ProMix: Combating Label Noise via Maximizing Clean Sample Utility

Ruixuan Xiao[1], **Yiwen Dong**[1], **Haobo Wang**[1*], **Lei Feng**[2],
Runze Wu[3], **Gang Chen**[1] and **Junbo Zhao**[1]
[1]Zhejiang University, Hangzhou, China
[2]Nanyang Technological University, Singapore
[3]NetEase Fuxi AI Lab, Hangzhou, China
{xiaoruixuan, dyw424, wanghaobo, cg, j.zhao}@zju.edu.cn,
lfengqaq@gmail.com, wurunze1@corp.netease.com

IJCAI 2023

Related Work

1. Learning with Noisy Labels:

The vanilla CE is proved to easily overfit corrupted data. Existing methods can be roughly categorized into two types:

- the estimated noise transition matrix to explicitly correct the loss function, but hard in the presence of heavy noise and a large number of classes.
- filter out a clean subset for robust training, e.g. DivideMix(to leverage the unselected examples): existing a quality-quantity trade-off.

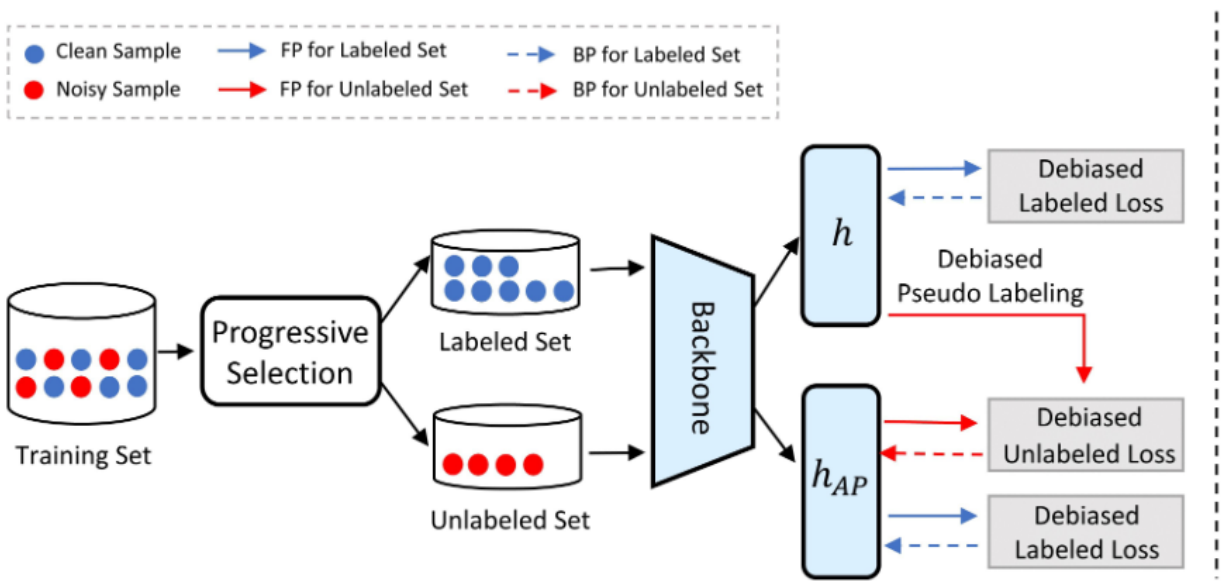
Motivation: our work targets to maximize the utility of clean samples.

2. Debiased Learning:

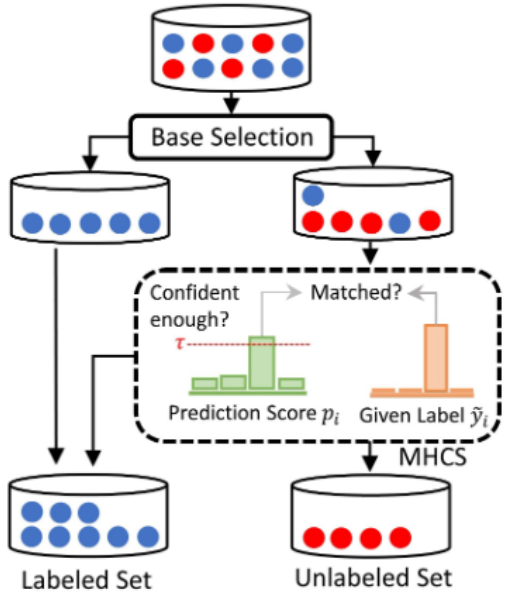
In long-tailed learning, models can be easily biased towards dominant classes. As for SSL, confirmation bias(unreliable pseudo-labels) may hurt generalization performance.

