Supplementary Material for PEFAT: Boosting Semi-supervised Medical Image Classification via Pseudo-loss Estimation and Feature Adversarial Training

Qingjie Zeng[1*] Yutong Xie[2*] Zilin Lu[1] Yong Xia[1†]
[1] School of Computer Science and Engineering, Northwestern Polytechnical University, China
[2] The University of Adelaide, Australia
maxwell@mail.nwpu.edu.cn, yutong.xie678@gmail.com, luzl@mail.nwpu.edu.cn, yxia@nwpu.edu.cn

1. More Description of FAT and VAT

Algorithm 1 and Algorithm 2 present the pseudo-code of FAT and VAT in the Pytorch-like style, respectively. Here we can draw four observations: (1) VAT generates adversarial noise based on self-prediction, which may backfire when the original prediction is unsatisfied; (2) VAT simulates image-level adversarial noise, which may be learned by deep neural networks and thus show limited effects; (3) alternatively, in our work, we approximate the adversarial noise using two different views for the unselected low-confidence data. Therefore, we can mitigate the issue of undesirable adversarial noise caused by confirmation-biased self-prediction; (4) FAT injects feature-level adversarial noise, making the model adapt to a more difficult task, thereby boosts performance by better robustness.

2. Class-level AUC on Chest X-Ray14

Table 1 shows the performance comparision of class-level AUC on Chest X-Ray14 with label percentage of 20%. PEFAT achieves the best or second best class-level performance in 11 out of 14 categories. Specifically, compared to pseudo-labeling methods UPS and ACPL, our method has 0.81%$\sim$2.66% performance gain in averaged AUC even if the input image size is smaller. Beyond that, PETAF outperforms consistency-based method SRC-MT and self-supervised learning-based method S^2MTS^2 by 3.35% and 1.52%, repsectively.

3. Results on CIFAR-10 and CIFAR-100

We also conduct experiments on CIFAR-10 [2] and CIFAR-100 [2], following the same setup as [13]. Table 2 presents the results. We can find that PEFAT shows better performance on CIFAR-100 and is on par with other state-of-the-art methods on CIFAR-10. For most methods, there is little improvement on CIFAR-10 even if increasing the number of labeled data, and the main reason is that models generally make correct predictions on unlabeled data, making the results close to the fully supervised baseline. As for CIFAR-100, a relatively complicated dataset, PEFAT consistently achieves the best results under different settings, demonstrating its superiority of unlabeled data mining.

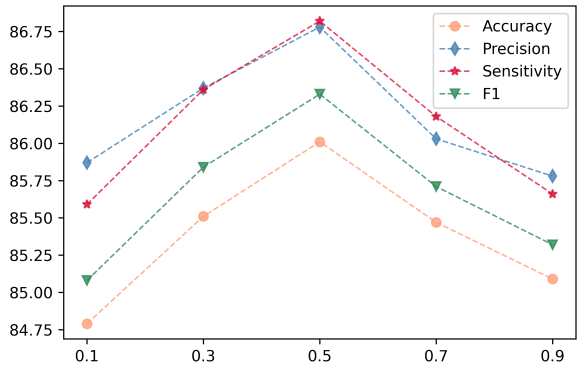

Figure 1. Four evaluation metrics under different η, which are conducted on NCT-CRC-HE dataset.

4. Discussion of Hyper-parameter η

In the proposed PEFAT, η is a hyper-parameter to balance the weight between two cross pseudo-losses (see Eq. (10)). As shown in Figure 1, we conduct a grid search of $\eta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ on NCT-CRC-HE dataset. We can find that PEFAT achieves the best performance when $\eta = 0.5$. This result indicates that it is the most effective to select trustworthy pseudo-labeled data when treating two cross pseudo-losses equally.

*Equal contribution. †Corresponding author.

