Appendix for "Towards Effective Visual Representations for Partial-Label Learning"

Shiyu Xia[1,*] Jiaqi Lv[2,*] Ning Xu[1] Gang Niu[2,1] Xin Geng[1,†]
[1]Southeast University [2]RIKEN Center for Advanced Intelligence Project
{shiyu_xia, xning, xgeng}@seu.edu.cn, {is.jiaqi.lv, gang.niu.ml}@gmail.com

1. Additional Ablation Results

The impact of encoder and projection network. We report that PaPi obtains better results with deeper encoder networks as shown in Table 1. In addition, we adopt two variants for the projection network: (1) 1-layer linear projection. (2) 2-layer MLP with one additional hidden layer and ReLU activation. We observe that PaPi performs better with 2-layer MLP, which is also found in contrastive learning literature [1–3].

Architecture	CIFAR-10 $q = 0.5$	CIFAR-100 $q = 0.1$
18-layer ResNet	96.90 ± 0.09%	81.65 ± 0.27%
34-layer ResNet	**97.55 ± 0.07%**	**82.51 ± 0.16%**
2-layer MLP	**96.90 ± 0.09%**	**81.65 ± 0.27%**
1-layer Linear	96.71 ± 0.05%	81.57 ± 0.12%

Table 1. Classification accuracy (mean ± std) under different encoders and projection networks on CIFAR-10 ($q = 0.5$) and CIFAR-100 ($q = 0.1$).

Prototype evolving factor γ. We report the results of varying γ that controls the prototype evolving formula in Table 2. On CIFAR-10 and CIFAR-100, the performance is stable with varying γ. Meanwhile, we find that some fixed values can achieve comparable results against dynamic value which ramps down from 0.9 to 0.5.

Disambiguation target updating factor λ. We report the results of varying λ that controls the disambiguation target updating speed in Table 3. The overall trends on both datasets tend to be stable. Specifically, PaPi achieves the best result on CIFAR-100 when $\lambda = 0.9$, and the performance slight drops when $\lambda = 0.99$.

Balancing factor φ. We explore the influence of the balancing factor by comparing the accuracy between fixed $\varphi \in \{0.1, 0.5, 1.0, 2.0, 5.0\}$ and dynamic φ with $\eta = 1.0$. From Fig. 1, we observe that dynamic balancing function gives better performance compared with the fixed one. When $\varphi = 5.0$, PaPi achieves inferior results on both two datasets.

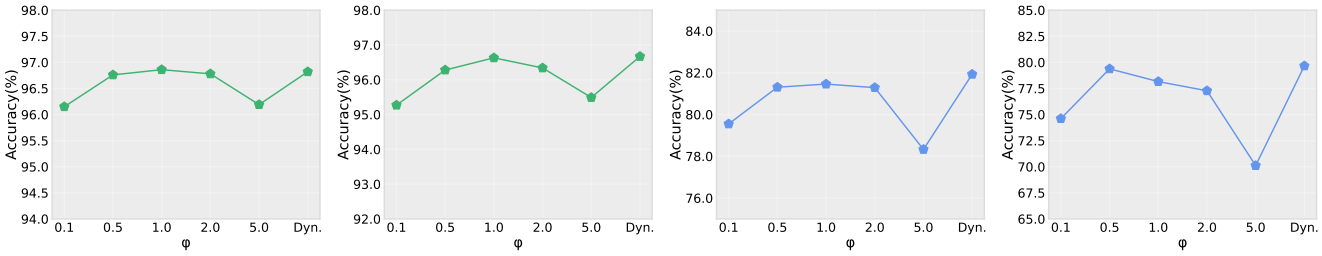
γ	CIFAR-10 $q = 0.5$	CIFAR-100 $q = 0.1$
0.1	96.77 ± 0.08%	81.04 ± 0.03%
0.5	96.91 ± 0.06%	81.02 ± 0.08%
0.9	96.88 ± 0.08%	81.64 ± 0.12%
0.99	**96.96 ± 0.02%**	81.52 ± 0.13%
dynamic	96.90 ± 0.09%	**81.65 ± 0.27%**

Table 2. Classification accuracy (mean ± std) under different γ on CIFAR-10 ($q = 0.5$) and CIFAR-100 ($q = 0.1$).

