MHPL: Minimum Happy Points Learning for Active Source Free Domain Adaptation: Appendix

Fan Wang[1], Zhongyi Han[1], Zhiyan Zhang[1], Rundong He[1], Yilong Yin[1*]
[1] School of Software, Shandong University, China
{fanwang,rundong_he}@mail.sdu.edu.cn, {hanzhongyicn,zyzhangcs}@gmail.com, ylyin@sdu.edu.cn

A. Limitations

While we can verify that exploring and exploiting minimum happy points are critical for ASFDA through a significant improvement, the existence of the MH points is challenging to prove due to the lacking of theoretical guarantees and interpretability of deep networks. Further, ASFDA relies on the well-trained source model that may be impaired by some causes, *e.g.,* model transfer, model sharing, and source model training process. In these uncontrollable and unforeseen circumstances, the robustness of ASFDA methods would face serious challenges.

B. Implementation details

Model Details. We implement our experiments on the PyTorch platform. For a fair comparison, we adopt the backbone of ResNet-50 [4] for Office-Home and Office31, ResNet101 for VisDA-2017. Besides, we also add experiments of ResNet-50 on VisDA-2017, VGG16 [12] and Alexnet [7] on Office-Home and Office-31. We utilize the same network architecture as SHOT and conduct label smoothing as SHOT in the process of training the source model. For all datasets without a train-validation split, we view the all source data as a test set and train the optimal source model using the test set as validation. The maximization number of epochs for Office-31, Office-Home, and VisDA is set as 100, 50, and 10. Meanwhile, when we train the target model, we train 20 epochs for Office-31 and Office-Home, and 10 epochs for VisDA. We run three times and report the average results. In the process of training, we adopt minibatch SGD with momentum 0.9 and weight decay 1e-3. The learning rate is set to 1e-2 for Office-31 and Office-Home, and 1e-3 for VisDA-2017. We adopt the existing schedule: $\eta = \eta_0(1 + 10p)^{-0.75}$ [10], where η_0 is the initial learning rate and p is the training progress changing from 0 to 1. Besides, we set the batch size to 64 for all the tasks. For the number of neighbors q, we set nine for Office-31, twenty for Office-Home, and five for VisDA-2017. Besides, we set α = 3, β = 0.3, and o = 5 for all datasets. Experiments are conducted on a TITAN Xp.

Labeling budget. In active source free domain adaptation, a few samples are utilized to aid the adaptation process. Following the previous active domain adaptation works [3, 15], we mainly shows the results of 5% actively labeled target data. We also report the results of various selection ratios, *e.g.,* 1%,5%,..., 10%, for analysis.

Baseline implementation. The active domain adaptation methods cannot be reproduced without the source data in ASFDA, so we report the best results shown in papers corresponding to our active source free domain adaptation, such as AADA [13], EADA [15], and so on. Similarly, because the ELPT [8] does not open the code, we report its original results. We now elaborate on our implementation of other baseline algorithms:

(1) **Base:** we acquire the pseudo-labels of the target samples based on a deep clustering [9], and train the model with entropy loss, KL divergence and the standard cross-entropy loss. Here the pseudo-labels are not credible, so we apply the weight β = 0.3 and α = 0.3 to all samples.

(2) **Random**: select samples randomly.

(3) **Entropy** [14]: select samples for which the model has highest predictive entropy.

(4) **BVSB** (Best-Versus-Second-Best) [6]: select samples for which the the smallest difference between the top-2 predicted probabilities.

(5) **LC** (Least Confidence) [5]: select samples for which the smallest of the maximum probability of model output.

(6)**CoreSet** [11]: CoreSet formulates active sampling as a set-cover problem. We implement the CoreSet using the released code: https://github.com/JordanAsh/badge.

(7) **CTC** (closet to its center): select samples which are the closet to their centers after the Kmeans clustering.

(8) **BADGE** (batch active learning by diverse gradient embeddings) [2]: BADGE ensures diverse batches by running Kmeans++ [1] on 'gradient embeddings', which incorporates model uncertainty and diversity. We implement BADGE using the released code: https://github.com/JordanAsh/badge.

*denotes corresponding author.

Table 1. Accuracy (%) on Office-Home of different networks with 5% labeled target samples.

