Supplementary: Unleashing the Potential of Adaptation Models via Go-getting Domain Labels

Xin Jin[1]*, Tianyu He[2], Xu Shen[2], Songhua Wu[3], Tongliang Liu[3], Jingwen Ye[4], Xinchao Wang[4], Jianqiang Huang[2], Zhibo Chen[5], and Xian-Sheng Hua[2]

[1] Eastern Institute for Advanced Study
[2] Alibaba Group
[3] The University of Sydney
[4] National University of Singapore
[5] University of Science and Technology of China

1 Impact Discussion

In this paper, we present a simple yet effective training strategy, with a novel concept of go-getting domain labels (Go-labels), for domain adaptation, which significantly improves the performance of many adversarial based domain adaptation variants while adding almost no extra computational cost. This method can benefit any vision task where unsupervised domain adaptation (UDA) is required, such as cross-domain scene parsing/segmentation [14, 17, 28], cross-domain person re-identification [13, 1, 29, 9], cross-domain object retrieval [12, 11, 7] *etc.* It can be used to boost the domain adaptation capability of these vision tasks, and therefore these developed vision tasks will further promote the evolution of many practical industrial applications, *e.g.*, Autonomous Vehicle, Large-scale Surveillance System, *etc.*

Our technique can also be useful when a model is required to have a life-long learning ability [3, 20], because training model under the unsupervised domain adaptation setting in a high-efficient way can help obtain an easy-to-learning and easy-to-transfer model.

2 More Details about Toy Experiments

Random Point Classification. In the main manuscript, we observe the behavior of our proposed training strategy of dynamic adversarial domain adaptation method with go-getting domain labels (Go-labels) on toy problem of *2D random point classification*, in which we used *numpy.random* [19] to synthesize the toy source and target samples that share the same label space for validation.

For the network structure, we use the totally same architecture for two schemes of *Baseline* and *Ours*. We adopt a multilayer perceptron (MLP) [6] as the feature extractor F (refer to Eq. (1) of the main manuscript for notation), which MLP is composed of three fully connected layers with BatchNorm1d and ReLU layers

* Corresponding Author, e-mail: jinxin@eias.ac.cn

for stable training. The category classifier C is an one-layer fully connected layer followed by a sigmoid function to output the classification result ('0' – red point or '1' – green point). For the domain discriminator D (refer to Eq. (2) of the main manuscript for notation), it is also composed of three fully connected layers with inserted dropout and ReLU layers for stable training following [15, 2], followed by a sigmoid function to output the domain classification result. A gradient reversal layer (GRL) [4, 5, 15] is used to connect feature extractor F and domain discriminator D to achieve the adversarial function by multiplying the gradient from D with a certain negative constant during the back-propagation to the feature extractor F.

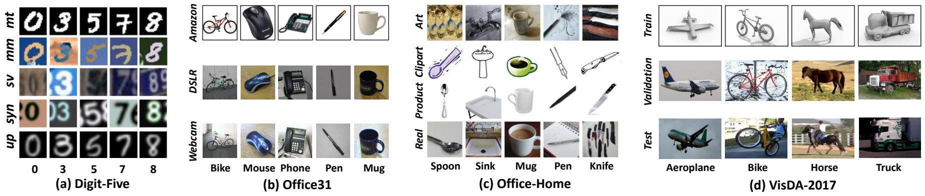
For the basic optimization hyper-parameters, we employ stochastic gradient descent (SGD) as optimizer with an initial learning rate of 0.01 train all the schemes of *Baseline* and *Ours*. Batch size is set as 100 and total training epoch is set as 10.

Inter-twinning Moons. For this toy problem, we conduct experiment fully based on the codebase [6] released by [5], we recommend readers to get more details from their original paper.

3 Code, Datasets, and Training Details

Source Code. We have uploaded the source code that corresponds to our proposed dynamic adversarial domain adaptation method with go-getting domain labels (Go-labels). Please find details and reproduce the main experimental results in the uploaded materials of 'Domain_Adaptation_with_Go-labels.zip'.

Datasets. Here we further provide more details about the used datasets, including Digit-Five, Office31, Office-Home, and VisDA-2017. Fig. 1 shows some samples of some datasets.



(a) Digit-Five (b) Office31 (c) Office-Home (d) VisDA-2017

Fig. 1: Some examples covering different domains of datasets we used.

