Supplementary Material: Distilling and Refining Domain-Specific Knowledge for Semi-Supervised Domain Adaptation

Ju Hyun Kim[1]
kjhyun18@dgu.ac.kr
Ba Hung Ngo[1]
ngohung@dgu.ac.kr
Jae Hyeon Park[1]
pjh0011@dongguk.edu
Jung Eun Kwon[1]
kje_9912@dgu.ac.kr
Ho Sub Lee[2]
hslee34@daegu.ac.kr
Sung In Cho[1]
csi2267@dongguk.edu

[1] Department of Multimedia Engineering
Dongguk University
Seoul, Korea

[2] Department of Electronic Engineering
Daegu University
Gyeongsan, Korea

A. Additional Experimental Results

We report extensive experimental results in various ways.

Experimental results on varying shots of labeled target data. To provide the experimental results using ResNet34 [1] for more shots of labeled target data, we use five- and ten-shot settings of labeled target samples on DomainNet [7] as in APE [3] As shown in Table 6, the proposed DARK achieves state-of-the-art (SOTA) performances in all domain adaptation tasks. Specifically, DARK surpasses the second-best method, CDAC [5], by 2.5% and 1.9% in the five- and ten-shot settings, respectively.

Experimental results with various backbone networks. We provide the additional experimental results of the benchmark methods and the proposed method, extracted with various backbone networks such as AlexNet [4] and VGG-16 [10] for the three-shot setting on Office-Home [10]. The comparison results are reported in Table 7. DARK shows the highest classification accuracy. Specifically, the classification accuracy of DARK is improved by 0.6% compared to the second-best approach, CLDA [11] for AlexNet. Besides, DARK provides 0.2% higher average classification accuracy than the recent SOTA approach, ASDA [3], for VGG-16.

# Shot	Method	R → C	R → P	P → C	C → S	S → P	R → S	P → R	Mean
5-shot	MME [31]	75.5	70.4	74.0	65.0	68.2	65.5	79.9	71.2
	APE [25]	77.7	73.0	76.9	67.0	71.4	68.8	80.5	73.6
	CLDA [14]	80.3	76.0	77.8	71.6	74.5	72.9	84.0	76.7
	CDAC [25]	80.8	75.3	79.9	72.1	74.7	72.9	83.2	76.9
	DARK (ours)	**82.0**	**78.8**	**82.1**	**75.0**	**77.9**	**73.8**	**86.4**	**79.4**
10-shot	MME [31]	77.1	71.9	76.3	67.0	69.7	67.8	81.2	73.0
	APE [25]	79.8	75.1	78.9	70.5	73.6	70.8	82.9	76.8
	CLDA [14]	81.2	77.7	80.3	74.1	77.1	74.1	85.1	78.5
	CDAC [25]	**83.1**	77.2	81.7	74.3	76.3	74.6	84.7	78.9
	DECOTA [13]	81.8	75.1	81.3	73.7	73.4	73.7	80.7	77.1
	DARK (ours)	**83.1**	**79.7**	**82.4**	**75.6**	**79.0**	**75.3**	**87.8**	**80.4**

Table 6: Quantitative results (%) on DomainNet of the five- and ten-shot settings. The best accuracy is indicated in bold.

B. Additional Ablation Studies

Ablation for the *Distilling* strategy with SDWR. Table 8 shows the effectiveness of the soft label (SL) with the sample-wise dynamic weight based on prediction reliability (SDWR) in the *Distilling* strategy. Specifically, we analyze the effect of SL-based *Distilling* on DomainNet over the three scenarios, in which the extracted results with SL-based *Distilling* are compared to the hard label (HL)-based *Distilling*. We use a one-hot encoded pseudo label as HL by setting the threshold value to 0.9. The threshold value is chosen that could provide the best DA performance.