Model	Method	Ar→Cl	Ar→Pr	Ar→Re	Cl→Ar	Cl→Pr	Cl→Re	Pr→Ar	Pr→Cl	Pr→Re	Re→Ar	Re→Cl	Re→Pr	Avg
Alexnet	Source-only	26.1	37.4	49.5	24.5	39.6	40.0	25.3	24.0	49.8	40.0	31.3	59.8	37.3
	Base	30.7	50.3	56.2	34.5	51.4	53.4	33.0	26.8	58.9	44.1	36.2	64.9	45.0
	Random	37.8	58.0	61.0	39.7	58.1	59.4	36.2	34.2	62.8	48.7	45.0	69.8	50.9
	CTC	37.3	55.3	58.5	38.2	56.6	56.9	35.4	33.3	60.4	45.5	43.0	65.6	48.8
	CoreSet [11]	36.1	55.1	59.9	38.9	57.9	58.8	38.0	31.3	62.4	47.7	42.0	69.1	49.8
	BADGE [2]	37.3	54.7	59.7	39.3	58.3	58.1	38.8	33.3	63.6	49.1	43.1	69.1	50.4
	Entropy [14]	38.3	56.1	62.7	40.5	59.0	60.7	37.3	35.9	64.6	48.8	45.3	71.0	51.7
	BVSB [6]	39.2	57.7	62.8	39.7	60.1	61.5	36.7	36.7	64.9	48.9	45.8	71.6	52.1
	LC [5]	39.0	56.8	63.4	39.8	59.7	60.6	37.4	36.7	64.9	48.5	45.8	71.4	52.0
	MHPL	**45.5**	**64.4**	**64.9**	**42.8**	**67.3**	**61.7**	**40.8**	**44.4**	**66.4**	**50.5**	**50.9**	**75.7**	**56.3**
VGG	Source-only	35.2	59.8	69.4	48.0	60.5	62.6	46.7	30.8	70.5	58.8	35.4	74.3	54.3
	Base	44.3	73.1	74.0	59.3	72.5	71.5	55.9	42.0	76.6	63.5	46.6	80.4	63.3
	Random	52.6	77.0	76.9	61.8	76.8	74.9	58.7	49.4	78.8	66.3	53.7	83.3	67.5
	CTC	50.8	74.3	76.0	60.3	75.2	73.5	58.2	47.9	77.3	64.9	52.1	81.0	66.0
	CoreSet [11]	49.7	77.2	76.6	63.3	77.2	74.9	60.1	46.4	79.0	66.5	50.7	83.4	67.1
	BADGE [2]	51.0	77.4	77.0	63.2	77.0	75.3	60.0	47.2	79.3	66.9	51.4	83.4	67.4
	Entropy [14]	52.6	78.1	78.0	63.1	77.9	72.2	60.4	49.0	80.6	68.5	54.1	84.1	68.6
	BVSB [6]	54.7	77.1	78.6	63.5	78.6	77.1	59.7	50.7	80.2	68.3	54.6	85.3	69.0
	LC [5]	53.0	78.7	78.4	63.0	78.0	76.8	60.4	49.2	80.5	68.5	54.3	85.1	68.8
	MHPL	**60.4**	**82.4**	**80.5**	**66.7**	**82.3**	**78.4**	**63.5**	**56.9**	**81.9**	**70.8**	**59.3**	**87.2**	**72.5**

Table 2. Accuracy (%) on Office-31 of different networks with 5% labeled target samples.