- (1) *Digit-Five* consists of five different digit recognition datasets: MNIST [10], MNIST-M [4], USPS [8], SVHN [18] and SYN [4]. We follow the same split setting as [21] to utilize such dataset.
- (2) *Office31* [23] is the most widely used dataset for visual domain adaptation, with 4,652 images and 31 categories collected from three distinct domains: Amazon (**A**), Webcam (**W**) and DSLR (**D**). We evaluate all methods on six

[6] https://github.com/GRAAL-Research/domain_adversarial_neural_network

transfer tasks **A** → **W**, **D** → **W**, **W** → **D**, **A** → **D**, **D** → **A**, and **W** → **A**, respectively.

- (3) *Office-Home* [27] is a more difficult dataset (with relative large domain discrepancy) than *Office31*. It consists of 15,500 images of 65 object classes in office and home settings. It has four dissimilar domains: Artistic images (**Ar**), ClipArt (**Cl**), Product images (**Pr**), and Real-World images (**Rw**). Among the four domains, there are a total of 12 DA tasks.
- (4) *VisDA-2017* [22] is a simulation-to-real dataset for UDA with over 280,000 images across 12 categories in the training, validation and testing domains.

Training. In all experiments, SGD with momentum is used as the optimizer and the cosine annealing rule [16] is adopted for learning rate decay. All our experiments are implemented on PyTorch and conducted on a single 12G NVIDIA 1080ti GPU.

For Digit-Five, the CNN backbone is constructed with three convolution layers and two fully connected layers, termed as Cov_3FC_2 following [21]. For each mini-batch, we sample 64 images for training. The model is trained with an initial learning rate of 0.05 for totally 30 epochs.

For Office-31 and Office-Home, following [15, 30, 2], we use ResNet-50 as backbone. The initial learning rate is set to 1e-3. The input image size is 224×224 and the batch size is 36. We train the models for 500 epochs (nearly 16,000 iterations) and evaluate their adaptation performance. We use the default train/test/val split protocol as [2, 15] for both the two datasets.

For VisDA-2017, following [15, 2], we also use ResNet-50 as backbone. The initial learning rate is set to 1e-4. The input image size is 224×224, and the batch size is 36. We follow the train/val/test split protocol of [2] and train the models for 150 epochs.

4 More Experimental Results

4.1 Validation on Toy Problems

2D Random Point Classification. First, we observe the behavior of our method on toy problem of *2D random point classification*, in which we use *numpy.random* [19] to generate the source and target samples that share the same label space. For the source samples, we generate point samples with 2 classes, labeled as '0' (marked as red) and '1' (marked as green), respectively. For each class, it contains 3 data clusters with different scales (*i.e.*, large head cluster has 10,000 samples, middle cluster has 5,000 samples, small tail cluster has 200 samples), this design aims to simulate the data imbalance situation in real-world, *i.e.*, the problem Go-labels focused. For the target samples, we totally generate 10,200 samples for each class. Each class has two clusters, one large head cluster with 10,000 samples and one small tail cluster with 200 samples. We compared the class decision boundary of our method with *Baseline* obtained from the domain discriminator trained with immutable domain labels. To better

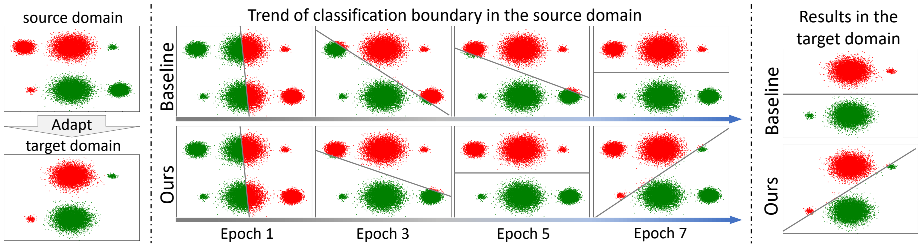


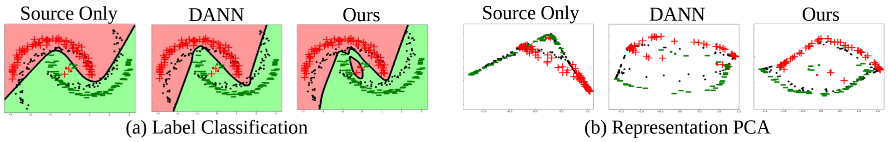
Fig. 2: *2D random point classification.* Red and green points indicate the samples of class '0' and '1', respectively (left). The solid line denotes the class decision boundary, and we use "color change" to indicate its changing trend (middle) during the training process. Adaptation performance on the target set is shown on right. For the fairness of comparison, the hyper-parameters, including learning rates and total training iterations, are same for both baseline and our method (see **Supplementary** for more details).

evaluate adaptation performance of the trained model, we visualize source and target data separately. Other details are provided in **Supplementary**.