1) When using only *Distilling* (cases 1, 2 and 4): SL-based *Distilling* (case 2) shows slightly lower than 0.4% compared to HL-based *Distilling* (case 1) on the average accuracy. This is because the negative effects of SL cannot be completely minimized. However, if we use the LS for SL-based *Distilling* (case 4), we can successfully reduce the negative effect and obtain better accuracy than HL-based *Distilling* (case 1).

2) When using both *Refining* and *Distilling* (cases 5, 6 and 8): SL-based *Distilling* (case 5) shows higher or similar performance for three scenarios compared with HL-based *Distilling* (case 6). HL-based *Distilling* cannot achieve the highest performance due to the confirmation bias and poor knowledge transfer. In case of our proposed method (case 8), it shows the highest performance for the three scenarios in Table 8 through *Refining*.

3) The comparison with the performance of the proposed method without and with SDWR for *Distilling* (cases 3, 4, 7 and 8): When using SL-based *Distilling* without SDWR (case 3 and 7), the performance is decreased because the model cannot handle the negative effect of low confident samples. We can observe that when SDWR is applied (case 4 and 8), SL-based *Distilling* becomes more stable; therefore, the classification performance is increased.

Ablation for SDWR of the bridging loss. To prove the contribution of SDWR (in *Refining*) for the classification performance, we compare the extracted results of the proposed method with SDWR and a ramp-up weight for the consistency loss used in Li *et al.* [25], as shown in Table 9. As shown in this table, the training model using the ramp-up weight gradually collapses and shows low performance. Furthermore, this approach highly relies on the selected hyperparameter to maintain the consistency of the underconfident samples. In contrast, the usage of bridging loss with SDWR does not need to find a separate hyperparameter and achieves a higher performance than the ramp-up weight-based method.

Ablation for the dynamic weight for $\mathcal{L}_{scc}$. We design the dynamic weight λ_{scc} using $\mathcal{L}_{wcc}$ to minimize the negative effect of $\mathcal{L}_{scc}$. In the previous attempts to design the weight, we

Network	Method	R → C	R → P	R → A	P → R	P → C	P → A	A → P	A → C	A → R	C → R	C → A	C → P	Mean
AlexNet	MME [31]	51.2	73.0	50.3	61.6	47.2	40.7	63.9	43.8	61.4	59.9	44.7	64.7	55.2
	APE [28]	51.9	74.6	51.2	61.6	47.9	42.1	65.5	44.5	60.9	57.1	44.3	64.8	55.6
	CDAC [20]	**54.9**	**75.8**	51.8	64.3	**51.3**	43.6	65.1	**47.5**	63.1	63.0	44.9	65.6	57.6
	CLDA [23]	51.5	74.1	54.3	67.0	47.9	**47.0**	65.8	47.4	66.6	**64.1**	**46.8**	**67.5**	58.3
	DARK (ours)	52.4	**75.8**	**56.5**	**67.5**	48.6	46.8	**67.6**	47.2	**66.7**	63.4	46.6	67.4	**58.9**
VGG-16	MME [31]	56.9	82.9	65.7	76.7	53.6	59.2	75.7	54.9	75.3	72.9	61.1	76.3	67.6
	APE [28]	56.0	81.0	65.2	73.7	51.4	59.3	75.0	54.4	73.7	71.4	61.7	75.1	66.5
	ASDA [8]	59.3	83.6	68.0	**78.3**	56.8	61.8	78.6	**55.7**	75.3	74.0	**63.3**	78.9	69.5
	DECOTA [23]	**59.9**	83.9	67.7	77.3	**57.7**	60.7	78.0	54.9	76.0	74.3	63.2	78.4	69.3
	DARK (ours)	57.6	**84.4**	**69.7**	76.8	55.4	**62.0**	**79.5**	54.4	**79.2**	**75.3**	62.5	**79.3**	**69.7**

Table 7: Quantitative results (%) on Office-Home of the three-shot setting using AlexNet [4] and VGG-16 [10]. The best accuracy is indicated in bold.

