模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

On Learning Contrastive Representations for Learning with Noisy Labels

Li Yi[1] Sheng Liu[2] Qi She[3] A. Ian McLeod[1] Boyu Wang*[1, 4]

[1]University of Western Ontario, [2]NYU Center for Data Science

[3]ByteDance Inc. [4]Vector Institute

lyi7@uwo.ca shengliu@nyu.edu sheqi1991@gmail.com

aimcleod@uwo.ca bwang@csd.uwo.ca

CVPR 2022

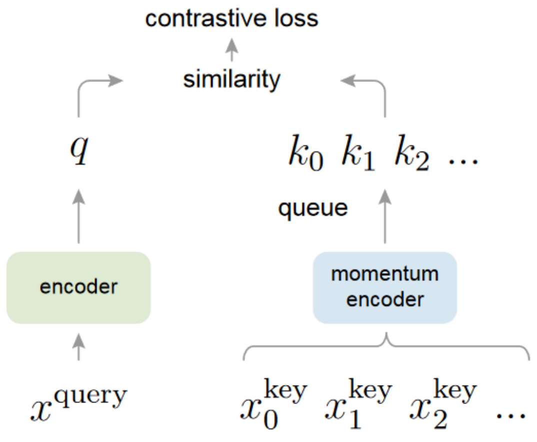
Background

Contrastive Learning Model Structure

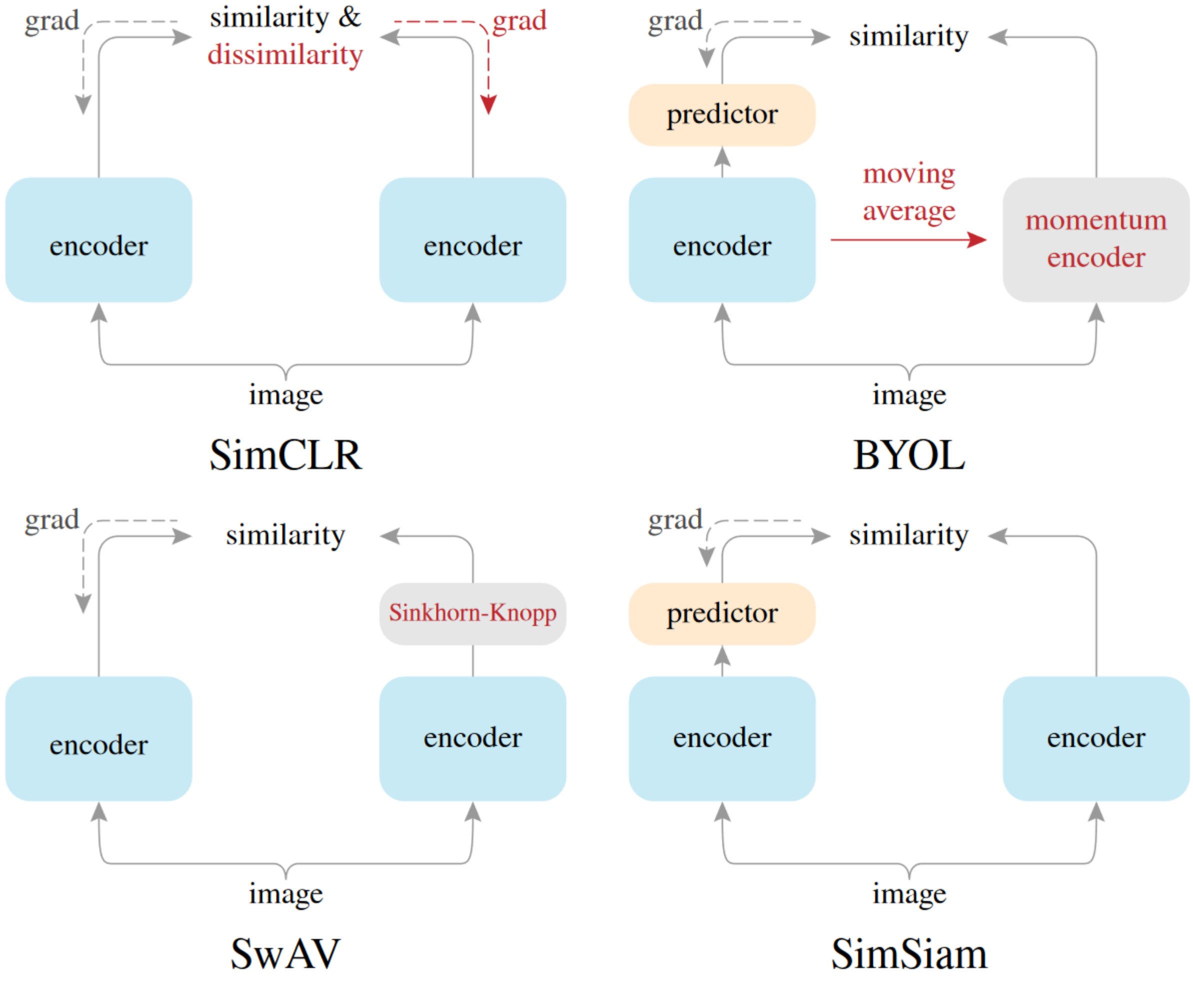
- Momentum encoder or sharing parameters