Motivation: we attempt to alleviate the biases in our selection and pseudo-labeling procedures.

Method

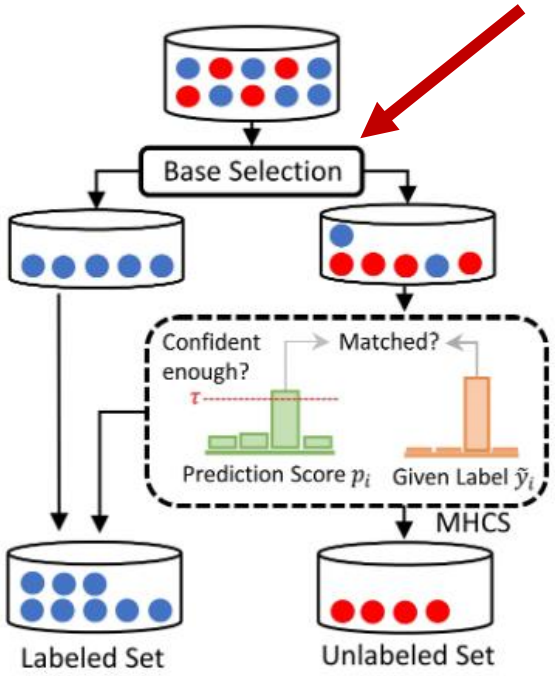


(a) Overall Framework of ProMix.



(b) Progressive Selection.

Method/Class-wise Small-loss Selection (CSS)



(b) Progressive Selection.

Motivation:

- In the training, the easy classes are usually fitted.
- The loss value of example with different observed labels may not be comparable.

Datasets D to C sets according to the noisy labels at each epoch:

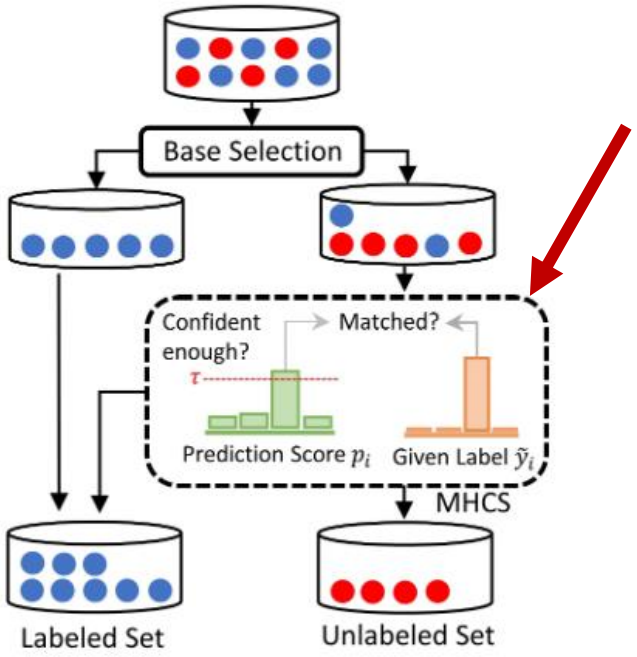
$$\mathcal{S}_j = \{(\boldsymbol{x}_i, \tilde{y}_i) \in \mathcal{D} | \tilde{y}_i = j\}$$

Produce a roughly balanced base set:

$$k = \min(\lceil \frac{n}{C} \times R \rceil, |\mathcal{S}_j|)$$

$$\mathcal{D}_{CSS} = \cup_{j=1}^{C} \mathcal{C}_j$$

Method/Matched High Confidence Selection (MHCS)



(b) Progressive Selection.

