Domain Adaptation with Auxiliary Target Domain-Oriented Classifier *

Jian Liang [1] Dapeng Hu [1] Jiashi Feng [2,1]
[1] National University of Singapore (NUS) [2] Sea AI Lab (SAIL)
liangjian92@gmail.com dapeng.hu@u.nus.edu elefjia@nus.edu.sg

Abstract

Domain adaptation (DA) aims to transfer knowledge from a label-rich but heterogeneous domain to a label-scare domain, which alleviates the labeling efforts and attracts considerable attention. Different from previous methods focusing on learning domain-invariant feature representations, some recent methods present generic semi-supervised learning (SSL) techniques and directly apply them to DA tasks, even achieving competitive performance. One of the most popular SSL techniques is pseudo-labeling that assigns pseudo labels for each unlabeled data via the classifier trained by labeled data. However, it ignores the distribution shift in DA problems and is inevitably biased to source data. To address this issue, we propose a new pseudo-labeling framework called Auxiliary Target Domain-Oriented Classifier (ATDOC). ATDOC alleviates the classifier bias by introducing an auxiliary classifier for target data only, to improve the quality of pseudo labels. Specifically, we employ the memory mechanism and develop two types of non-parametric classifiers, i.e. the nearest centroid classifier and neighborhood aggregation, without introducing any additional network parameters. Despite its simplicity in a pseudo classification objective, ATDOC with neighborhood aggregation significantly outperforms domain alignment techniques and prior SSL techniques on a large variety of DA benchmarks and even scare-labeled SSL tasks.

1. Introduction

Despite remarkable progress in classification tasks over the past decades, deep neural network models still suffer poor generalization performance to another new domain e.g. classifying real-world object images using a classification model trained on simulated object images [54], due to the well-known dataset shift [55] or domain shift [66] problem. Hence, to avoid expensive human labeling and utilize prior related labeled datasets, lots of research efforts have been devoted to developing domain adaptation (DA) methods [21, 20, 27, 67, 59, 29, 40] to transfer knowledge in the label-rich dataset to a label-scare scenario. Depending on the availability of labeled data in the target domain, one can further divide existing DA methods into two categories, i.e., unsupervised DA [20] and semi-supervised DA [59].

This paper mainly focuses on unsupervised DA where no labeled data is available in the target domain during training. Recently, deep unsupervised DA approaches have almost dominated this field with promising results [47, 20, 48, 37, 31, 14], and most of them focus on learning domain-invariant feature representations that achieve a small error on the source domain at the same time. With the assumption about covariate shift in [2], the learned representations together with the classifier built on the source domain are able to generalize well to the target domain. However, the strict assumption does not always hold in real-world applications. Another line of research ignores transferable representation learning but directly resorts to semi-supervised learning (SSL) techniques for the DA problems [18, 8, 57, 16, 30], where the target domain could be readily treated as the unlabeled set in SSL. For instance, MixMatch [3], a popular SSL approach, has been successfully applied by [57] that wins prizes of the *VisDA 2019 Challenge*. But these SSL-based DA methods may fail to classify target samples far away from the source domain due to the ignorance of domain shift.

One of the most popular SSL techniques—pseudo labeling [38] iteratively assigns pseudo labels corresponding to the maximum prediction scores for each unlabeled data in the target domain and then retrains the network with the pseudo-labeled data in a supervised manner. However, the network is inevitably biased to the labeled source data during training, giving low-quality pseudo labels and propagating errors in the target domain. To tackle this issue, we propose a new pseudo-labeling framework termed Auxiliary Target Domain-Oriented Classifier (ATDOC) for DA problems. Generally, ATDOC attempts to alleviate the labeling bias by introducing an auxiliary classifier for target data only. In particular, we design two types of non-parametric classifiers, i.e., nearest centroid classifier (NC) and neighborhood aggregation (NA), to avoid additional network parameters. Both class centroids and local neighborhood structures are

*This work was partially supported by AISG-100E-2019-035, MOE2017-T2-2-151, NUS_ECRA_FY17_P08 and CRP20-2017-0006.