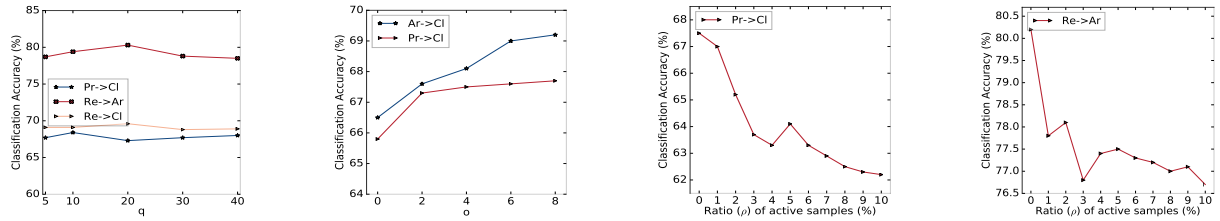
Model	Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg
Alexnet	Source-only	53.6	47.9	37.9	94.2	36.4	97.8	61.3
	Base	69.2	63.1	69.6	95.7	48.6	98.6	70.8
	Random	73.1	68.2	58.2	95.9	55.4	99.6	75.0
	CTC	71.5	65.5	56.3	95.6	52.7	99.0	73.4
	CoreSet [11]	70.3	69.9	59.2	96.0	54.2	98.4	74.7
	BADGE [2]	71.3	69.9	57.2	96.4	53.9	99.4	74.7
	Entropy [14]	74.2	71.2	57.1	98.9	54.8	99.6	76.0
	BVSB [6]	73.1	73.8	61.7	98.7	58.6	99.6	77.6
	LC [5]	74.1	74.1	58.6	99.1	55.4	99.6	76.8
	MHPL	**81.1**	**78.6**	**68.9**	**98.5**	**67.0**	**100.0**	**82.4**
VGG	Source-only	75.2	72.7	63.8	95.4	63.7	99.8	78.5
	Base	88.0	87.2	72.9	97.2	71.6	99.8	86.1
	Random	87.4	89.1	74.4	97.5	73.8	99.8	87.0
	CTC	88.0	86.9	73.4	97.1	71.9	99.8	86.2
	CoreSet [11]	89.0	88.7	75.1	97.5	74.1	99.8	87.4
	BADGE [2]	89.2	88.1	75.5	97.5	73.3	99.8	87.2
	Entropy [14]	87.8	89.1	77.7	98.2	74.7	100.0	87.9
	BVSB [6]	91.6	88.4	76.3	98.5	75.2	100.0	88.3
	LC [5]	90.8	88.9	77.2	98.5	75.3	100.0	88.5
	MHPL	**93.0**	**93.0**	**79.9**	**98.9**	**80.3**	**100.0**	**90.9**

Methods (3), (4), and (5) are based on model uncertainty, (6) and (7) are diversity-based, and (8) is a hybrid approach that combines uncertainty and diversity.

Regarding implementation details, MHPL selects active samples and learns selected samples with a proposed neighbor focal loss compared with the base. In contrast to other active strategies, MHPL explores more informative MH points and exploits these MH points with a proposed neighbor focal loss instead of the standard cross-entropy loss.