λ	CIFAR-10 $q = 0.5$	CIFAR-100 $q = 0.1$
0.1	96.41 ± 0.05%	81.08 ± 0.11%
0.5	**97.01 ± 0.06%**	81.44 ± 0.09%
0.9	96.81 ± 0.07%	**81.73 ± 0.10%**
0.99	96.42 ± 0.10%	81.34 ± 0.06%
dynamic	96.90 ± 0.09%	81.65 ± 0.27%

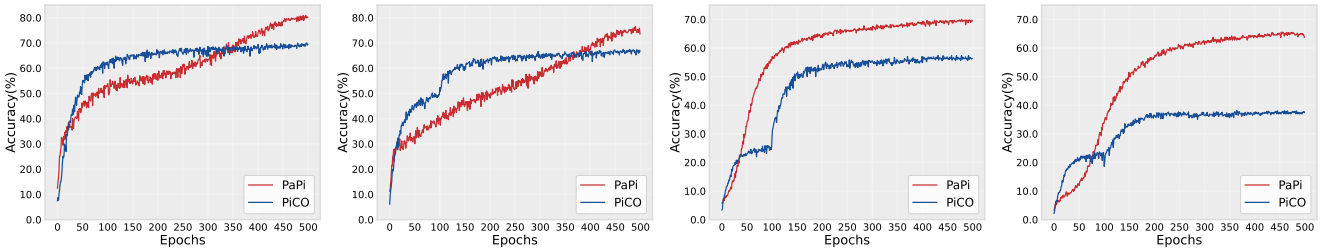
Table 3. Classification accuracy (mean ± std) under different λ on CIFAR-10 ($q = 0.5$) and CIFAR-100 ($q = 0.1$).

The performance of learned prototypical classifier. In Fig. 2, we report more results about the classification accuracy calculated with the learned prototypes. We can observe that PaPi achieves better performance especially facing *high ambiguity levels*, which demonstrates the competitiveness of our learned prototypical classifier. As is shown, PaPi outperforms PiCO by $\mathbf{10.75}\%$, $\mathbf{6.63}\%$, $\mathbf{13.33}\%$ and $\mathbf{26.29}\%$ respectively on CIFAR-100-H ($q = 0.5$), CIFAR-100-H ($q = 0.7$), Mini-Imagenet ($q = 0.1$) and Mini-Imagenet ($q = 0.2$).



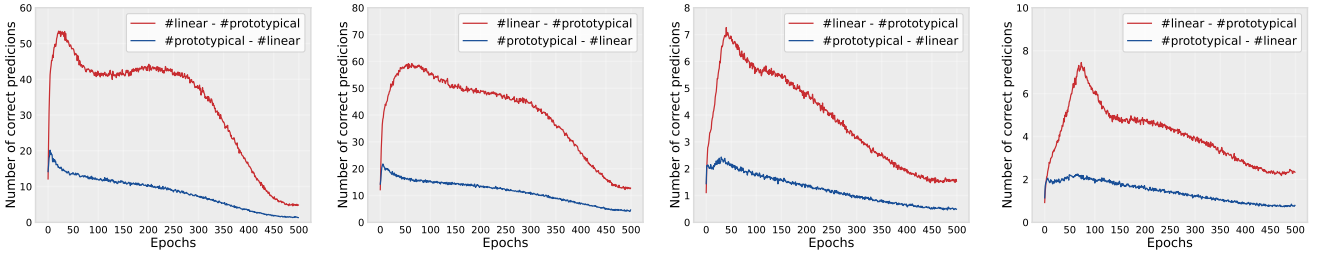
(a) CIFAR-10 ($q = 0.5$) (b) CIFAR-10 ($q = 0.7$) (c) CIFAR-100 ($q = 0.1$) (d) CIFAR-100 ($q = 0.2$)

Figure 1. Classification accuracy under different balancing factor choices on CIFAR-10 ($q = 0.5, 0.7$) and CIFAR-100 ($q = 0.1, 0.2$).