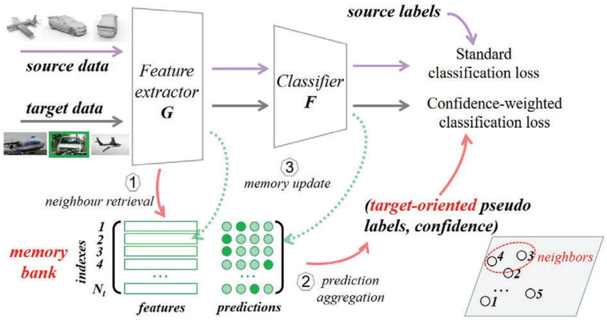


Figure 1. The pipeline of our proposed framework ATDOC with neighborhood aggregation (NA) for UDA. Different from existing methods that mostly rely on feature-level domain alignment, ATDOC addresses domain shift by alleviating the classifier bias via an auxiliary classifier for target data only during adaptation. ATDOC-NA employs a memory bank and develops neighborhood aggregation to help build the domain-specific classifier $\mathbf{F}_t$, and expects to generate unbiased accurate pseudo labels along with confidence weights for unlabeled data.

capable of representing the target domain, thus the generated target-oriented pseudo labels are fairly unbiased and reliable.

To enable global structure learning with mini-batch optimization, we introduce a memory module to store information over all the unlabeled target samples. Besides, since no labeled data is available in the target domain, noisy pseudo labels are directly exploited as an alternative. Specifically, concerning the NC classifier training, we follow [72, 75] and construct a memory bank to store feature representations of the class centroids. The exponential moving average strategy is adopted to dynamically update centroids in the memory bank. Built on the centroids, the NC classifier readily offers a target-oriented pseudo-label for each unlabeled data. Compared with coarse class centroids, training a NA classifier needs to construct a large memory bank that consists of the features along with soft predictions over all the target samples. Through simple aggregation, NA feasibly generates confidence as an instance weight besides the one-hot pseudo label. An overview of ATDOC-NA is shown in Fig. 1. Generally, the network at first several iterations is still biased to source data, but as the number of training iterations increases, retraining with target-oriented pseudo labels gradually adjusts the network to unlabeled target data, achieving domain alignment via optimizing two classification objectives.

Our contributions are summarized as follows: (i) we propose ATDOC, a new framework to combat classifier bias that provides a new perspective of addressing domain shift. (ii) we exploit the memory bank and develop two types of non-parametric classifiers, not involving complicated network architectures with extra parameters. (iii) despite its simplicity, ATDOC achieves competitive or better results than prior state-of-the-arts under a variety of DA settings, e.g., partial-set unsupervised DA [5] and semi-supervised DA. ATDOC can be seamlessly integrated into existing domain-invariant feature learning methods and further boost their adaptation performance. Besides, we study an SSL setting with only a few annotated data points available and find ATDOC also performs better than other SSL techniques even without domain shift.

2. Related Work

2.1. Deep Domain Adaptation

Deep domain adaptation methods [15, 34, 73]. aim to learn more transferable representations by embedding domain adaptation in the pipeline of deep learning. Generally, the weights of the deep architecture containing a feature encoder and a classifier layer are shared for both domains, and various distribution discrepancy measures [69, 47, 19] are developed to promote domain confusion in the feature space. Among them, maximum mean discrepancy (MMD) [23] and $\mathcal{H}\Delta\mathcal{H}$-distance [2] are two most favored measures. To circumvent the problem that marginal distribution alignment cannot guarantee different domains are semantically aligned, following works [48, 14] exploit pseudo labels on the target domain to perform conditional distribution alignment. The learned classifier still fails to generalize well on the target domain, as it is mainly built on the labeled source data.