5. Limitation

The proposed FAT utilizes loss backpropagation to approximate the feature-level adversarial noise, which may require higher demand for computing and storage on dense prediction tasks, $i.e.$, medical image segmentation. But for classification, the aforementioned cost can be almost ignored, since the classification head is lightweight.

References

[1] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, pages 4700–4708, 2017. 3

[2] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 1, 3

[3] Junnan Li, Caiming Xiong, and Steven CH Hoi. Comatch: Semi-supervised learning with contrastive graph regularization. In *ICCV*, pages 9475–9484, 2021. 3

[4] Fengbei Liu, Yu Tian, Yuanhong Chen, Yuyuan Liu, Vasileios Belagiannis, and Gustavo Carneiro. Acpl: Anti-curriculum pseudo-labelling for semi-supervised medical image classification. In *CVPR*, pages 20697–20706, 2022. 3

[5] Fengbei Liu, Yu Tian, Filipe R Cordeiro, Vasileios Belagiannis, Ian Reid, and Gustavo Carneiro. Self-supervised mean teacher for semi-supervised chest x-ray classification. In *International Workshop on Machine Learning in Medical Imaging*, pages 426–436, 2021. 3

[6] Quande Liu, Lequan Yu, Luyang Luo, Qi Dou, and Pheng Ann Heng. Semi-supervised medical image classification with relation-driven self-ensembling model. *IEEE Transactions on Medical Imaging*, 39(11):3429–3440, 2020. 3

[7] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. In *ICLR*, 2020. 3

[8] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, pages 596–608, 2020. 3

[9] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, pages 1195–1204, 2017. 3

[10] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *NeurIPS*, pages 6256–6268, 2020. 3

[11] Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised learning with dynamic thresholding. In *ICML*, pages 11525–11536, 2021. 3

[12] Fan Yang, Kai Wu, Shuyi Zhang, Guannan Jiang, Yong Liu, Feng Zheng, Wei Zhang, Chengjie Wang, and Long Zeng. Class-aware contrastive semi-supervised learning. In *CVPR*, pages 14421–14430, 2022. 3

[13] Mingkai Zheng, Shan You, Lang Huang, Fei Wang, Chen Qian, and Chang Xu. Simmatch: Semi-supervised learning with similarity matching. In *CVPR*, pages 14471–14481, 2022. 1, 3

Table 1. Performance of **class-level AUC** on Chest X-Ray14 dataset under the label percentage of **20%**. Note that * denotes the methods employee DenseNet-169 as backbone with 384×384 input size, † means the methods use DenseNet-121 as backbone with 512×512 input size. Our method takes DenseNet-121 as backbone with input size of 224×224.

Methods	Baseline [1]	MT [9]	SRC-MT* [6]	UPS [7]	$S^2MTS^{2\dagger}$ [5]	APCL† [4]	Ours
Atelectasis	75.75	75.12	75.38	77.09	78.57	79.53	**79.77**
Cardiomegaly	80.71	87.37	87.70	85.73	88.08	**89.03**	88.52
Effusion	79.87	80.81	81.58	81.35	82.87	83.56	**86.30**
Infiltration	69.16	70.67	70.40	70.82	70.68	**71.40**	70.98
Mass	78.40	77.72	78.03	81.82	**82.57**	82.49	81.56
Nodule	74.49	73.27	73.64	76.34	76.60	77.73	**77.82**
Pneumonia	69.55	69.17	69.27	70.96	72.25	73.86	**74.18**
Pneumothorax	84.70	85.63	86.12	85.86	86.55	86.95	**87.80**
Consolidation	71.85	72.51	73.11	74.35	75.47	75.50	**78.92**
Edema	81.61	82.72	82.94	83.56	84.83	84.95	**87.43**
Emphysema	89.75	88.16	88.98	91.00	91.88	**93.36**	92.86
Fibrosis	79.30	78.24	79.22	80.87	81.73	**81.86**	80.94
Pleural Thicken	73.46	74.43	75.63	75.55	76.86	**77.60**	75.42
Hernia	86.05	87.74	87.27	85.62	85.98	85.89	**90.20**
Mean	78.19	78.83	79.23	79.92	81.06	81.77	**82.58**

Table 2. Accuracy comparison (mean and std over 5 runs) with other state-of-the-art SSL methods on CIFAR-10 [2] and CIFAR-100 [2].

