, we manipulate the imbalanced factor as well as the inclusion of OOD samples to simulate different settings. Figure 4 compares the confusion matrices of SSL methods on imbalanced training data with and without OOD data. The ID data uses the training set of CIFAR-100 with an imbalance factor of 100, while OOD data uses the testing set of Tiny ImageNet (TIN). It can be seen from the figure that the long-tailed problem leads to performance degradation on the tail class because many tail class samples are misclassified as a head class. The presence of OOD samples exacerbates the long-tailes problem for existing SSL methods.

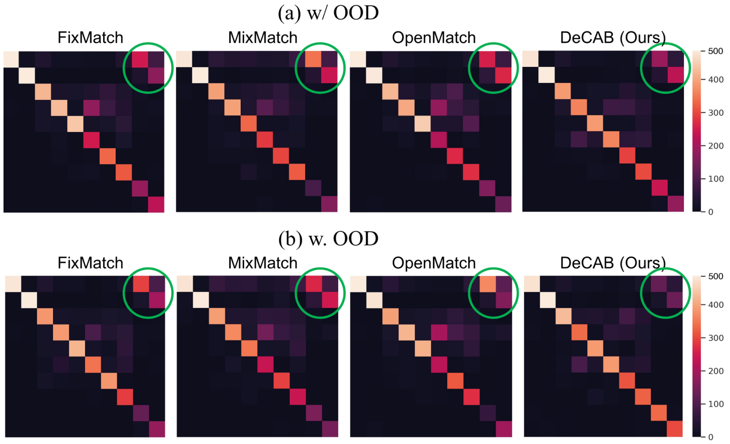


Fig. 4. Confusion matrices of SSL methods on the testing set of CIFAR-10 under two scenarios. The training set of CIFAR-10 is utilized as the labeled and unlabeled ID data with an imbalance factor of 100, and the testing set of Tiny ImageNet (TIN) is used as OOD data. (a) shows the case without OOD data, while (b) shows the case with OOD data.

B Algorithm Flowchart

The algorithm of DeCAB is shown in Algorithm 1.

Algorithm 1. The Proposed DeCAB Algorithm

```
Input: Labeled data $\mathcal{D}_L$, unlabeled data $\mathcal{D}_U$, feature extractor $\theta$, classifier $\phi$, projector $g$, number of epochs $E$, number of iterations per epoch $I$ and learning rate $\eta$.
Output: Feature extractor $\theta$, classifier $\phi$;
Initialize $\theta$, $\phi$ and projector $g$;
for each epoch $t = 1, \ldots, E$ do
    for each iteration $i = 1, 2, \ldots, I$ do
        $S \leftarrow \textsc{sampleReweighting}(X, \theta, \phi)$; // Obtain the selective sample weight by Eq. (3);
        $W \leftarrow \textsc{pairReweighting}(X, \theta)$; // Obtain the posi- tive-pair score of sample pairs by Eq. (4);
        Compute $L_{total}$ by Eq. (8) with $S$ and $W$;
        Update $\theta$, $\phi$, $g$ by $L_{total}$ and $\eta$;
    end for
end for
return $\theta, \phi$.
```

C Visualized Comparison

In order to evaluate the learning of the model on the feature space, we perform a visualized comparative analysis of the test set features extracted from the backbone of the model obtained by each method. Figure 5 shows the t-SNE visualization of feature space about the testing set of CIFAR-10, where the model is trained on CIFAR-10-LT (IF=100, TIN). In the figure, the black circle circles the space of the easily confused head and tail classes. The feature space that is learned by other methods typically shows an aggregation of similar head and tail classes, with a significant proportion of misclassified tail classes located in the middle of the black circles. In contrast, DeCAB exhibits a much clearer separation between head and tail classes in the feature space, resulting in fewer samples being misclassified in the middle. These results illustrate that our method can obtain a better feature space.