Another line of research [49, 56, 45] exploits the individual characteristics of each domain by dropping the weight-sharing assumption fully or partially. Shu *et al.* [63] propose non-conservative domain adaptation and incrementally refine the preciously learned classification boundary to fit the target domain only. With the classifier shared, Tzeng *et al.* [68] first learn the source feature encoder and then the target feature encoder sequentially. While Bousmalis *et al.* [4] jointly learn the domain-shared encoder and domain-specific private encoders. Besides, Chang *et al.* [7] share all other model parameters but specialize batch normalization layers within the feature encoder. Liang *et al.* [45] learn the target-specific feature extractor while only operating on the hypothesis induced from the source data. Long *et al.* [49] pursue classifier adaptation through a new residual transfer module that bridges the source classifier and target classifier. Compared with these methods, ATDOC does not introduce any new layers and aims to learn one shared classifier for both domains with a virtual target-oriented classifier.

2.2. Semi-supervised Learning with Regularization

Semi-supervised learning (SSL) is a classical machine learning approach, which aims to exploit unlabeled data to produce a considerable improvement in learning accuracy. Typically, besides the classification objective for labeled data, SSL methods [80, 70] resort to the cluster assumption or low-density separation assumption to fully exploit unlabeled data as regularization, e.g., entropy minimization [22].

The virtual adversarial training loss [51] is proposed to measure local smoothness of the conditional label distribution around each input data point against local perturbation. By contrast, temporal ensembling [36] is developed to pursue the consistency regularization over multiple training epochs. Further, graph-based techniques e.g. label propagation [81] and manifold regularization [1] are favored by a majority of researchers before the deep learning area, which has been effectively extended in recent works [33, 28]. In addition, data augmentation techniques such as MixUp and random crop are also utilized in MixMatch [3] which achieves state-of-the-art results in multiple SSL benchmarks.

In fact, unsupervised DA can be considered as a special case of transductive SSL where labeled data and unlabeled data are sampled from different distributions. Recent studies [8, 16, 30] show that regularization terms on unlabeled data without explicit feature-level domain alignment achieve promising adaptation results. In particular, the MaxSquare loss is developed in [8] to prevent the training process from being dominated by easy-to-transfer samples in the target domain. In contrast, the diversity of conditional predictions is considered through batch nuclear-norm maximization [16] and class confusion minimization [30], respectively.

2.3. Pseudo-labeling in Transductive Learning

Pseudo-labeling is a heuristic approach to transductive SSL, providing an alternative solution to regularization. As a typical example, 'Pseudo-Label' [38] progressively treats high-confidence predictions on unlabeled data as true labels (called pseudo labels) and employ a standard cross-entropy loss during re-training. Following works [62, 17] incorporate pseudo labels to perform discriminative clustering for features of unlabeled data. To alleviate noises in the instance-wise pseudo labels, [28] relies on graph-based label propagation and [83] considers the class imbalance problem, obtaining better pseudo labels. Besides, the weights of different pseudo labels are considered, [13] adopts a curriculum learning strategy and [82] treats pseudo-labels as continuous latent variables jointly optimized via alternating optimization. Our method is different from all such prior work in that pseudo-labels are inferred by an auxiliary parameter-free target-oriented classifier effectively and efficiently.

2.4. Transductive Learning with Memory Bank

A memory bank can be read and written to remember past facts, enabling global structure learning with mini-batch optimization [64]. A recent study [10] first exploits the memory mechanism in the network training for SSL and computes the memory prediction for each training sample by the key addressing and value reading. Inspired by instance discrimination [74], Saito *et al*. [60] employ a memory bank and propose an entropy minimization loss to encourage neighborhood clustering in the target domain. Xie *et al*. [75] utilize the memory bank to compute centroids from both domains for centroid matching. Besides, Zhong *et al*. [79] leverage an exemplar memory module that saves up-to-date features for target data and computes the invariance learning loss for unlabeled target data. Among them, [10] is the most closely related work to ours, but [10] is merely proposed for SSL that only utilizes the labeled data for memory update and ignores self-learning in the unlabeled data. Besides, in contrast to supervised center loss [72], the memory bank in the ATDOC framework e.g. storing class centroids is updated in an unsupervised way.