Case	Hard label	Soft label	Label smoothing	*Refining*	R → P	P → C	C → S	Mean
1	✓	✗	✗	✗	78.0	75.9	69.3	74.4
2	✗	✓	✗	✗	76.3	75.7	69.9	74.0
3	✗	✓*	✓	✗	72.2	71.5	65.9	70.0
4	✗	✓	✓	✗	76.8	77.5	71.2	75.2
5	✓	✗	✗	✓	78.1	78.1	73.1	76.4
6	✗	✓	✗	✓	78.1	79.3	73.9	77.1
7	✗	✓*	✓	✓	76.5	79.2	72.6	76.1
8	✗	✓	✓	✓	**78.6**	**81.0**	**74.8**	**78.1**

Table 8: Ablation study of components of *Distilling*. We report the classification accuracy (%) on DomainNet of the three scenarios for the three-shot setting. The asterisk symbol (*) of case 4 and 6 indicates the soft label without SDWR.

define a linear weight function to use $1-\mathcal{L}_{wcc}$ with an upper limit β of the weight as follows:

$$\lambda_{scc}^{s,t} = \begin{cases} \beta, & \mathcal{L}_{wcc}^{s,t} \geq 1-\beta \\ 1-\mathcal{L}_{wcc}^{s,t}, & 1-\beta < \mathcal{L}_{wcc}^{s,t} \leq 1 \\ 0, & \mathcal{L}_{wcc}^{s,t} > 1. \end{cases} \tag{11}$$

As shown in Table 10, in the P → C scenario, the classification accuracy is the highest when $\beta = 0.5$. However, for the case of R → S, the highest classification accuracy is derived when $\beta = 0.1$. These results reveal that the performance of training model highly depends on β. To solve this problem, we purpose the dynamic weight strategy that can find the optimal weight value of the trained model with the exponential function. As shown in Table 10, the usage of the purposed dynamic weight achieves the highest accuracies for both P → C and R → S scenarios.

Hyperparameter setting. In the proposed method, the label smoothing parameter α in *Distilling* and the temperature scaling factor T in *Refining* are hyperparameters that were set manually. We apply $\alpha = 0.1$, which is generally used value for label smoothing [6], and set T as 2.5 that is provided by [2]. Figures 3 (a) and 3 (b) show that the performance is not significantly varied when α is in the range of $[0.1\ 0.15]$, and when T is in the range of $[2.0\ 2.5]$. Therefore, the proposed method is robust to the hyperparameters setting and shows the pleasing result even when the setting follows the existing method.

Weight	P → C		C → S		Mean
	1shot	3-shot	1shot	3shot	
Ramp-up [5]	59.0	65.8	56.7	62.0	60.9
SDWR	**79.1**	**81.0**	**71.8**	**74.8**	**76.7**

Table 9: Ablation study for SDWR of the bridging loss and alternative components. We report the classification accuracy (%) on DomainNet of the two scenarios for the three-shot setting.

Task	β			Proposed weight
	0.1	0.3	0.5	
P → C	80.4	80.6	**81.0**	**81.0**
R → S	**74.0**	73.9	73.5	**74.0**

Table 10: Ablation study for dynamic weight in *Refining*. This table implies experimental results for various β of Equation.11 on DomainNet of the two scenarios for the three-shot setting.