(a) CIFAR-100-H ($q = 0.5$) (b) CIFAR-100-H ($q = 0.7$) (c) Mini-Imagenet ($q = 0.1$) (d) Mini-Imagenet ($q = 0.2$)

Figure 2. Classification accuracy calculated with the learned prototypes on CIFAR-100-H ($q = 0.5, 0.7$) and Mini-Imagenet ($q = 0.1, 0.2$).



(a) CIFAR-100-H ($q = 0.5$) (b) CIFAR-100-H ($q = 0.7$) (c) Mini-Imagenet ($q = 0.1$) (d) Mini-Imagenet ($q = 0.2$)

Figure 3. PaPi learns with a correct disambiguation guidance. The red lines indicate the number of samples that were correctly classified by the linear classifier and incorrectly classified by the prototypical classifier per mini-batch, and the blue lines are the opposite.

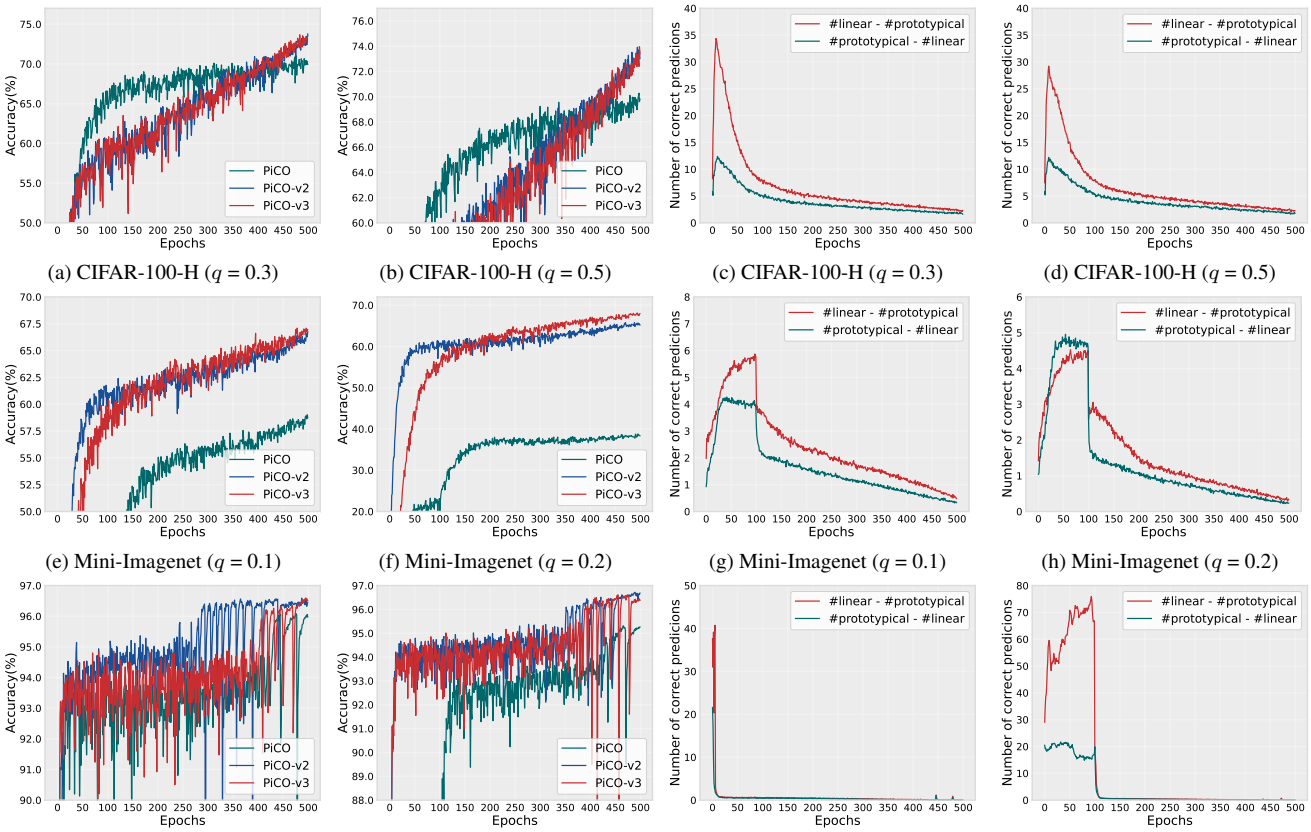
PaPi learns with a correct disambiguation guidance. Moreover, we present more empirical results about our proposed disambiguation guidance direction in Fig. 3. As is shown, we find that the linear classifier always makes more correct predictions than the prototypical classifier and the linear classifier always has something new to teach the prototypical classifier until convergence, which justifies the effectiveness of our disambiguation guidance.