3. Methodology

For the unsupervised DA (UDA) task, we are given a labeled source domain $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ with K categories and an unlabeled target domain $\mathcal{D}_{tu} = \{(x_i^t)\}_{i=1}^{N_{tu}}$, while in semi-supervised DA (SSDA), we are given an additional labeled subset of the target domain $\mathcal{D}_{tl} = \{(x_i^t, y_i^t)\}_{i=1}^{N_{tl}}$. $\mathcal{D}_t = \mathcal{D}_{tu} \cup \mathcal{D}_{tl}$ denotes the entire target domain, and $\mathcal{D}_{tl}$ is empty in UDA. This paper mainly focuses on the vanilla closed-set setting where two domains share the same categories. The ultimate goal of both UDA and SSDA is to label the target samples in $\mathcal{D}_{tu}$ via training the model on $\mathcal{D}_s \cup \mathcal{D}_t$.

As shown in Fig. 1, we employ the widely-used architecture [19] which consists of two basic modules, a feature extractor $\mathbf{G}$ and a classifier $\mathbf{F}$. Based on where to align, DA approaches can be roughly categorized into three main cases, i.e., pixel-level [26, 61], feature-level [19, 68, 48, 41] and output-level [8, 16, 30]. Compared with time-consuming pixel-level transfer methods, feature-level domain alignment is cheap and flexible, attracting the most attention. Belonging to the output-level domain alignment, our ATDOC framework well combats the classifier bias and can be seamlessly combined with feature-level DA approaches.

3.1. A Closer Look at Pseudo-labeling

To fully utilize the unlabeled data, following classic self-training [80], Lee [38] presents a simple method with training deep neural networks for SSL. It picks up the class $\hat{y}$ with the maximum predicted probability as true labels each time the weights are updated.

$$\begin{aligned} \hat{y}_i &= \arg\max_k p_{i,k}, \ i = 1, 2, \cdots, N_{tu}, \\ \mathcal{L}_{pl} &= -\frac{\alpha}{N_{tu}} \sum\nolimits_{i=1}^{N_{tu}} \log p_{i,\hat{y}_i}, \end{aligned} \tag{1}$$

where $p_i = F(G(x_i^t))$ is the K-dimensional prediction. The coefficient α before the pseudo-labeled term is properly designed to grow from 0 to 1 gradually, which can mitigate the noises in the pseudo labels at early iterations to some degree, avoiding the error accumulation.

Since the pseudo labels are not equally confident, in this work, we readily take the maximum predicted probabilities as weights and incorporate them into the standard cross-entropy loss, forming the following objective to adapt the model with unlabeled data (*termed as 'pseudo-labeling'*),

$$\hat{y}_i = \arg\max_k p_{i,k},\ i = 1, 2, \cdots, N_t, \\ \mathcal{L}_{pl}^{ours} = -\frac{\lambda}{N_{tu}} \sum\nolimits_{i=1}^{N_{tu}} p_{i,\hat{y}_i} \log p_{i,\hat{y}_i}. \tag{2}$$

Different from [38] where α involves two hyper-parameters, we adopt a simplified linear scheduler for λ in this work.

As stated in [38], $\mathcal{L}_{pl}$ favors a low-density separation between classes and is in effect equivalent to entropy regularization [22] that is employed to reduce the class overlap. However, both regularization approaches [38, 22] and another recent regularization method [8] ignore the structure of unlabeled data and only focus on the instance-wise prediction itself. Considering the data structure especially the diversity among predictions of unlabeled data, Jin *et al*. [30] propose to minimize the pair-wise class confusion within a mini-batch of training data. Cui *et al*. [16] pursue a lower output matrix rank within a mini-batch to ensure both discriminability and diversity. Both approaches have been proven to achieve much better results than vanilla entropy minimization, implying that the structure of the classification output matrix is essential for unlabeled data.