$$\theta_{\mathrm{k}} \leftarrow m\theta_{\mathrm{k}} + (1-m)\theta_{\mathrm{q}}.$$

- Use negative samples or not
- Additional predictor/projector
- Stop gradient or not



MoCo



Motivation

Loss design

- CE → Not robust to label noise
- Noise robust loss → suffer from the underfitting problem
- A trade-off → Explicitly or implicitly jointly used with the CE loss

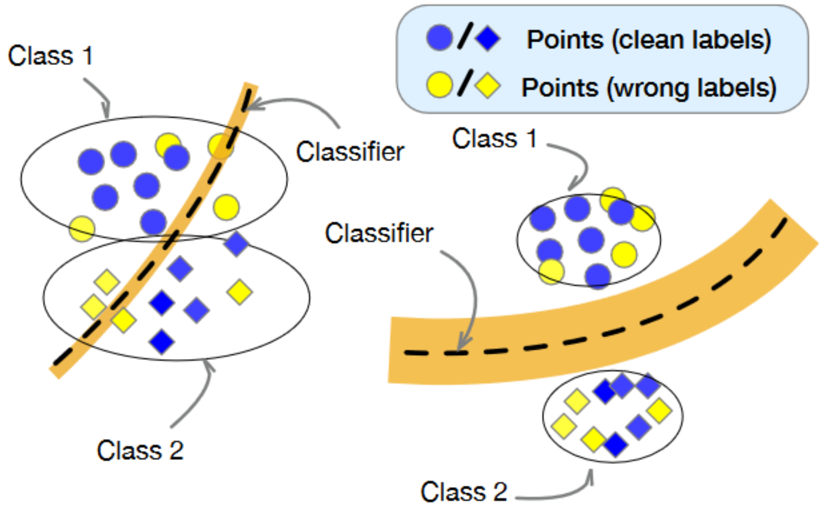


Figure 1. Illustration of the proposed method with noisy labels. Black curves are the best classifiers that are learned during training. **Left:** Deep networks without contrastive regularization. **Right:** Deep networks with contrastive regularization. Two classes are better separated by deep networks that points with the same class are pulled into a tight cluster and clusters are pushed away from each other.

Motivation

Representations Induced by Contrastive Regularization

- Key component

positive contrastive pair (x_1, x_2)

- Unsupervised CL

Correct positive contrastive pairs are formed from two different augmentations from the same image.

- Supervised CL

Correct positive contrastive pairs are formed by examples from the same class.

- When encountering with noisy labels ?

