Appendix for Enhancing Semi-Supervised Domain Adaptation via Effective Target Labeling

Anonymous submission

Datasets and Implementation Details

Datasets. We conduct experiments on three benchmarks: Office-31, Office-Home, and DomainNet. **Office-31** (Saenko et al. 2010) is a widely used benchmark for the DA task, which contains three domains: Amazon (**A**), Webcam (**W**) and DSLR (**D**) with 31 categories and a total of 4,110 images. **Office-Home** (Venkateswara et al. 2017) is a more challenging medium-sized benchmark for the DA task, which consists of four distinct domains: Artistic images (**A**), Clip Art (**C**), Product images (**P**) and Real-World (**R**). It contains 65 classes in each domain and 15,500 images in total. **DomainNet** (Peng et al. 2019) is a large-scale benchmark dataset, which involves 365 categories and 6 different domains: Clipart (**C**), Infograph (**I**), Painting (**P**), Quickdraw (**Q**), Real (**R**) and Sketch (**S**). Following previous works (Saito et al. 2019; Kim and Kim 2020; Jiang et al. 2020), we select 126 categories and 4 domains, i.e., Real, Clipart, Painting, Sketch for evaluation.

Implementation Details. We use the ResNet-34 (He et al. 2016) on ImageNet as our backbone, where the last fully-connected (FC) layer is replaced by two task-specific FC layers like (Saito et al. 2019). The SGD optimizer with momentum 0.9 and weight decay 5e-4 is employed for training. We use the same learning rate scheduler and batch size setting as those in MME (Saito et al. 2019) and the initial learning is set as 1e-2 for all experiments. We train the source model for 20 epochs and target model for 100 epochs. Since the number of the neighbors of ADN and AEN depends on the scale of the dataset, we choose $M_1 = \lfloor n_t/(3 \times K) \rfloor$ and $M_2 = \lfloor M_1/5 \rfloor$ to adaptively construct the directed graph across all datasets. The trade-off hyperparameter α in Eq. (7) is fixed to 0.1. For baseline FixMME, the threshold τ in Eq. (7) is set to be 0.85 for all datasets except 0.8 for *DomainNet*. Finally, we conduct three independent experiments and report the average results.

Derivations of P^1 and P^2

Let P^1 and P^2 denote the probabilistic matrices obtained by the Gaussian mixture model clustering and K-means clustering, respectively. We implement the clustering algorithms in the embedding space $\mathbb{R}^m$, i.e., each example is represented by $\mathbf{f}_i = F_s(x_t^i)$.

Gaussian mixture model clustering. We use the Expectation-Maximization (EM) algorithm to estimate the parameters $(\alpha_k, \mu_k, \Sigma_k)$ for $k = 1, ..., K$, where α_k denotes the probability of the k-th Gaussian distribution, and μ_k, Σ_k represent the mean vector and covariance matrix of the k-th Gaussian distribution, respectively. Then, we have

$$p(x_t^i|\mu_k, \Sigma_k) = \frac{\exp(-\frac{1}{2}(\mathbf{f}_i - \mu_k)^\top \Sigma_k^{-1}(\mathbf{f}_i - \mu_k))}{(2\pi)^{\frac{m}{2}}|\Sigma_k|^{\frac{1}{2}}},$$

and

$$P_{ik}^1 = \frac{\alpha_k p(x_t^i|\mu_k, \Sigma_k)}{\sum_{k=1}^K \alpha_k p(x_t^i|\mu_k, \Sigma_k)}.$$

K-means clustering. Let $\{c_k^0\}_{k=1}^K$ denote the initial class centers, where

$$c_k^0 = \frac{\sum_{i=1}^{n_t} \mathbb{1}(\hat{y}_t^i = k) F_s(x_t^i)}{\sum_{i=1}^{n_t} \mathbb{1}(\hat{y}_t^i = k)}.$$

We update the class centers according to the cosine similarity, until they converge. We denote by $\{c_k\}_{k=1}^K$ the final class centers. Then, we have