3.2. Auxiliary Target Domain-Oriented Classifier

In this paper, we propose a new regularization approach called Auxiliary Target Domain-Oriented Classifier (ATDOC) that fully exploits the structure of unlabeled data to get reliable pseudo labels with the presence of domain shift. In particular, ATDOC aims to learn an extra specific classifier F_t for the target domain. However, it is quite challenging to learn F_t without labeled target data. Fortunately, according to prior studies [48, 52], there exist some source-like samples whose output predictions are reliable, which can be used to help build the classifier proposed here and teach the remaining samples sequentially. To avoid the trivial sample selection and alternate training, ATDOC employs a memory module that collects or stores information of all the target samples to generate more accurate pseudo labels. In the following, we develop two types of non-parametric target-oriented classifiers and describe the details of them.

3.2.1 Nearest centroid Classifier (NC)

Nearest centroid (NC) classifier is one of the most simple yet powerful classifiers that merely requires the class centroids. Motivated by prior works [43, 45] which utilize class centroids to bridge the domain gap for DA problems, we describe the data structure of the target domain with its class centroids. Specifically, we introduce a memory bank to store the information of target-oriented class centroids and dynamically generate pseudo-labels with the NC classifier for unlabeled data in each mini-batch.

Memory bank update. Since the target data is unlabeled during training, we first obtain the one-hot pseudo labels via $\hat{y}_i = \arg\max_k p_{i,k}$ and then utilize them to update the centroids in the memory bank via the exponential moving averaging (EMA) strategy,

$$c_j = \sum\nolimits_{i \in B_t} \mathbb{1}_{[j=\hat{y}_i]} G(x_i^t) / \sum\nolimits_{i \in B_t} \mathbb{1}_{[j=\hat{y}_i]}, \\ c_j^m = \gamma c_j + (1-\gamma) c_j^m,\ m = 1, 2, \cdots, K, \tag{3}$$

where B_t denotes the index set of a min-batch from the target domain, and γ is the smoothing parameter fixed to 0.1 as default.

Pseudo-labeling. With the class centroids, we readily construct the NC classifier and generate the pseudo labels for each unlabeled datum x_i^t below,

$$\hat{y}_i = \arg\min_{j=1}^{K} d\left(G(x_i^t), c_j^m\right),\ i = 1, 2, \cdots, N_t, \tag{4}$$

where $d(\cdot, \cdot)$ measures the distance between features and centroids, here we adopt the cosine distance as default. Finally, a standard cross-entropy loss is developed as

$$\mathcal{L}_{nc} = -\frac{\lambda}{N_{tu}} \sum\nolimits_{i=1}^{N_{tu}} \log p_{i,\hat{y}_i}. \tag{5}$$

3.2.2 Neighborhood Aggregation (NA)

Class centroids can only coarsely characterize the domain structure, which fails to depict the local data structure. Therefore, we follow the idea of message passing via neighbors and further develop a neighborhood aggregation (NA) strategy as the classifier. Different from the NC classifier, we need to construct a large memory bank to store information e.g. features and soft predictions of each target data.

Memory bank update. To avoid ambiguity in the target predictions, we directly sharpen the output predictions $p_i, x_i \in \mathcal{D}_t$ via temperature scaling [24, 3] with $T = 0.5$,

$$\check{p}_{i,k}^m = p_{i,k}^{\frac{1}{T}} / \sum\nolimits_k p_{i,k}^{\frac{1}{T}}. \tag{6}$$

As $T \to 0$, the probability collapses to a point mass like Pseudo-Label [38]. Then, the sharpened prediction $\check{p}_i^m$ along with its L2-normalized feature vector $f_i^m = G(x_i^t)/\|G(x_i^t)\|_2$ is written in the memory module based on the index. Here we do not adopt any moving average strategies for updating the values in the memory.

