Universal Semi-supervised Model Adaptation via Collaborative Consistency Training Supplementary File

Zizheng Yan[1,2*] Yushuang Wu[1,2*] Yipeng Qin[3] Xiaoguang Han[1,2]
Shuguang Cui[1,2] Guanbin Li[4,5†]

[1]FNii, CUHKSZ [2]SSE, CUHKSZ [3]Cardiff University [4]Sun Yat-sen University
[5]Research Institute, Sun Yat-sen University, Shenzhen

Contents

1. More Analysis

1.1. Augmentation Strategies.

We denote standard random resized crop and flip as weak augmentation and RandAugment [2] as strong augmentation, and experimentally study how the performance of our method changes with different augmentation combinations. We conduct the experiments on the task of *Real* → *Clipart* and *Real* → *Sketch* of *Domainnet*. As Table 1 shows, the weak-strong combination yields the best performance in terms of H-score. We conjecture that the weakly augmented view produces more stable predictions for pseudo labeling, while training with the strongly augmented input prevents the network from overfitting pseudo label noises.

Table 1. *H-score* w.r.t. augmentation strategies in task R → C and R → S on *Domainnet*.

x'	x''	R → C	R → S
Strong	Strong	75.6	62.1
Strong	Weak	60.0	56.7
Weak	Weak	72.8	61.3
Weak	Strong	**77.7**	**66.8**

1.2. Robustness against Hyper-parameter τ.

We explore the sensitivity of our method with respect to the choice of hyper-parameter, *i.e.*, threshold τ used in sample-wise consistency regularization. As Table 2 shows, our method (CCT) outperforms FixMatch [7] for all choices of τ. Moreover, it can be observed that our CCT is less sensitive to the thresholds compared with FixMatch [7]. In addition, more analysis of hyper-parameters can be found in the supplementary material.

Table 2. *H-score* w.r.t. threshold τ in task R → C on *Domainnet*.

	Threshold τ				
Method	0.35	0.5	0.65	0.8	0.95
FixMatch	66.9	65.7	67.9	70.4	68.6
CCT	**73.7**	**73.5**	**73.4**	**76.3**	**77.7**

1.3. Accuracy of Common-Private Set Samples

In addition to H-score, we also report the accuracy of common and private set samples on *Domainnet*. As Table 3 shows, the accuracy of private set significantly outperforms all compared methods. It is worth noting that i) the performance gap between common set accuracy of FixMatch and CCT is relatively small, ii) while for the private set accuracy, the gap is significant larger, which implies that the CCT can effectively improve the performance of $F(\cdot|\theta_s)$ on private set.

1.4. Pseudo Label Generation Strategy

We empirically validate multiple pseudo label generation strategies, *e.g.*, *ensemble* and *weighted ensemble*, where *ensemble* refers to the strategy where pseudo labels are generated by thresholding $[F(x'|\theta_s) + F(x'|\theta_t)]/2$, and *weighted ensemble* refers to the strategy of thresholding $[wF(x'|\theta_s) + (1-w)F(x'|\theta_t)]$, where $w \in [0,1]$ is com-

[*] Equal contributions.
[†] Corresponding Author.

Table 3. Accuracy of Common ($a_{\mathcal{C}}$) and Private ($a_{\mathcal{P}}$) set on the *DomainNet* 5-shot setting using ResNet34 as the backbone.

Method	R → C		P → C		C → S		R → P		S → P		R → S		**Mean**	
	$a_{\mathcal{C}}$	$a_{\mathcal{P}}$	$a_{\mathcal{C}}$	$a_{\mathcal{P}}$	$a_{\mathcal{C}}$	$a_{\mathcal{P}}$	$a_{\mathcal{C}}$	$a_{\mathcal{P}}$	$a_{\mathcal{C}}$	$a_{\mathcal{P}}$	$a_{\mathcal{C}}$	$a_{\mathcal{P}}$	$a_{\mathcal{C}}$	$a_{\mathcal{P}}$
CE	73.6	47.9	72.2	47.4	65.5	46.8	67.3	50.4	67.3	51.4	65.0	41.4	68.5	47.6
MME	74.8	59.8	73.7	56.7	62.4	49.5	63.1	61.6	65.8	61.9	62.4	43.4	67.0	55.5
FixMatch	82.2	58.9	81.8	55.9	72.7	52.8	71.5	65.7	73.9	63.4	73.1	57.1	75.9	59.0
CCT	**83.0**	**73.1**	**81.6**	**73.6**	**74.1**	**60.8**	**77.3**	**73.8**	**75.8**	**74.7**	**75.1**	**60.2**	**77.8**	**69.4**