As shown in Fig. 2, the *Baseline* scheme is prone to miss the small tail cluster, especially when it is very closed to a large cluster belonged to the different class. In contrast, our method could better leverage both large/head and small/tail data clusters in the different domains to reduce discrepancy. The trend of our classification boundary in the source domain has demonstrated this point. As a result, the adaptation performance on the target set of ours is obviously superior to that of *Baseline*.

Inter-twinning Moons. Furthermore, we observe the behavior of Go-labels on toy problem of *inter-twinning moons* [5, 25]. In particular, we additionally generate some outlier samples near the center of each moon to mimic the imbalanced data distribution. For the source data, a lower moon and an upper moon are generated, and labeled as '0' and '1'. Each of them is accompanied by two extra outliers, totally 152 samples. Target data are generated by re-sampling from the source distribution. Then, we rotate each sample by 35° and remove its label to obtain an unlabeled target set. We compare our method with the model trained with source data only and DANN [5] in the Fig. 3. We observe that both baselines of *Source only* and *DANN* neglect the outlier samples. In contrast, our method not only gets a satisfactory classification boundary between two classes in the source domain, but also covers these minority tail data well and classifies them to the correct class. Besides, after performing PCA, we can see that our method also achieves a better feature alignment in comparison with other two baselines, where target samples that denoted as black points are homogeneously spread out among source points.

Feature Distributions Visualization. Here, we further visualize the learned feature distributions by t-SNE [24] for $W \rightarrow A$ setting of Office31 in Fig. 4. We observe the scheme of *Source Only* that without considering domain adaptation



(a) Label Classification (b) Representation PCA

Fig. 3: The second toy game of *inter-twinning moons*. Red "+", green "-", and black "·" markers indicate the source positive samples (label 1), source negative samples (label 0), and target samples, respectively. (a) The solid black line is the class decision boundary. (b) We also show the feature alignment situations of different schemes via a principal components analysis (PCA) transformation.

only works well in source domain but poorly in target domain. The adversarial training based baseline scheme of *DANN* [5] aligns most samples in the source and target domains well. When applying the proposed go-getting domain label (Go-labels) technique into DANN, the scheme of *DANN + Go-labels (ours)* achieves a much better domain alignment results, where the clusters with the same class are more compact and less data points scatter at the boundaries between clusters. This group of visualization results validates the effectiveness of our Go-labels for adversarial domain adaptation.

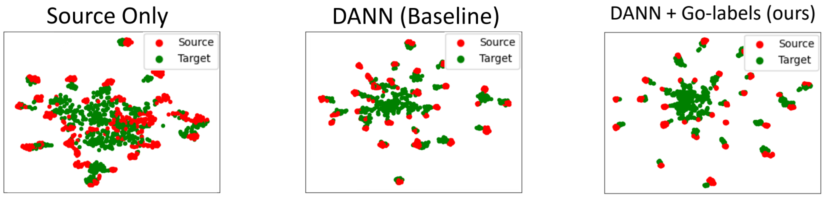


Fig. 4: Visualization of t-SNE distributions, where samples are from source webcam (W) and target amazon (A) domains of Office31.

Feature Map Visualization. Except t-SNE visualization results, in Fig. 5, we also visualize the learned feature maps of *DANN (Baseline)* and *DANN + Go-labels (ours)* by Grad-CAM [26] w.r.t. object category classification. This group of experimental visualization aims to explore whether the go-getting domain label (Go-labels) could help the feature extractor to learn better domain-invariant and object-focus visual representations. We see that the baseline scheme *DANN (Baseline)* is prone to ignore some discriminative regions, which impedes the transferability across domains. In contrast, with Go-labels, the learned feature representations could better focus on the discriminative regions that related to foreground objects, enabling a higher classification accuracy.