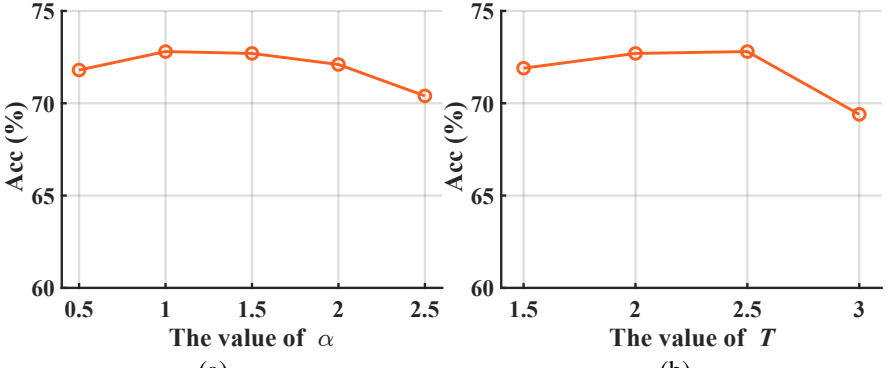


Figure 3: Performance variation depending on hyperparameters for R→S scenario on DomainNet for the one-shot setting. (a) The accuracy depending on α. (b) The accuracy depending on T.

C. Training Scheme

Algorithm 1 explains a training scheme of DARK.

Algorithm 1: Training scheme of DARK

Input : Source dataset D_S, target dataset D_T, unlabeled target dataset D_U, generator $\theta_{\mathcal{G}}$, source-view classifier $\theta_{\mathcal{F}_s}$, target-view classifier $\theta_{\mathcal{F}_t}$, weak augmentation function $\mathcal{A}'(\cdot)$, strong augmentation function $\mathcal{A}''(\cdot)$, learning rate η, iteration N, batch size of each dataset B^S, B^T and B^U

for $n \leftarrow 1$ **to** N **do**

$\mathbf{Batch}\ \mathbf{B}_S \leftarrow \{(\mathcal{A}''(\mathbf{x}_S^i), \mathbf{y}_S^i)\}_{i=1}^{B^S}$ from D_S

$\mathbf{Batch}\ \mathbf{B}_T \leftarrow \{(\mathcal{A}''(\mathbf{x}_T^i), \mathbf{y}_T^i)\}_{i=1}^{B^T}$ from D_T

$\mathbf{Batch}\ \mathbf{B}_{U'} \leftarrow \{\mathcal{A}'(\mathbf{x}_U^i)\}_{i=1}^{B^U}$ from D_U

$\mathbf{Batch}\ \mathbf{B}_{U''} \leftarrow \{\mathcal{A}''(\mathbf{x}_U^i)\}_{i=1}^{B^U}$ from D_U

// `Source-specific knowledge scheme`

$\mathbf{w}_s' = \text{topk}_{[\text{k}=1]}\left(\mathbf{p}(\mathbf{B}_{U'}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_s})\right) - \text{topk}_{[\text{k}=2]}\left(\mathbf{p}(\mathbf{B}_{U'}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_s})\right)$

$\mathbf{w}_s'' = \text{topk}_{[\text{k}=1]}\left(\mathbf{p}(\mathbf{B}_{U''}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_s})\right) - \text{topk}_{[\text{k}=2]}\left(\mathbf{p}(\mathbf{B}_{U''}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_s})\right)$

$\theta_{\mathcal{G}} \leftarrow \theta_{\mathcal{G}} - \eta\left(\nabla\mathcal{L}_{sup}^s(\mathbf{B}_S) + \nabla\mathbf{w}_s' \cdot \mathcal{L}_{dis}^{s\to t}(\mathbf{B}_U', \mathbf{B}_U'') + \nabla\mathcal{L}_{ref}^s(\mathbf{B}_U', \mathbf{B}_U'', \mathbf{w}_s', \mathbf{w}_s'')\right)$

$\theta_{\mathcal{F}_s} \leftarrow \theta_{\mathcal{F}_s} - \eta\left(\nabla\mathcal{L}_{sup}^s(\mathbf{B}_S) + \nabla\mathcal{L}_{ref}^s(\mathbf{B}_U', \mathbf{B}_U'', \mathbf{w}_s', \mathbf{w}_s'')\right)$

$\theta_{\mathcal{F}_t} \leftarrow \theta_{\mathcal{F}_t} - \eta\left(\nabla\mathbf{w}_s' \cdot \mathcal{L}_{dis}^{s\to t}(\mathbf{B}_U', \mathbf{B}_U'')\right)$