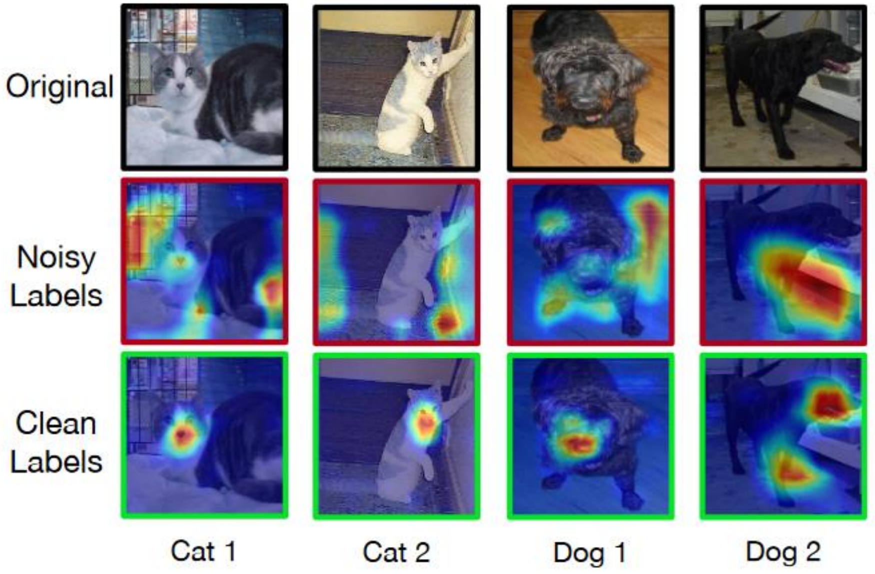


Figure 2. An example of Grad-CAM [35] results of Resnet34 trained on noisy dataset with 40% symmetric label noise and clean dataset, separately. When there is label noise, information related to corrupted labels captured by the model varies from image to image (e.g. window bars in Cat 1 v.s. floor and wall in Cat 2). When there is no label noise, information related to true labels are similar for images from the same class (e.g. cat face in Cat 1 v.s. cat face in Cat 2).