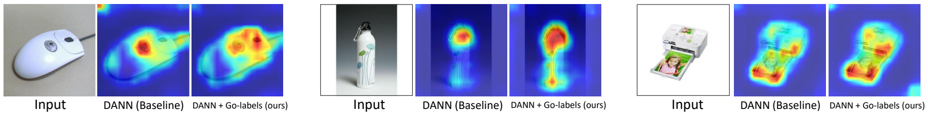


Fig. 5: Visualization of inner feature maps, where samples are also from source webcam (W) and target amazon (A) domains of Office31.

References

1. Chen, Y., Zhu, X., Gong, S.: Instance-guided context rendering for cross-domain person re-identification. In: ICCV. pp. 232–242 (2019)
2. Cui, S., Wang, S., Zhuo, J., Su, C., Huang, Q., Tian, Q.: Gradually vanishing bridge for adversarial domain adaptation. In: CVPR. pp. 12455–12464 (2020)
3. Fischer, G.: Lifelong learning—more than training. Journal of Interactive Learning Research **11**(3), 265–294 (2000)
4. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: ICML. pp. 1180–1189. PMLR (2015)
5. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. The Journal of Machine Learning Research **17**(1), 2096–2030 (2016)
6. Gardner, M.W., Dorling, S.: Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. Atmospheric environment **32**(14-15), 2627–2636 (1998)
7. Huang, J., Feris, R.S., Chen, Q., Yan, S.: Cross-domain image retrieval with a dual attribute-aware ranking network. In: ICCV. pp. 1062–1070 (2015)
8. Hull, J.J.: A database for handwritten text recognition research. IEEE TPAMI **16**(5) (1994)
9. Jin, X., Lan, C., Zeng, W., Chen, Z.: Global distance-distributions separation for unsupervised person re-identification. In: ECCV. pp. 735–751. Springer (2020)
10. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: IEEE (1998)
11. Li, W.H., Xiang, S., Nie, W.Z., Song, D., Liu, A.A., Li, X.Y., Hao, T.: Joint deep feature learning and unsupervised visual domain adaptation for cross-domain 3d object retrieval. Information Processing & Management **57**(5), 102275 (2020)
12. Liu, A., Xiang, S., Li, W., Nie, W., Su, Y.: Cross-domain 3d model retrieval via visual domain adaptation. In: IJCAI. pp. 828–834 (2018)
13. Liu, J., Zha, Z.J., Chen, D., Hong, R., Wang, M.: Adaptive transfer network for cross-domain person re-identification. In: CVPR. pp. 7202–7211 (2019)
14. Liu, Y., Deng, J., Gao, X., Li, W., Duan, L.: Bapa-net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation. In: ICCV. pp. 8801–8811 (2021)
15. Long, M., Cao, Z., Wang, J., Jordan, M.I.: Conditional adversarial domain adaptation. In: NeurIPS. pp. 1640–1650 (2018)
16. Loshchilov, I., Hutter, F.: Stochastic gradient descent with warm restarts. In: ICLR (2017)
17. Lv, F., Liang, T., Chen, X., Lin, G.: Cross-domain semantic segmentation via domain-invariant interactive relation transfer. In: CVPR. pp. 4334–4343 (2020)
18. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning. In: NeurIPS-W (2011)
19. Oliphant, T.E.: A guide to NumPy, vol. 1. Trelgol Publishing USA (2006)
20. Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., Wermter, S.: Continual lifelong learning with neural networks: A review. Neural Networks **113**, 54–71 (2019)
21. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: ICCV. pp. 1406–1415 (2019)
22. Peng, X., Usman, B., Kaushik, N., Hoffman, J., Wang, D., Saenko, K.: Visda: The visual domain adaptation challenge. arXiv preprint arXiv:1710.06924 (2017)

23. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: ECCV. pp. 213–226. Springer (2010)
24. Saito, K., Ushiku, Y., Harada, T., Saenko, K.: Strong-weak distribution alignment for adaptive object detection. In: CVPR. pp. 6956–6965 (2019)
25. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: CVPR. pp. 3723–3732 (2018)
26. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: ICCV. pp. 618–626 (2017)
27. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: CVPR (2017)
28. Wang, H., Shen, T., Zhang, W., Duan, L.Y., Mei, T.: Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In: ECCV. pp. 642–659. Springer (2020)
29. Zhang, X., Cao, J., Shen, C., You, M.: Self-training with progressive augmentation for unsupervised cross-domain person re-identification. In: ICCV. pp. 8222–8231 (2019)
30. Zhang, Y., Tang, H., Jia, K., Tan, M.: Domain-symmetric networks for adversarial domain adaptation. In: CVPR. pp. 5031–5040 (2019)