Table 4. Results of different pseudo label generation strategies for sample-wise consistency on *Domainnet*

Consistency Loss	R → C	C → S
Ensemble	75.3	65.6
Weighted ensemble	75.4	65.6
Sample-wise	**77.7**	**66.8**

Table 5. Comparison of class-wise consistency with MIM and MCC on *Domainnet* 5-shot setting

Method	R → C	C → S
MIM	55.2	51.0
MCC	65.8	59.0
Class-wise (ours)	**70.6**	**60.6**
Sample-wise + MIM	68.7	62.6
Sample-wise + MCC	71.7	64.4
CCT (ours)	**77.7**	**66.8**

Table 6. H-score of R → P and S → P on *Domainnet* 5 settings.

Method	R → P	S → P
RotPred	71.8	73.9
SCL+SimCLR	**75.5**	**75.3**

puted by the entropy of the two networks. As Table 4 shows, our L_{sample} is more effective than the above methods.

1.5. Analysis of Class-wise Consistency

As we have mentioned in Section. 3.4, class-wise consistency has much more merits than the previous regularization functions like Mutul Information Maximization (MIM) [6] and Minimum Class Confution (MCC) [4]. In addition, we experimentally compare class-wise consistency with MIM and MCC on *Domainnet* 5-shot setting. As Table. 5 shows, class-wise consistency achieves much superior performance than MIM and MCC.

1.6. Justification of Our $F(\cdot|\theta_t)$ Pre-training Method

To justify our choice of the pre-training method, SCL [5] + SimCLR [1], for $F(\cdot|\theta_t)$, we compare it with rotation prediction [3] on *Real* → *Painting* and *Sketch* → *Painting* of *Domainnet*. As shown in Table 6, our SCL [5] + SimCLR [1] yields better results than rotation prediction [3].

Table 7. H-score on the *Domainnet* 5-shot setting.

Method	R → C	P → C	C → S	R → P	S → P	R → S	**Mean**
$F(\cdot\|\theta_s)$	77.7	77.4	66.8	75.5	75.3	66.9	**73.3**
$F(\cdot\|\theta_t)$	77.9	77.4	66.7	75.5	75.1	66.8	73.2

Table 8. (a) Average H-score w.r.t. loss weight λ_1 on the *Office-Home* 5-shot setting. Note that λ_2 is fixed to 0.5. (b) Average H-score w.r.t. loss weight λ_2 on the *Office-Home* 5-shot setting. Note that λ_1 is fixed to 1.

λ_1	0	0.5	1	1.5
H-score	72.3	73.5	73.5	72.8

(a)