Method

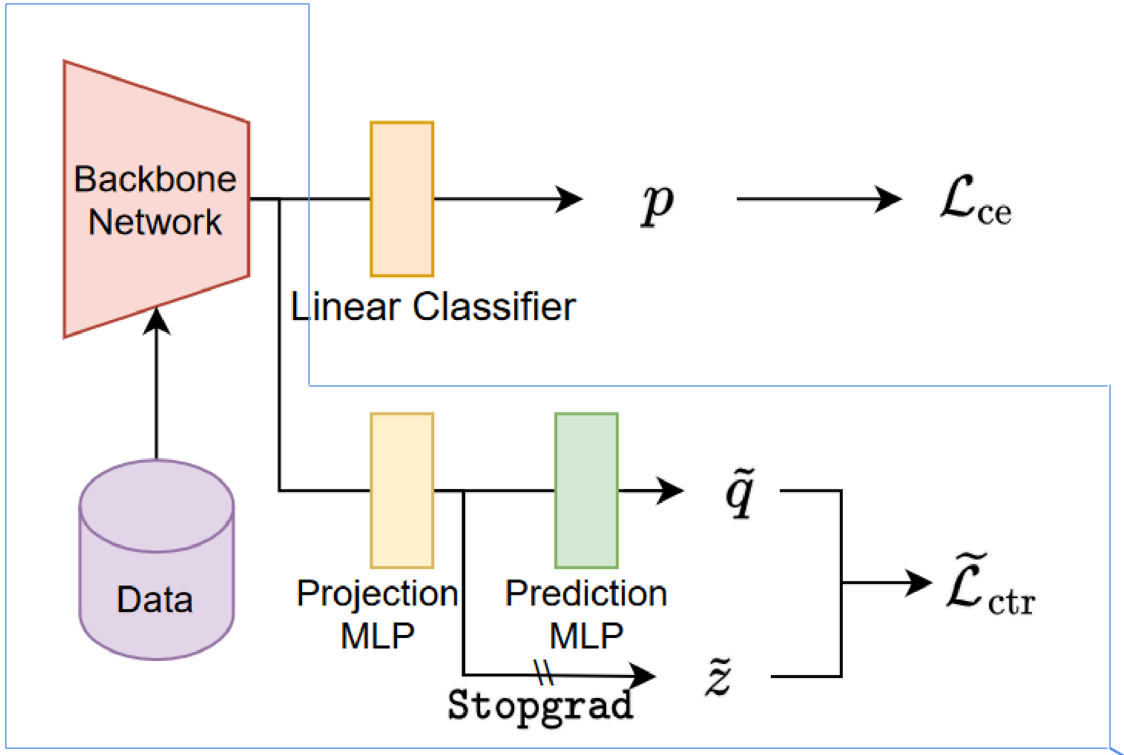


Figure 5. Illustration of our framework.

Simsiam

$$\mathcal{L} = \mathcal{L}_{\text{ce}} + \lambda \widetilde{\mathcal{L}}_{\text{ctr}},$$

Method

Design 1

Initial contrastive regularization function

$$\mathcal{L}_{\text{ctr}}(x_i, x_j) = -(\langle \tilde{q}_i, \tilde{z}_j \rangle + \langle \tilde{q}_j, \tilde{z}_i \rangle)\mathbb{1}\{y_i = y_j\}$$

Method

Design 2

Deep networks first fit examples with clean labels and the probabilistic outputs of these examples are higher than examples with corrupted labels.

$$\mathcal{L}'_{\text{ctr}}(x_i, x_j) = -\left(\langle \tilde{q}_i, \tilde{z}_j \rangle + \langle \tilde{q}_j, \tilde{z}_i \rangle\right) \mathbb{1}\boxed{\{p_i^\top p_j \geq \tau\}}$$

In early stage, $p_i^T p_j \approx 1$ for clean pair and 0 for noise pair

After this period?

Consider two clean examples x_i, x_j with clean label $y_i = y_j$

One wrongly labeled example x_m with $\tilde{y}_m = y_i = y_j$

$$\left\| \frac{\partial \mathcal{L}'_{\text{ctr}}(x_i, x_m)}{\partial q_i} \right\|_2^2 = c_i \underbrace{(1 - \tilde{q}_i^\top \tilde{q}_m)}_{\approx 1}$$

$$\gg c_i \underbrace{(1 - \tilde{q}_i^\top \tilde{q}_j)}_{\approx 0} = \left\| \frac{\partial \mathcal{L}'_{\text{ctr}}(x_i, x_j)}{\partial q_i} \right\|_2^2,$$

Method

Design 3

$$\widetilde{\mathcal{L}}_{\text{ctr}}(x_i, x_j) =$$
$$\Big(\log\left(1 - \langle \tilde{q}_i, \tilde{z}_j \rangle\right) + \log\left(1 - \langle \tilde{q}_j, \tilde{z}_i \rangle\right) \Big) \mathbb{1}\{p_i^\top p_j \geq \tau\}$$

$$\left\| \frac{\partial \widetilde{\mathcal{L}}_{\text{ctr}}(x_i, x_j)}{\partial q_i} \right\|_2^2 = c_i (1 + \tilde{q}_i^\top \tilde{q}_j)$$

$(1 + \tilde{q}_i^\top \tilde{q}_j > 1 + \tilde{q}_i^\top \tilde{q}_m \approx 1)$ → Does it overfit on clean data?