// `Target-specific knowledge scheme`

$\mathbf{w}_t' = \text{topk}_{[\text{k}=1]}\left(\mathbf{p}(\mathbf{B}_{U'}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_t})\right) - \text{topk}_{[\text{k}=2]}\left(\mathbf{p}(\mathbf{B}_{U'}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_t})\right)$

$\mathbf{w}_t'' = \text{topk}_{[\text{k}=1]}\left(\mathbf{p}(\mathbf{B}_{U''}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_t})\right) - \text{topk}_{[\text{k}=2]}\left(\mathbf{p}(\mathbf{B}_{U''}; \theta_{\mathcal{G}}, \theta_{\mathcal{F}_t})\right)$

$\theta_{\mathcal{G}} \leftarrow \theta_{\mathcal{G}} - \eta\left(\nabla\mathcal{L}_{sup}^t(\mathbf{B}_T) + \nabla\mathbf{w}_s' \cdot \mathcal{L}_{dis}^{t\to s}(\mathbf{B}_U', \mathbf{B}_U'') + \nabla\mathcal{L}_{ref}^t(\mathbf{B}_U', \mathbf{B}_U'', \mathbf{w}_t', \mathbf{w}_t'')\right)$

$\theta_{\mathcal{F}_t} \leftarrow \theta_{\mathcal{F}_t} - \eta\left(\nabla\mathcal{L}_{sup}^t(\mathbf{B}_T) + \nabla\mathcal{L}_{ref}^t(\mathbf{B}_U', \mathbf{B}_U'', \mathbf{w}_t', \mathbf{w}_t'')\right)$

$\theta_{\mathcal{F}_s} \leftarrow \theta_{\mathcal{F}_s} - \eta\left(\nabla\mathbf{w}_t' \cdot \mathcal{L}_{dis}^{t\to s}(\mathbf{B}_U', \mathbf{B}_U'')\right)$

end

Output: Domain-invariant model $\mathcal{G}$, $\mathcal{F}_s$, and $\mathcal{F}_t$

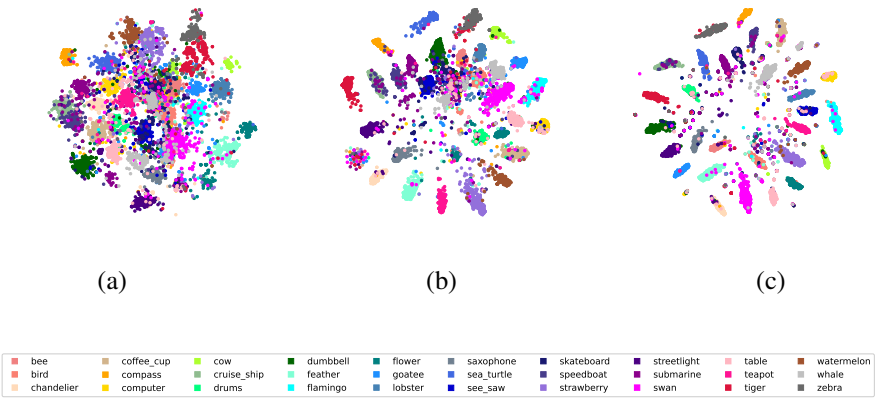


Figure 4: The visualization of the embedding space of DARK using t-SNE [12] of the P → C scenario on DomainNet for the three-shot setting. We show the representations of randomly selected 30 classes on the target domain. (a) The discriminability of the target domain without *Distilling* and *Refining*. (b) The discriminability of the target domain with only *Distilling*. (c) The discriminability of the target domain with *Distilling* and *Refining*. (d) The list of randomly selected classes.