Motivation:
Find out the potentially clean samples missed by CSS

Inspiration:
the DNNs can assign an arbitrary label with high confidence and select them as clean, which results in a vicious cycle.

we choose those samples to have high confidence in their given labels, which tend to be originally clean.

$$e_i = \max_j p_i^j \qquad y_i' = \arg\max_j p_i^j$$

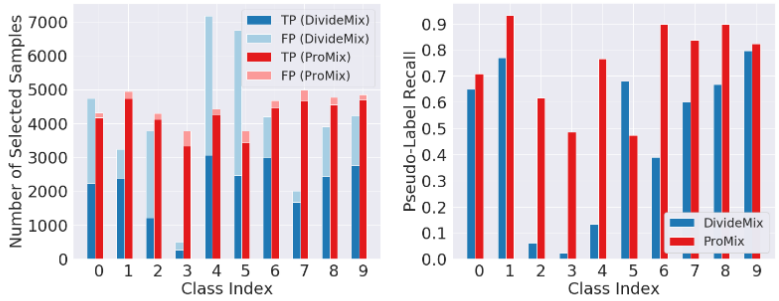
$$(\max_j f_1^j(\boldsymbol{x}) \geq \tau) \wedge (\max_j f_2^j(\boldsymbol{x}) \geq \tau) \wedge (\arg\max_j f_1^j(\boldsymbol{x}) == \arg\max_j f_2^j(\boldsymbol{x})) \quad (8)$$

Label Guessing by Agreement (LGA)

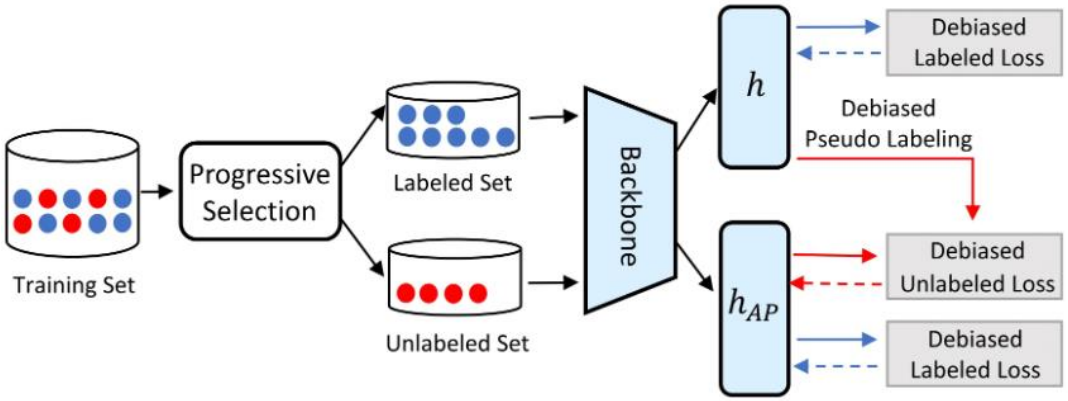
$$\mathcal{D}_{MHCS} = \{(\boldsymbol{x}_i, \tilde{y}_i) \in \mathcal{D} | e_i \geq \tau, y_i' = \tilde{y}_i\} \quad (3)$$

$$\mathcal{D}_l = \mathcal{D}_{CSS} \cup \mathcal{D}_{MHCS}$$

Method/Debiased Semi-Supervised Training



(a) Number of selected samples. (b) Recall of pseudo-labels.



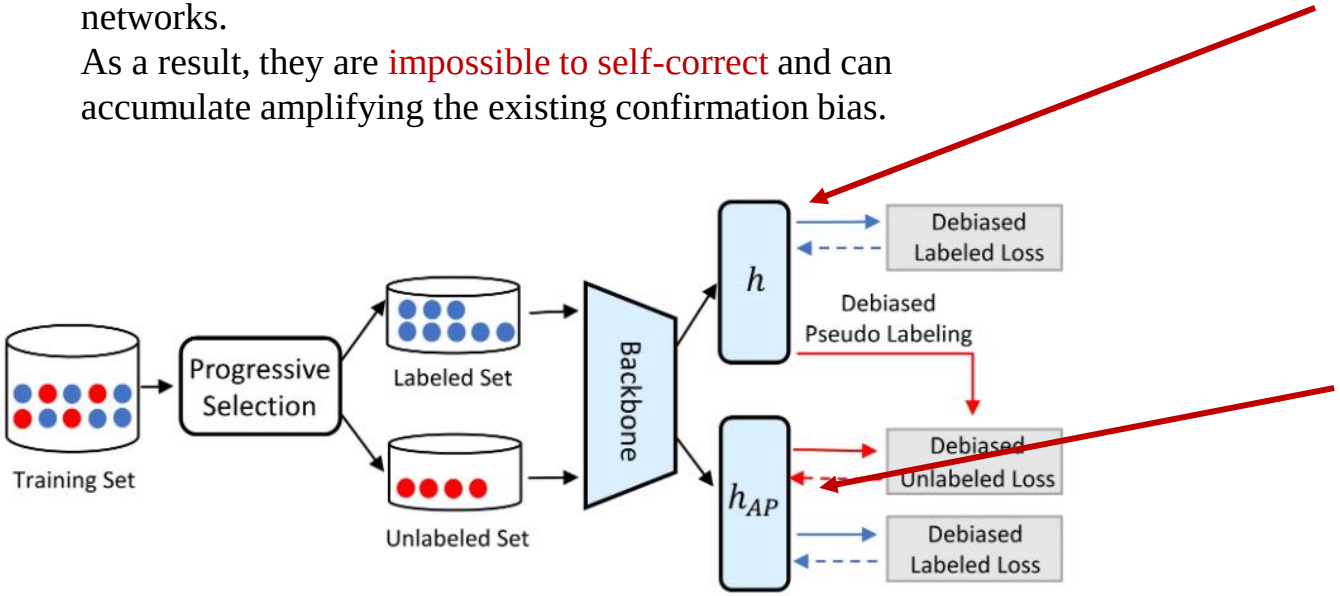
Motivation:

1. to utilizes the remaining noisy samples to boost performance.
2. Distribution Bias: The selected clean samples may imbalance, since some labels are typically more ambiguous than others. Bias towards domain classes.
3. Confirmation Bias: Pseudo-labels bring confirmation bias.

Method/Mitigating Confirmation Bias

Motivation:
In most SSL pipelines, the generation and utilization of pseudo-labels are usually achieved by the same network blocks or between interdependent peer networks.
As a result, they are impossible to self-correct and can accumulate amplifying the existing confirmation bias.

The task-specific head that generates pseudo-labels for the unlabeled data but is merely optimized with labeled samples in Dl.

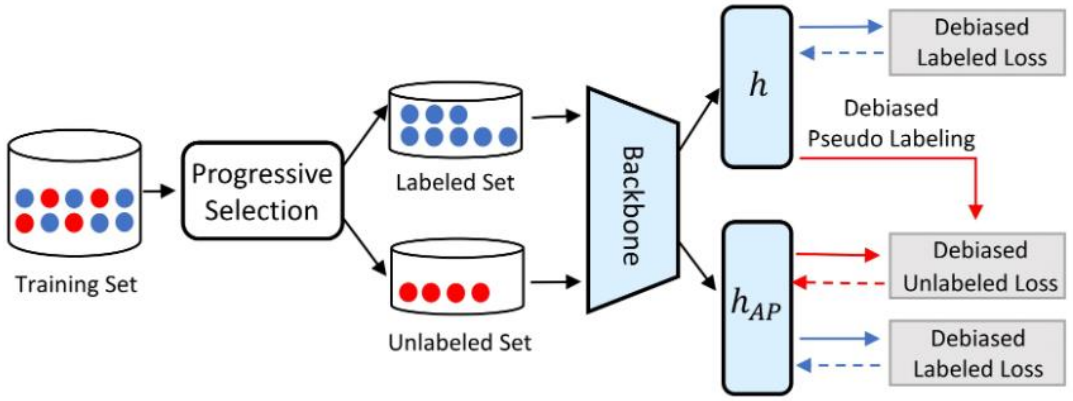


Share same representation with **h,** but independent parameters. It receives the pseudo-labels from h and is trained with both labeled set Dl and unlabeled set Du.

$$\mathcal{L}_{cls} = \underbrace{\mathcal{L}_x(\{\tilde{\boldsymbol{y}}_i, \boldsymbol{p}_i\})}_{\text{original head}} + \underbrace{\mathcal{L}_x(\{\tilde{\boldsymbol{y}}_i, \boldsymbol{p}'_i\}) + \lambda_u \mathcal{L}_u(\{\text{Sharpen}(\boldsymbol{p}_i, T), \boldsymbol{p}'_i\})}_{\text{auxiliary pseudo head}} \quad (4)$$

$$\text{Sharpen}(\boldsymbol{p}_i, T) = \frac{(p_i^c)^{\frac{1}{T}}}{\sum_{c=1}^{C}(p_i^c)^{\frac{1}{T}}} \text{ for } c = 1, 2, .., C.$$