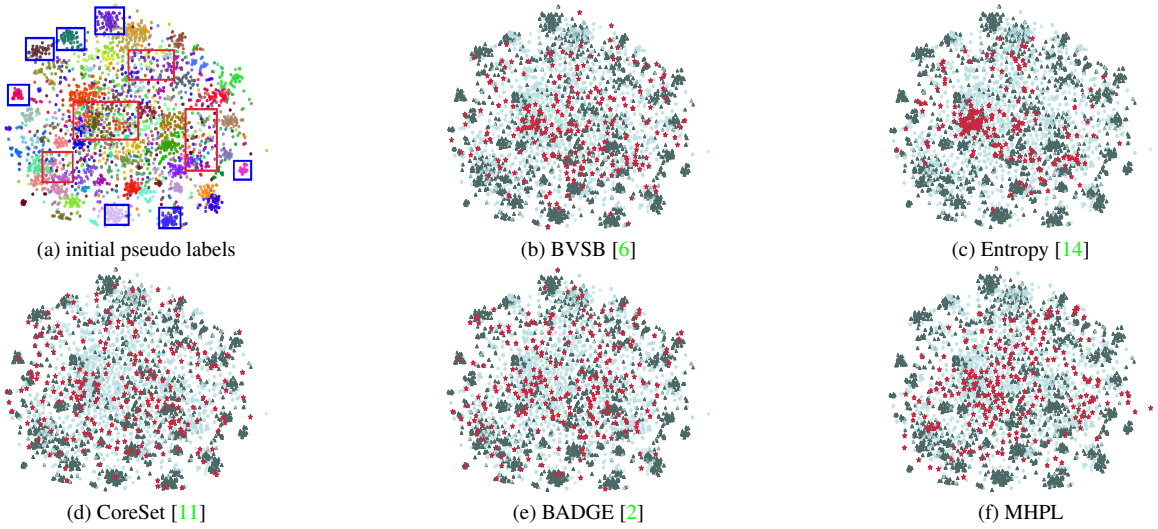
C. Additional experimental results

Full experimental results under different networks. To further prove the effectiveness of MHPL, we conduct experiments on Office-31 and Office-Home with different backbones: VGG and Alexnet. We find two essential observations from the results in Table 1 and 2. (1) Even a small set of labeled target data may bring larger performance gains, which proves the practicality and effectiveness of ASFDA. On average, when only 5% of the Office-home target data is used for annotation, MHPL is 11.3% and 9.2% higher than the base with Alexnet and VGG, respectively. Similarly, the accuracy of MHPL is 11.6% and 4.8% higher than the base on Office-31 with Alexnet and VGG, respectively. (2) Our method MHPL could fully explore and exploit MH points which are crucial for ASFDA. As shown in Table 1 and 2, the performance gains brought from MHPL are significantly higher than other active strategies in all tasks with VGG and Alexnet, especially in several challeng-



(a) Neighbor number of NAU (b) Neighbor number of NDR (c) Pr→Cl (d) Re→Ar

Figure 1. Ablation studies on hyperparameters q, o and one-shot querying.



(a) initial pseudo labels (b) BVSB [6] (c) Entropy [14]

(d) CoreSet [11] (e) BADGE [2] (f) MHPL

Figure 2. Feature visualization for the source model with 5% actively labeled target data on Ar→Cl task. Different colors in (a) represent different classes of pseudo-labels by clustering. Blue blocks include easily-adaptive source-similar samples with label-clean neighbors that can be learned well by SFDA methods. Red blocks include the hard-adaptive source-dissimilar samples with label-chaotic neighbors. In (b), (c), (d), (e), and (f), the dark green indicates that the pseudo-label is consistent with the true label, and light blue indicates the opposite. The red stars indicate the selected samples based on different criteria, respectively.

ing tasks, *e.g.,* Ar→Cl, Pr→Cl, Re→Cl, D→A, and W→A.

Ablation study on number of neighbors q. To study the sensitivity to a crucial hyperparameter q, we run ablation experiments on three tasks, Pr→Cl, Re→Ar, and Re→Cl, in ResNet-50. As shown in Fig. 1 (a), utilizing 5% labeled target data, results with $q = 5/10/20/30/40$ show that our method is not sensitive to the choice of a reasonable q on three tasks.

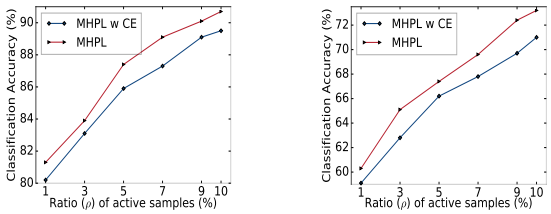
Ablation study on number of neighbors o in NDR. To study the sensitivity to a hyperparameter o in neighbor diversity relaxation, we run ablation experiments on Ar→Cl and Pr→Cl in ResNet-50. As shown in Fig. 1 (b), when o = 0, the NDR is not being executed, and the model is not performing well. The model's accuracy is significantly improved with the increase of o value. When the value of o is larger, the accuracy gradually tends to be stable. We set o = 5 on all datasets.

Ablation study on one-shot querying. To further verify the effect of one-shot querying, we conduct the experiments on Pr→Cl and Re→Ar to compare the effect with the samples selected at different epochs of model training. The results in Fig. 1 (c) and (d) verify that the samples chosen by the source model are source-dissimilar and effective.

Feature visualization. As shown in Fig. 2, compared with BVSB [6], Coreset [11], and BADGE [2], the samples selected by our MHPL mostly fall into source-dissimilar and label-chaotic regions. Even the samples chosen by Entropy [14] are also mostly label-chaotic, but they are not diverse. Therefore, the samples selected by our MHPL meet all three conditions of minimum happy points: neighbor-chaotic, individual-different, and source-dissimilar.

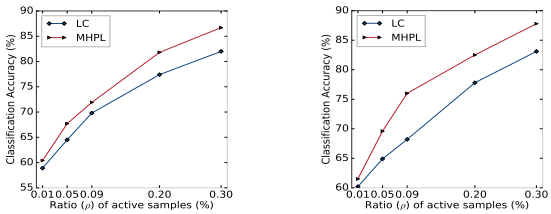
Ablation on focal loss. As shown in Fig. 3, with different active sample ratios in Cl→Pr and Pr→Cl, the results from focal loss are always higher than those from CE loss.