2. Additional Results about Rethinking PiCO

In this section, we present more visualization results of the impacts of the unreliability of the pseudo positives and the improper direction of disambiguation guidance in PiCO. Fig. 4a, Fig. 4b, Fig. 4e, Fig. 4f, Fig. 4i and Fig. 4j show accuracy of three versions of PiCO. Fig. 4c, Fig. 4d, Fig. 4g, Fig. 4h, Fig. 4k and Fig. 4l show the performance differences between the linear and prototypical classifier during training. As is shown, PiCO-v2 outperforms PiCO in most cases, which verifies the significant performance degradation caused by noisy pseudo-labels. Moreover, PiCO-v3 outperforms PiCO-v2 in most cases, which confirms the importance of self-teaching fashion of the linear classifier. From Fig. 4c, Fig. 4d, Fig. 4g, Fig. 4h, Fig. 4k and Fig. 4l, we find that the number of samples that were correctly classified by the linear classifier is always larger than that were correctly classified by the prototypical classifier in PiCO, which illustrates the improper guidance direction.

References

[1] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of 37th International Conference on Machine*



(a) CIFAR-100-H ($q = 0.3$) (b) CIFAR-100-H ($q = 0.5$) (c) CIFAR-100-H ($q = 0.3$) (d) CIFAR-100-H ($q = 0.5$)

(e) Mini-Imagenet ($q = 0.1$) (f) Mini-Imagenet ($q = 0.2$) (g) Mini-Imagenet ($q = 0.1$) (h) Mini-Imagenet ($q = 0.2$)

(i) SVHN ($q = 0.3$) (j) SVHN ($q = 0.5$) (k) SVHN ($q = 0.3$) (l) SVHN ($q = 0.5$)

Figure 4. Visualizations of impacts of the unreliability of the pseudo positives and the improper direction of disambiguation guidance in PiCO. In (a)-(b), (e)-(f) and (i)-(j), PiCO-v2 means positives are selected based on fully supervised information, *i.e.*, true labels are known by the contrastive learning module. Further, PiCO-v3 removes the guidance of prototypical classifier to linear classifier, such that the linear classifier performs self-teaching. The red lines in (c)-(d), (g)-(h) and (k)-(l) indicate the number of samples that were correctly classified by the linear classifier and incorrectly classified by the prototypical classifier per mini-batch, and the green lines are the opposite. The first 100 epochs shown in (h) are in a warm-up period.

Learning (ICML'20), pages 1597–1607, 2020. 1

[2] X. Chen, H. Fan, R. Girshick, and K. He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1

[3] J. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. Ávila Pires, Z. Guo, M. Gheshlaghi Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko. Bootstrap your own latent - a new approach to self-supervised learning. *Advances in Neural Information Processing Systems 33 (NeurIPS'20)*, pages 21271–21284, 2020. 1