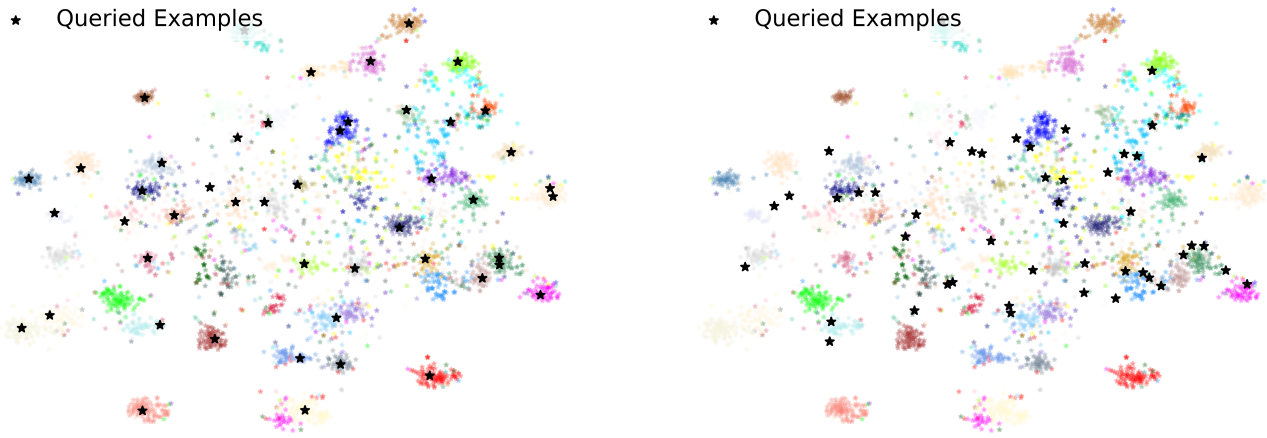
$$\tilde{P}_{ik}^2 = \frac{(F_s(x_t^i))^\top c_k}{\|F_s(x_t^i)\|\|c_k\|},$$

$$P_{ik}^2 = \frac{\exp(\tilde{P}_{ik}^2/T)}{\sum_{k=1}^K \exp(\tilde{P}_{ik}^2/T)},$$

where T is the sharpness hyper-parameter, and we fix it to 0.05 for all experiments.

Additional Experiments

Representativeness and diversity of NDNS. We visualize the *representativeness* and *diversity* of NDNS in Fig. (1). We can see that the queried examples through NDNS are *intra-class representative* and *inter-class diverse*. The instances obtained by using pure margin criteria are close the region where the decision boundaries of the target learning model are unclear. However, the latter is lack of diversity and may contain outliers. Thus, it is more reasonable to combine NDNS with model uncertainty to select more effective target instances for labeling.



(a) NDNS without considering margin

(b) Margin Only

Figure 1: t-SNE visualization of queried examples exhibited as the black stars based on different methods in the feature space. The classes are displayed in different colors and each colored point represents a target example. (a): The samples are selected using NDNS without considering model uncertainty. (b): The samples are selected by the criteria of small margins.

Table 1: Classification accuracy (%) on the ADA task with 5% labeling budget on *Office-Home* benchmark using ResNet-50 as the backbone, where we only use the labeled source and target data to train the model for fair comparisons with TQS and SDM.

ADA Method	A → C	A → P	A → R	C → A	C → P	C → R	P → A	P → C	P → R	R → A	R → C	R → P	Avg.
ResNet-50	42.1	66.3	73.3	50.7	59.0	62.6	51.9	37.9	71.2	65.2	42.6	76.6	58.3
RANDOM	52.5	74.3	77.4	56.3	69.7	68.9	57.7	50.9	75.8	70.0	54.6	81.3	65.8
AADA (Su et al. 2020)	56.6	78.1	79.0	58.5	73.7	71.0	60.1	53.1	77.0	70.6	57.0	84.5	68.3
BADGE (Ash et al. 2020)	59.2	81.0	81.6	60.8	74.9	73.3	63.7	54.2	79.2	73.6	59.7	85.7	70.6
TQS (Fu et al. 2021)	58.6	81.1	81.5	61.1	76.1	73.3	61.2	54.7	79.7	73.4	58.9	86.1	70.5
SDM (Xie et al. 2022b)	61.2	82.2	82.7	**66.1**	77.9	76.1	66.1	58.4	81.0	**76.0**	62.5	87.0	73.1
NDNS (ours)	**66.7**	**84.3**	**82.9**	66.0	**82.9**	**78.6**	**67.4**	**65.5**	**82.4**	74.7	**68.6**	**89.0**	**75.8**
EADA (Xie et al. 2022a)	63.6	84.4	83.5	70.7	83.7	80.5	73.0	63.5	85.2	78.4	65.4	88.6	76.7
DiaNA (Huang et al. 2023)	64.5	86.0	84.9	72.3	84.6	82.5	73.3	63.7	85.6	78.5	67.2	89.5	77.7