λ_2	0	0.5	1	1.5
H-score	70.5	73.5	73.3	72.6

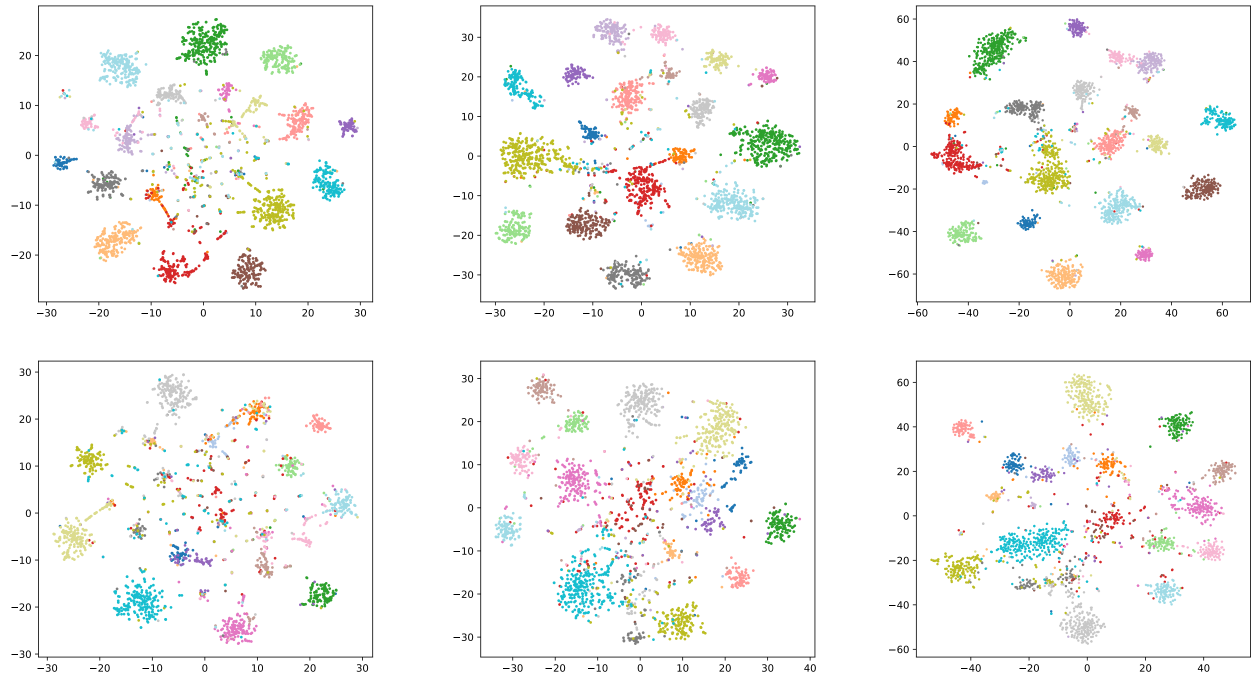
(b)

1.7. Justification of Choice between $F(\cdot|\theta_s)$ and $F(\cdot|\theta_t)$

As mentioned in the main paper, the performance of $F(\cdot|\theta_s)$ and $F(\cdot|\theta_t)$ will converge to the same point after training and thus we simply choose $F(\cdot|\theta_s)$ as the final model. To support our choice, we show the H-score of $F(\cdot|\theta_s)$ and $F(\cdot|\theta_t)$ on *Domainnet* in Table 7. It can be observed that the performance of $F(\cdot|\theta_s)$ and $F(\cdot|\theta_t)$ are very close, and $F(\cdot|\theta_s)$ performs slightly better, which justifies our choice.

1.8. Analysis of Hyper-parameters

The proposed CCT has three hyper-parameters, *i.e.* the threshold τ, the loss weight of sample-wise consistency λ_1, and the loss weight of class-wise consistency λ_2. In this section, we study how λ_1 and λ_2 influence the performance since τ has been studied in the main paper. We conduct the experiments on *Office-Home*. As Table 8 shows, we achieve the best performance when $\lambda_1 = 1$ and $\lambda_2 = 0.5$. It can be observed that when $\lambda_2 = 0$, *i.e.* without sample-wise consistency, the performance is much lower, which demonstrates the efficacy of our sample-wise consistency. Furthermore, it can be observed that the model is not sensitive w.r.t λ_1 and λ_2.



(a) MME. (b) FixMatch. (c) CCT.

Figure 1. TSNE [8] visualization of the learned features of common (top) and private (bottom) sets samples respectively. We randomly sample 20 classes for both sets in task R → C on *Domainnet*.

Table 9. Mean and standard deviation of H-score over five runs on *Domainnet* in the 5-shot setting. Note that **Avg** denotes the Mean and standard deviation of the mean H-score over five runs.

Method	R → C	P → C	C → S	R → P	S → P	R → S	**Avg**
Mean	77.8	77.4	66.4	75.1	75.4	66.3	73.1
STD	0.4	0.3	0.3	0.8	0.6	0.4	0.2

1.9. Analysis of Performance Stability

We investigate the performance stability of the proposed CCT in multiple runs. Table 9 shows the results of averaged H-score and the standard deviation of five runs on *Domainnet* in the 5-shot setting. The standard deviation of the averaged H-score is very small, *i.e.* 0.2, demonstrating the stability of our CCT.

1.10. Feature Visualization

In addition to the quantitative results, we also show the qualitative results of the learned features by TSNE [8]. As Fig. 1 shows, the features of CCT are more compact and form into well-separated clusters in both common and private sets, which verifies that our CCT learns more discriminative features.

References

[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 2

[2] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020. 1

[3] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018. 2

[4] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *European Conference on Computer Vision*, pages 464–480. Springer, 2020. 2

[5] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020. 2

[6] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020. 2

[7] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised

learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020. 1

[8] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 3