Experiment

Method	CIFAR-10						
	Sym.						Asym.
	0%	20%	40%	60%	80%	90%	40%
CE	$93.97_{\pm 0.22}$	$88.51_{\pm 0.17}$	$82.73_{\pm 0.16}$	$76.26_{\pm 0.29}$	$59.25_{\pm 1.01}$	$39.43_{\pm 1.17}$	$83.23_{\pm 0.59}$
Forward	$93.47_{\pm 0.19}$	$88.87_{\pm 0.21}$	$83.28_{\pm 0.37}$	$75.15_{\pm 0.73}$	$58.58_{\pm 1.05}$	$38.49_{\pm 1.02}$	$82.93_{\pm 0.74}$
GCE	$92.38_{\pm 0.32}$	$91.22_{\pm 0.25}$	$89.26_{\pm 0.34}$	$85.76_{\pm 0.58}$	$70.57_{\pm 0.83}$	$31.25_{\pm 1.04}$	$82.23_{\pm 0.61}$
Co-teaching	$93.37_{\pm 0.12}$	$92.05_{\pm 0.15}$	$87.73_{\pm 0.17}$	$85.10_{\pm 0.49}$	$44.16_{\pm 0.71}$	$30.39_{\pm 1.08}$	$77.78_{\pm 0.59}$
LIMIT	$93.47_{\pm 0.56}$	$89.63_{\pm 0.42}$	$85.39_{\pm 0.63}$	$78.05_{\pm 0.85}$	$58.71_{\pm 0.83}$	$40.46_{\pm 0.97}$	$83.56_{\pm 0.70}$
SLN	$93.21_{\pm 0.21}$	$88.77_{\pm 0.23}$	$87.03_{\pm 0.70}$	$80.57_{\pm 0.50}$	$63.99_{\pm 0.79}$	$36.64_{\pm 1.77}$	$81.02_{\pm 0.25}$
SL	$94.21_{\pm 0.13}$	$92.45_{\pm 0.08}$	$89.22_{\pm 0.08}$	$84.63_{\pm 0.21}$	$72.59_{\pm 0.23}$	$51.13_{\pm 0.27}$	$83.58_{\pm 0.60}$
APL	$93.97_{\pm 0.25}$	$92.51_{\pm 0.39}$	$89.34_{\pm 0.33}$	$85.01_{\pm 0.17}$	$70.52_{\pm 2.36}$	$49.38_{\pm 2.86}$	$84.06_{\pm 0.20}$
CTRR	$\mathbf{94.29}_{\pm 0.21}$	$\mathbf{93.05}_{\pm 0.32}$	$\mathbf{92.16}_{\pm 0.31}$	$\mathbf{87.34}_{\pm 0.84}$	$\mathbf{83.66}_{\pm 0.52}$	$\mathbf{81.65}_{\pm 2.46}$	$\mathbf{89.00}_{\pm 0.56}$

Table 1. Test accuracy on CIFAR-10 with different noise types and noise levels. All method use the same model PreAct ResNet18 and their best results are reported over three runs.

Method	CIFAR-100					
	Sym.					Asym.
	0%	20%	40%	60%	80%	40%
CE	$73.21_{\pm 0.14}$	$60.57_{\pm 0.53}$	$52.48_{\pm 0.34}$	$43.20_{\pm 0.21}$	$22.96_{\pm 0.84}$	$44.45_{\pm 0.37}$
Forward	$73.01_{\pm 0.33}$	$58.72_{\pm 0.54}$	$50.10_{\pm 0.84}$	$39.35_{\pm 0.82}$	$17.15_{\pm 1.81}$	-
GCE	$72.27_{\pm 0.27}$	$68.31_{\pm 0.34}$	$62.25_{\pm 0.48}$	$53.86_{\pm 0.95}$	$19.31_{\pm 1.14}$	$46.50_{\pm 0.71}$
Co-teaching	$73.39_{\pm 0.27}$	$65.71_{\pm 0.20}$	$57.64_{\pm 0.71}$	$31.59_{\pm 0.88}$	$15.28_{\pm 1.94}$	-
LIMIT	$65.53_{\pm 0.91}$	$58.02_{\pm 1.93}$	$49.71_{\pm 1.81}$	$37.05_{\pm 1.39}$	$20.01_{\pm 0.11}$	-
SLN	$63.13_{\pm 0.21}$	$55.35_{\pm 1.26}$	$51.39_{\pm 0.48}$	$35.53_{\pm 0.58}$	$11.96_{\pm 2.03}$	-
SL	$72.44_{\pm 0.44}$	$66.46_{\pm 0.26}$	$61.44_{\pm 0.23}$	$54.17_{\pm 1.32}$	$34.22_{\pm 1.06}$	$46.12_{\pm 0.47}$
APL	$73.88_{\pm 0.99}$	$68.09_{\pm 0.15}$	$63.46_{\pm 0.17}$	$53.63_{\pm 0.45}$	$20.00_{\pm 2.02}$	$52.80_{\pm 0.52}$
CTRR	$\mathbf{74.36}_{\pm 0.41}$	$\mathbf{70.09}_{\pm 0.45}$	$\mathbf{65.32}_{\pm 0.20}$	$\mathbf{54.20}_{\pm 0.34}$	$\mathbf{43.69}_{\pm 0.28}$	$\mathbf{54.47}_{\pm 0.37}$