Method/Mitigating Distribution Bias



$$\mathcal{L}_{\text{LDAM}}((x,y);f) = -\log \frac{e^{z_y-\Delta_y}}{e^{z_y-\Delta_y}+\sum_{j\neq y} e^{z_j}} \tag{12}$$

$$\text{where } \Delta_j = \frac{C}{n_j^{1/4}} \text{ for } j \in \{1,\dots,k\} \tag{13}$$

[1] Learning imbalanced datasets with label-distribution-aware margin loss. NeurlPS2019

Motivation:

1. skewed label distribution.
2. pseudo-labels can naturally bias towards some easy classes even if training data are balanced.

a Debiased Margin-based Loss (DML) to encourage a larger margin between the sample-rich classes and the sample-deficient classes.

$$l_{\text{DML}} = -\sum_{j=1}^{C} \tilde{y_i}^j \log \frac{e^{f^j(\boldsymbol{x}_i)+\alpha\cdot\log\pi_j}}{\sum_{k=1}^{C} e^{f^k(\boldsymbol{x}_i)+\alpha\cdot\log\pi_k}} \tag{5}$$

we suppress the logits on those easy classes and ensure other classes are fairly learned:

$$\tilde{f}_i = f(\boldsymbol{x}_i) - \alpha \log \boldsymbol{\pi} \tag{6}$$

$$\boldsymbol{\pi} = m\boldsymbol{\pi} + (1-m)\frac{1}{|\mathcal{B}|}\sum_{\boldsymbol{x}_i\in\mathcal{B}} \boldsymbol{p}_i \tag{7}$$

$$\mathcal{L}_{total} = \mathcal{L}_{cls} + \gamma(\mathcal{L}_{cr} + \mathcal{L}_{mix}) \tag{9}$$

Experiments

Dataset	CIFAR-10					CIFAR-100			
	Sym.				Asym.	Sym.			
Methods\Noise Ratio	20%	50%	80%	90%	40%	20%	50%	80%	90%
CE	86.8	79.4	62.9	42.7	85.0	62.0	46.7	19.9	10.1
Co-Teaching+	89.5	85.7	67.4	47.9	-	65.6	51.8	27.9	13.7
JoCoR	85.7	79.4	27.8	-	76.4	53.0	43.5	15.5	-
M-correction	94.0	92.0	86.8	69.1	87.4	73.9	66.1	48.2	24.3
PENCIL	92.4	89.1	77.5	58.9	88.5	69.4	57.5	31.1	15.3
DivideMix	96.1	94.6	93.2	76.0	93.4	77.3	74.6	60.2	31.5
ELR+	95.8	94.8	93.3	78.7	93.0	77.6	73.6	60.8	33.4
LongReMix	96.2	95.0	93.9	82.0	94.7	77.8	75.6	62.9	33.8
MOIT	94.1	91.1	75.8	70.1	93.2	75.9	70.1	51.4	24.5
SOP+	96.3	95.5	94.0	-	93.8	78.8	75.9	63.3	-
PES(semi)	95.9	95.1	93.1	-	-	77.4	74.3	61.6	-
ULC	96.1	95.2	94.0	86.4	94.6	77.3	74.9	61.2	34.5
ProMix(last)	**97.59**	**97.30**	**95.05**	**91.13**	**96.51**	**82.39**	**79.72**	**68.95**	**42.74**
ProMix(best)	**97.69**	**97.40**	**95.49**	**93.36**	**96.59**	**82.64**	**80.06**	**69.37**	**42.93**

Table 1: Accuracy comparisons on CIFAR-10/100 with symmetric (20%-90%) and asymmetric noise (40%). We report both the averaged test accuracy over last 10 epochs and the best accuracy of ProMix. Results of previous methods are the best test accuracies cited from their original papers, where the blank ones indicate that the corresponding results are not provided. **Bold entries** indicate superior results.

Experiments/Ablation

① LGA could boost performance, especially under severe noise

② Because of inadequate labeled samples.

③ Employs clean and disregard unselected.

④ Generating and utilizing the pseudo-labels on the same classification head.