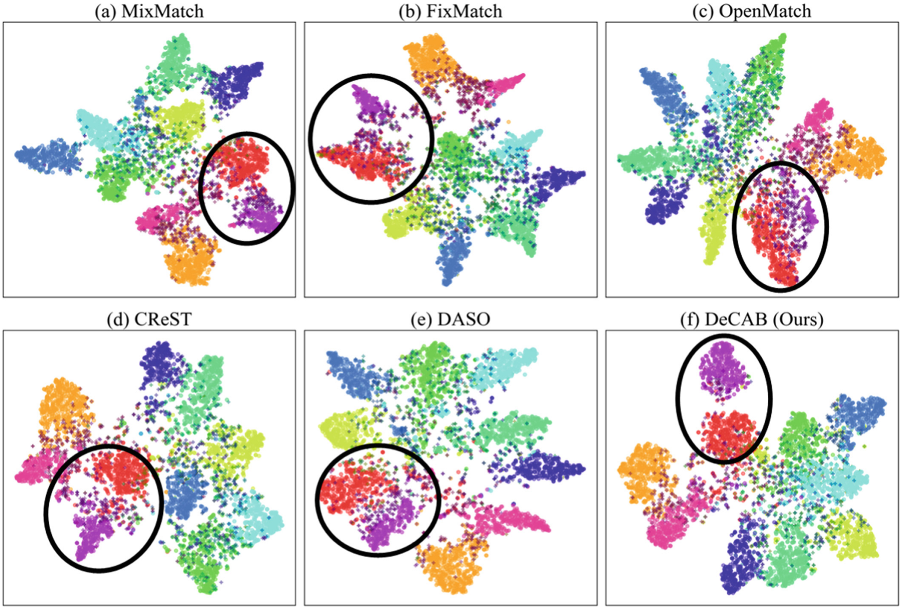


Fig. 5. The t-SNE visualization of feature space of CIFAR-10-LT test set, trained on CIFAR-10-LT with IF=100 and Tiny ImageNet as OOD data. The black circle circles the easily confused head and tail samples.

References

1. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: MixMatch: a holistic approach to semi-supervised learning. In: Advances in Neural Information Processing Systems, pp. 5050–5060 (2019)
2. Chen, Y., Zhu, X., Li, W., Gong, S.: Semi-supervised learning under class distribution mismatch. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 3569–3576 (2020)
3. Cubuk, E.D., Zoph, B., Mane, D., Vasudevan, V., Le, Q.V.: AutoAugment: learning augmentation policies from data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 113–123 (2019)
4. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.: RandAugment: practical automated data augmentation with a reduced search space. In: Advances in Neural Information Processing Systems, pp. 18613–18624 (2020)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009)
6. 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)
7. Guo, L.Z., Zhang, Z.Y., Jiang, Y., Li, Y.F., Zhou, Z.H.: Safe deep semi-supervised learning for unseen-class unlabeled data. In: International Conference on Machine Learning, pp. 3897–3906 (2020)

8. Hestness, J., et al.: Deep learning scaling is predictable, empirically. arXiv preprint arXiv:1712.00409 (2017)
9. Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., Wu, Y.: Exploring the limits of language modeling. arXiv preprint arXiv:1602.02410 (2016)
10. Ke, Z., Wang, D., Yan, Q., Ren, J., Lau, R.W.: Dual student: Breaking the limits of the teacher in semi-supervised learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6728–6736 (2019)
11. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Tech. rep. (2009)
12. Lee, D.H., et al.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on Challenges in Representation Learning, ICML (2013)
13. Li, T., et al.: 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)
14. Oh, Y., Kim, D.J., Kweon, I.S.: 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,786–9796 (2022)
15. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. 5485–5551 (2020)
16. Saito, K., Kim, D., Saenko, K.: OpenMatch: open-set consistency regularization for semi-supervised learning with outliers. arXiv preprint arXiv:2105.14148 (2021)
17. Sohn, K., et al.: FixMatch: simplifying semi-supervised learning with consistency and confidence. In: Advances in Neural Information Processing Systems, pp. 596–608 (2020)
18. Tarvainen, A., Valpola, H.: Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In: Advances in Neural Information Processing Systems, pp. 1195–1204 (2017)
19. Van Engelen, J.E., Hoos, H.H.: A survey on semi-supervised learning. Mach. Learn. 373–440 (2020)
20. Wang, H., et al.: Partial and asymmetric contrastive learning for out-of-distribution detection in long-tailed recognition. In: International Conference on Machine Learning, pp. 23446–23458 (2022)
21. Wei, C., Sohn, K., Mellina, C., Yuille, A., Yang, F.: 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, pp. 10857–10866 (2021)
22. Xu, Y., et al.: Dash: semi-supervised learning with dynamic thresholding. In: International Conference on Machine Learning, pp. 11525–11536 (2021)
23. Yang, F., et al.: Class-aware contrastive semi-supervised learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14421–14430 (2022)
24. Yang, L., Jiang, H., Song, Q., Guo, J.: A survey on long-tailed visual recognition. Int. J. Comput. Vis. 1837–1872 (2022)
25. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: LSUN: construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015)

26. Yu, Q., Ikami, D., Irie, G., Aizawa, K.: Multi-task curriculum framework for open-set semi-supervised learning. In: European Conference on Computer Vision, pp. 438–454 (2020)
27. Zhu, J., Wang, Z., Chen, J., Chen, Y.P.P., Jiang, Y.G.: Balanced contrastive learning for long-tailed visual recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6908–6917 (2022)
28. Zhu, X.J.: Semi-supervised learning literature survey (2005)