Table 2. Test accuracy on CIFAR-100 with different noise levels. All method use the same model PreAct ResNet18 and their best results are reported over three runs.

Method	ANIMAL-10N	Clothing1M
CE	$83.18_{\pm 0.15}$	$70.88_{\pm 0.45}$
Forward	$83.67_{\pm 0.31}$	$71.23_{\pm 0.39}$
GCE	$84.42_{\pm 0.39}$	$71.34_{\pm 0.12}$
Co-teaching	$85.73_{\pm 0.27}$	$71.68_{\pm 0.21}$
SLN	$83.17_{\pm 0.08}$	$71.17_{\pm 0.12}$
SL	$83.92_{\pm 0.28}$	$72.03_{\pm 0.13}$
APL	$84.25_{\pm 0.11}$	$72.18_{\pm 0.21}$
CTRR	$\mathbf{86.71}_{\pm 0.15}$	$\mathbf{72.71}_{\pm 0.19}$

Table 3. Test accuracy on the real-world datasets ANIMAL-10N and Clothing1M. The results are obtained based on three different runs.

Experiment

Forward correction corrects loss values by a estimated noise transition matrix.
GCE takes advantages of both MAE loss and CE and designs a robust loss function.
Co-teaching maintains two networks and uses small-loss examples to update.
LIMIT introduces noise to gradients to avoid memorization.
SLN adds Gaussian noise to noisy labels to combat label noise.
SL uses CE loss and a reverse cross entropy loss (RCE) as a robust loss function.
APL (NCE+RCE) combines two mutually boosted robust loss functions for training.

Experiment

Regularization Functions	CIFAR-10 0%	20%	40%	60%	80%	90%
$\mathcal{L}'_{\text{ctr}}$(6)	$93.58_{\pm 0.11}$	$86.05_{\pm 0.33}$	$82.34_{\pm 0.25}$	$74.35_{\pm 0.54}$	$54.83_{\pm 1.00}$	$40.96_{\pm 0.99}$
$\widetilde{\mathcal{L}}_{\text{ctr}}$(8)	**$94.29_{\pm 0.21}$**	**$93.05_{\pm 0.32}$**	**$92.16_{\pm 0.31}$**	**$87.34_{\pm 0.84}$**	**$83.66_{\pm 0.52}$**	**$81.65_{\pm 2.46}$**

Table 4. The performance of the model with respect to different regularization functions.

$$\mathcal{L}'_{\text{ctr}}(x_i, x_j) = -(\langle \tilde{q}_i, \tilde{z}_j \rangle + \langle \tilde{q}_j, \tilde{z}_i \rangle)\mathbb{1}\{p_i^\top p_j \geq \tau\}, \quad (6)$$

$$\widetilde{\mathcal{L}}_{\text{ctr}}(x_i, x_j) = \Big(\log\left(1 - \langle \tilde{q}_i, \tilde{z}_j \rangle\right) + \log\left(1 - \langle \tilde{q}_j, \tilde{z}_i \rangle\right) \Big) \mathbb{1}\{p_i^\top p_j \geq \tau\} \quad (8)$$