More ratios. Since LC [5] is the best approach in the active strategy, we compare LC with our MHPL in different ratios of active samples on the Pr→Cl and Re→Cl tasks. For the two tasks shown in Fig. 4, our MHPL is always far



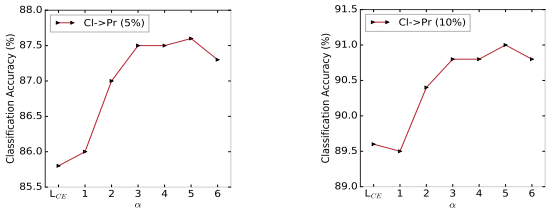
(a) Cl→Pr (b) Pr→Cl

Figure 3. Ablation study on focal loss. 'MHPL w CE' represents replacing the focal loss with normal CE loss in MHPL.



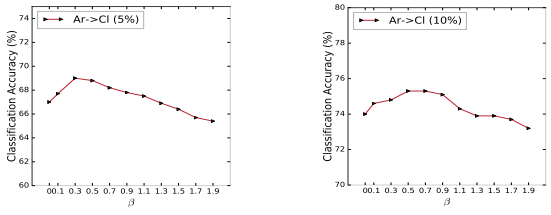
(a) Pr→Cl (b) Re→Cl

Figure 4. Comparison on more selective ratios.



(a) Cl→Pr (b) Cl→Pr

Figure 5. Ablation study on α.



(a) Ar→Cl (b) Ar→Cl

Figure 6. Ablation study on β.

superior to LC in different sample selection ratios.

Ablation study on hyperparameters α. We utilize 5% and 10% active labeled target samples in the Cl→Pr task to analyze the hyperparameter of α. As shown in Fig. 5 (a) and (b), the accuracy with αPu is significantly higher than that of cross-entropy loss. We set $\alpha = 3$ on all datasets.

Ablation study on hyperparameters β. We utilize 5% and 10% active labeled target samples in the Ar→Cl task to analyze the hyperparameter of β. When $\beta = 0$, the performance is worse as the D_t^U is not utilized. With the increase of β, the effect of the model first increases and then decreases, indicating that the model is supposed not to focus more on D_t^U. We set $\beta = 0.3$ for all datasets.

References

[1] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006. 1

[2] Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019. 1, 2, 3

[3] Bo Fu, Zhangjie Cao, Jianmin Wang, and Mingsheng Long. Transferable query selection for active domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 7272–7281. Computer Vision Foundation / IEEE, 2021. 1

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[5] Tao He, Xiaoming Jin, Guiguang Ding, Lan Yi, and Chenggang Yan. Towards better uncertainty sampling: Active learning with multiple views for deep convolutional neural network. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1360–1365. IEEE, 2019. 1, 2, 3

[6] Ajay J. Joshi. Multi-class active learning for image classification. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 2372–2379. IEEE Computer Society, 2009. 1, 2, 3

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 1

[8] Xinyao Li, Zhekai Du, Jingjing Li, Lei Zhu, and Ke Lu. Source-free active domain adaptation via energy-based locality preserving transfer. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5802–5810, 2022. 1

[9] 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. 1

[10] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *arXiv preprint arXiv:1705.10667*, 2017. 1

[11] Ozan Sener. Active learning for convolutional neural networks: A core-set approach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 1, 2, 3

[12] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[13] Jong-Chyi Su, Yi-Hsuan Tsai, Kihyuk Sohn, Buyu Liu, Subhransu Maji, and Manmohan Chandraker. Active adversarial domain adaptation. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020*, pages 728–737. IEEE, 2020. 1

[14] Dan Wang and Yi Shang. A new active labeling method for deep learning. In *2014 International Joint Conference on Neural Networks, IJCNN 2014, Beijing, China, July 6-11, 2014*, pages 112–119. IEEE, 2014. 1, 2, 3

[15] Binhui Xie, Longhui Yuan, Shuang Li, Chi Harold Liu, Xinjing Cheng, and Guoren Wang. Active learning for domain adaptation: An energy-based approach. In *AAAI*, volume 36, pages 8708–8716, 2022. 1