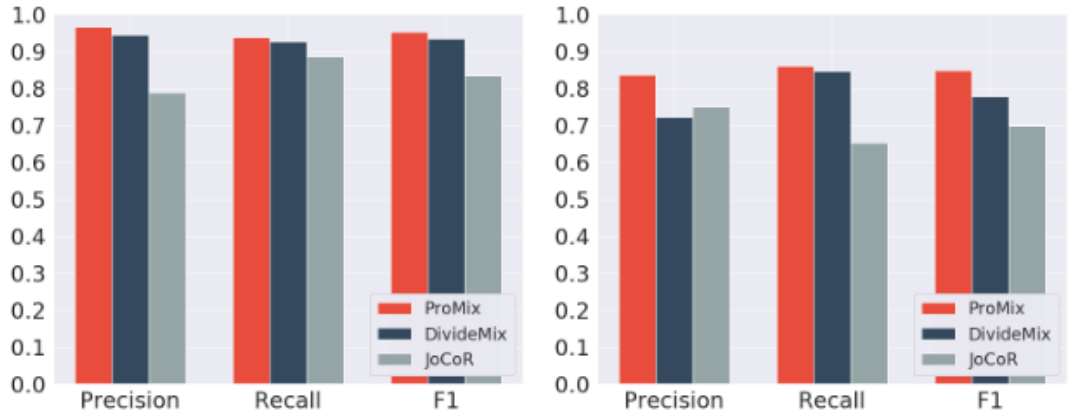
⑤ Sticks to the vanilla pseudo-labeling and adopts the standard cross entropy loss.

40.21% 40.20%

Ablation	LGA	Selection Strategy	CIFAR-10 Sym.80%	CIFAR-100 Sym.80%	CIFAR-10N Worst	CIFAR-100N Noisy Fine
ProMix	✓	CSS+MHCS	**95.05**	**68.95**	**96.34**	**73.79**
w/o LGA	✗	CSS+MHCS	94.32	61.19	95.57	73.10
w/o MHCS	✗	CSS	93.17	60.35	95.11	70.40
w/o Base Selection	✗	MHCS	39.59	21.01	74.09	40.75
w/o CBR	✗	CSS+MHCS	93.96	60.72	95.26	72.61
w/o DBR	✗	CSS+MHCS	94.07	60.91	95.07	72.51
with Only Clean	✗	CSS+MHCS	93.82	60.35	95.18	72.06

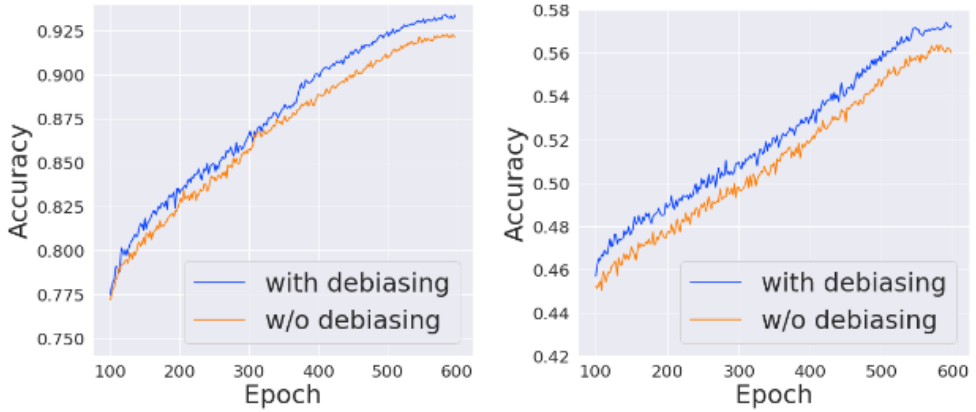
Table 5: Ablation study of ProMix on CIFAR-10-Symmetric 80%, CIFAR-100-Symmetric 80%, CIFAR-10N-Worst and CIFAR-100N-Noisy. ProMix with Only Clean denotes training with merely selected samples. CBR/DBR indicates Confirmation/Distribution Bias Removal.

Experiments/Ablation



(a) CIFAR-10-Symmetric 80%. (b) CIFAR-100-Symmetric 80%.

Figure 3: Comparison of clean sample selection on CIFAR-10/100 dataset with 80% symmetric noise. The threshold of DivideMix and forget rate of JoCoR have been re-tuned to get the best F1 score.



(a) CIFAR-10-Symmetric 80%. (b) CIFAR-100-Symmetric 80%.

Figure 4: Accuracy of pseudo-labels for the unlabeled data with and without debiasing on CIFAR-10/100 with 80% Symmetric noise. LGA is disabled in order to avoid unexpected interference.

Thank you