Method	CIFAR-10		CIFAR-100	
	250 labels	4000 labels	2500 labels	10000 labels
MT [9]	67.68±2.30	90.81±0.19	46.09±0.57	64.17±0.24
UDA [10]	91.18±1.08	95.12±0.18	66.87±0.22	75.50±0.25
FixMatch [8]	94.93±0.65	95.74±0.05	71.71±0.11	77.40±0.12
Dash [11]	95.44±0.13	95.92±0.06	72.82±0.21	78.03±0.14
CoMatch [3]	95.09±0.33	95.44±0.20	71.63±0.35	79.14±0.36
CCSSL [12]	94.86±0.55	95.54±0.20	75.70±0.63	80.68±0.16
SimMatch [13]	95.16±0.39	**96.04±0.01**	74.93±0.32	79.42±0.11
Ours	**95.52±0.37**	95.98±0.03	**75.92±0.56**	**80.88±0.32**

Algorithm 1: Pseudo-code of Feature Adversarial Training in a Pytorch-like style

```
# f: encoder; cls: classifier
# KL: Kullback-Leibler Divergence; norm_l2: L2 Normalization
for x, _ in unselected_unlabeled_loader:
    x1, x2 = strong_aug1(x), strong_aug2(x)
    z1, z2, logit1, logit2 = f(x1), f(x2), cls(f(x1)), cls(f(x2))
    p1, p2 = torch.softmax(logit1, dim=1), torch.softmax(logit2, dim=1)

    # initialize two random noises
    d1 = torch.normal(0, 1, size = z1.shape)
    d2 = torch.normal(0, 1, size = z2.shape)
    d1, d2 = norm_l2(d1), norm_l2(d2)

    # apply random noises
    z_d1 = torch.tensor(z1.clone().detach().cpu().data + d1, requires_grad=True)
    z_d2 = torch.tensor(z2.clone().detach().cpu().data + d2, requires_grad=True)
    logit_d1, logit_d2 = cls(z_d1), cls(z_d1)
    p_d1, p_d2 = torch.softmax(logit_d1, dim=1), torch.softmax(logit_d2, dim=1)

    # generate feature-level adversarial noises
    loss_kl = KL(p_d1, p2) + KL(p_d2, p1)
    loss_kl.backward(retain_graph=True)
    adv1, adv2 = norm_l2(z_d1.grad), norm_l2(z_d2.grad)

    # calculate FAT loss
    logit_adv1, log_adv2 = cls(z1 + adv1), cls(z2 + adv2)
    p_adv1, p_adv2 = torch.softmax(logit_adv1, dim=1), torch.softmax(log_adv2, dim=1)
    loss_FAT = KL(p_adv2, p1) + KL(p_adv1, p2)
```

Algorithm 2: Pseudo-code of Vitural Adversarial Training in a Pytorch-like style

```
# M: encoder + classifier;
for x, _ in unlabeled_loader:
    x = strong_aug(x)
    logit = M(x)
    p = torch.softmax(logit, dim=1)

    # initialize random noise
    d = torch.normal(0, 1, size = x.shape)
    d = norm_l2(d)

    # apply random noise
    x_d = torch.tensor(x + d, requires_grad=True)
    logit_d = M(x_d)
    p_d = torch.softmax(logit_d, dim=1)

    # generate image-level adversarial noise
    loss_kl = KL(p_d, p)
    loss_kl.backward(retain_graph=True)
    adv = norm_l2(x_d.grad)

    # calculate VAT loss
    logit_adv = M(x + adv)
    p_adv = torch.softmax(logit_adv, dim=1)
    loss_VAT = KL(p_adv, p)
```