Contrastive Frameworks	CIFAR-10 20%	40%	60%	80%	90%
CTRR (SimSiam)	$93.05_{\pm 0.32}$	$92.16_{\pm 0.31}$	$87.34_{\pm 0.84}$	$83.66_{\pm 0.52}$	$81.65_{\pm 2.46}$
CTRR (SimCLR)	$92.50_{\pm 0.35}$	$90.12_{\pm 0.43}$	$87.41_{\pm 0.83}$	$84.96_{\pm 0.44}$	$79.57_{\pm 1.32}$
CTRR (BYOL)	$93.31_{\pm 0.16}$	$92.12_{\pm 0.16}$	$88.71_{\pm 0.52}$	$86.99_{\pm 0.59}$	$84.31_{\pm 0.66}$

Table 5. Extending our method to other contrasitve learning frameworks.

Experiment

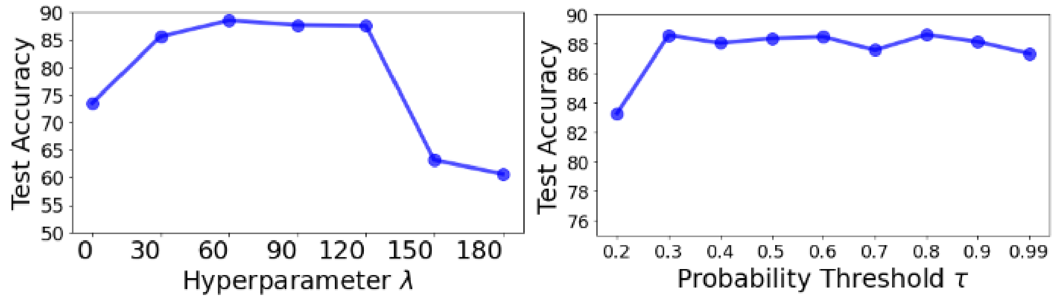


Figure 4. Analysis of λ and τ on CIFAR-10 with 60% symmetric label noise.

Label Correction Technique	CIFAR-10 20%	CIFAR-10 40%	CIFAR-10 60%	CIFAR-10 80%
✗	$93.05_{\pm 0.32}$	$92.16_{\pm 0.31}$	$87.34_{\pm 0.84}$	$83.66_{\pm 0.52}$
✓	$\mathbf{93.32}_{\pm \mathbf{0.11}}$	$\mathbf{92.76}_{\pm \mathbf{0.67}}$	$\mathbf{89.23}_{\pm \mathbf{0.18}}$	$\mathbf{85.40}_{\pm \mathbf{0.93}}$

Table 6. ✓/✗ indicates the label correction technique is enabled/disabled.

Method	CIFAR-10 20%	CIFAR-10 40%	CIFAR-10 60%	CIFAR-10 80%
GCE	$91.22_{\pm 0.25}$	$89.26_{\pm 0.34}$	$85.76_{\pm 0.58}$	$70.57_{\pm 0.83}$
CTRR	$93.05_{\pm 0.32}$	$92.16_{\pm 0.31}$	$87.34_{\pm 0.84}$	$83.66_{\pm 0.52}$
CTRR+GCE	$\mathbf{93.94}_{\pm \mathbf{0.09}}$	$\mathbf{93.06}_{\pm \mathbf{0.29}}$	$\mathbf{92.79}_{\pm \mathbf{0.06}}$	$\mathbf{90.25}_{\pm \mathbf{0.40}}$

Table 7. The performance of the model with respect to GCE, CTRR and CTRR+GCE.

Thanks