D. Visualization

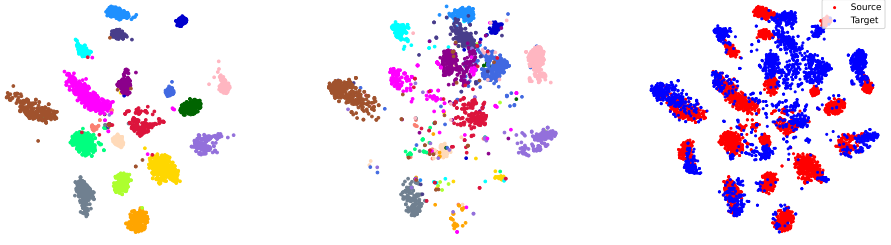
Figure 4 visualizes representations of the target domain using t-SNE [12] on the P → C scenario to analyze the effectiveness of the *Distilling* and *Refining* strategies. This figure shows that our strategies successfully enhance the class discriminability of the target domain. Figure 5 illustrates t-SNE visualization of the source and target domains on the C → S scenario to prove the transferability of the proposed method. In this figure, the first row is a case without *Distilling* and *Refining*, the second row is a case using only *Distilling* and the bridging loss, and the third row is a case using *Distilling* and *Refining*. The figure shows inter- and intra-domain alignment through *Distilling* and the evolutional intra-domain adaptation by *Refining*.

References

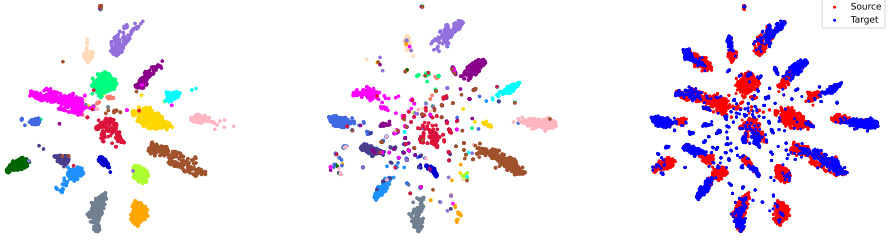
[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. cvpr. 2016. *arXiv preprint arXiv:1512.03385*, 2016.

[2] 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.

(a) Without *Distilling* and *Refining*



(b) With only *Distilling*



(c) With *Distilling* and *Refining*

(d) The list of selected classes (for left and middle columns)

Figure 5: The visualization of the embedding space of DARK using t-SNE [12]. We show the representations of 20 classes on the source and target domains of DARK (a) without *Distilling* and *Refining*, (b) with *Distilling* and the bridging loss, and (c) with *Distilling* and *Refining* for the C $\rightarrow$ S scenario on DomainNet for the three-shot setting. The left and middle columns of this figure are the representations of the source and target domains, respectively. (d) shows the list of the randomly selected classes for the left and middle columns. The right column shows the effectiveness of DARK, where the red and blue symbols denote the source and target distributions, respectively.

[3] Taekyung Kim and Changick Kim. Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation. In *European conference on computer vision*, pages 591–607. Springer, 2020.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[5] Jichang Li, Guanbin Li, Yemin Shi, and Yizhou Yu. Cross-domain adaptive clustering for semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2505–2514, 2021.

[6] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019.

[7] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019.

[8] Can Qin, Lichen Wang, Qianqian Ma, Yu Yin, Huan Wang, and Yun Fu. Semi-supervised domain adaptive structure learning. *arXiv preprint arXiv:2112.06161*, 2021.

[9] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8050–8058, 2019.

[10] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[11] Ankit Singh. Clda: Contrastive learning for semi-supervised domain adaptation. *Advances in Neural Information Processing Systems*, 34:5089–5101, 2021.

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

[13] Luyu Yang, Yan Wang, Mingfei Gao, Abhinav Shrivastava, Kilian Q Weinberger, Wei-Lun Chao, and Ser-Nam Lim. Deep co-training with task decomposition for semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8906–8916, 2021.