Comparisons with ADA methods. In the Experiments section of the main paper, we have seen that our proposed EFTL can easily cooperate with existing SSDA models and significantly improve their performance. For fair comparisons with ADA methods, we only use the proposed NDNS module to query target samples, where we train the model only using the labeled source and labeled target data. The results are reported in Table 1. We can see that NDNS can surpass previous ADA with significant margins on the most transfer tasks. In particular, NDNS excels at hard transfer tasks, e.g., A → C, P → C, and R → C, surpassing SDM (Xie et al. 2022b) with more significant margins. In addition, both EADA (Xie et al. 2022a) and DiaNA (Huang et al. 2023) used unsupervised training losses, e.g., entropy and consistency loss in DiaNA. Hence, we do not directly compare NDNS with EADA and DiaNA. Notably, FixMME + EFTL with 3-shot annotation budget can achieve a average result 78.7% on *Office-Home* benchmark, which surpasses DiaNA with a 1% margin.

References

Ash, J. T.; Zhang, C.; Krishnamurthy, A.; Langford, J.; and Agarwal, A. 2020. Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds. In *ICLR*.

Fu, B.; Cao, Z.; Wang, J.; and Long, M. 2021. Transferable query selection for active domain adaptation. In *CVPR*, 7272–7281.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Huang, D.; Li, J.; Chen, W.; Huang, J.; Chai, Z.; and Li, G. 2023. Divide and Adapt: Active Domain Adaptation via Customized Learning. In *CVPR*, 7651–7660.

Jiang, P.; Wu, A.; Han, Y.; Shao, Y.; Qi, M.; and Li, B. 2020. Bidirectional Adversarial Training for Semi-Supervised Domain Adaptation. In *IJCAI*, 934–940.

Kim, T.; and Kim, C. 2020. Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation. In *ECCV*, 591–607. Springer.

Peng, X.; Bai, Q.; Xia, X.; Huang, Z.; Saenko, K.; and Wang, B. 2019. Moment matching for multi-source domain adaptation. In *ICCV*, 1406–1415.

Saenko, K.; Kulis, B.; Fritz, M.; and Darrell, T. 2010. Adapting visual category models to new domains. In *ECCV*, 213–226. Springer.

Saito, K.; Kim, D.; Sclaroff, S.; Darrell, T.; and Saenko, K. 2019. Semi-supervised domain adaptation via minimax entropy. In *ICCV*, 8050–8058.

Su, J.-C.; Tsai, Y.-H.; Sohn, K.; Liu, B.; Maji, S.; and Chandraker, M. 2020. Active adversarial domain adaptation. In *WACV*, 739–748.

Venkateswara, H.; Eusebio, J.; Chakraborty, S.; and Panchanathan, S. 2017. Deep hashing network for unsupervised domain adaptation. In *CVPR*, 5018–5027.

Xie, B.; Yuan, L.; Li, S.; Liu, C. H.; Cheng, X.; and Wang, G. 2022a. Active learning for domain adaptation: An energy-based approach. In *AAAI*, volume 36, 8708–8716.

Xie, M.; Li, Y.; Wang, Y.; Luo, Z.; Gan, Z.; Sun, Z.; Chi, M.; Wang, C.; and Wang, P. 2022b. Learning Distinctive Margin toward Active Domain Adaptation. In *CVPR*, 7993–8002.