Neighborhood aggregation. The memory module keeps updating every mini-batch, and training a parametric classifier involving extra parameters would be time-consuming. To address this, we present a non-parametric neighborhood aggregation strategy as $\mathbf{F}_t$. We first retrieve m nearest neighbors from the memory module for each sample in the current mini-batch based on the cosine similarity between their features $G(x_i^t)$ and all the features in the memory bank f_j^m. Then, we aggregate corresponding soft predictions of these nearest neighbors by taking the average,

$$\hat{q}_i = \frac{1}{m} \sum\nolimits_{j \neq i, j \in \mathcal{N}_i} \check{p}_j, \tag{7}$$

where $\mathcal{N}_i$ denotes the index set of neighbors in the memory module for the data point x_i^t. In this manner, we obtain a new probability prediction via learning on the entire target data. Note that our strategy indeed considers the global structure beyond regularization within a mini-batch like [16, 30].

Pseudo-labeling. For each unlabeled datum x_i^t, we get the pseudo label $\hat{y}_i$ by choosing the category index with the maximum probability prediction $\hat{q}_i$, i.e., $\hat{y}_i = \arg\max_k \hat{q}_{i,k}$. Considering different neighborhoods $\mathcal{N}_i$ lie in regions of different densities, it is desirable to assign a larger weight for the target data in a neighborhood of higher density. Intuitively, the larger the maximum value $\hat{q}_{i,\hat{y}_i}$ is, the higher density it will be for the region the datum lies in. Thus, we directly utilize $\hat{q}_{i,\hat{y}_i}$ as the confidence (weight) for the pseudo label $\hat{q}_i$. Finally, a confidence-weighted cross-entropy loss is imposed on the unlabeled target data as below,

$$\mathcal{L}_{na} = -\frac{\lambda}{N_{tu}} \sum\nolimits_{i=1}^{N_{tu}} \hat{q}_{i,\hat{y}_i} \log p_{i,\hat{y}_i}. \tag{8}$$

Concerning the labeled data in $\mathcal{D}_s \cup \mathcal{D}_{tl}$, we employ the stand cross-entropy loss with label-smoothing regularization [65], denoted as $\mathcal{L}_{lsr}^s$ and $\mathcal{L}_{lsr}^t$, respectively. Integrating these losses together, we obtain the final objective as follows,

$$\mathcal{L} = \mathcal{L}_{lsr}^s(\mathcal{D}_s) + \mathcal{L}_{lsr}^t(\mathcal{D}_{tl}) + \mathcal{L}_{nc/na}(\mathcal{D}_{tu}). \tag{9}$$

Actually, we can readily incorporate $\mathcal{L}_{nc}$ or $\mathcal{L}_{na}$ into other domain alignment methods like CDAN [48] as an additional loss. Besides, for SSL methods like MixMatch [3], we just replace $p_{model}(y|u)$ with the **one-hot encoding** of $\hat{y}$ in the pseudo-labeling step of NC and NA, respectively.

4. Experiments

4.1. Setup

Datasets. Office-31 [58] is the most widely-used benchmark in the DA field, which consists of 3 different domains in 31 categories: Amazon (**A**) with 2,817 images, Webcam (**W**) with 795 images, and DSLR (**D**) with 498 images. There are 6 transfer tasks for evaluation in total.

Office-Home [71] is another popular benchmark that consists of images from 4 different domains: Artistic (**Ar**) images, Clip Art (**Cl**), Product (**Pr**) images, and Real-World (**Re**) images, totally around 15,500 images from 65 different categories. All 12 transfer tasks are selected for evaluation.

VisDA-C [54] is a large-scale benchmark used for the *VisDA 2017 Challenge* that consists of 2 very distinct kinds of images from twelve common object classes, i.e., 152,397 synthetic images and 55,388 real images. We focus on the challenging synthetic-to-real transfer task.

Table 1. Accuracy (%) on Office for UDA (ResNet-50). Best (**bold red**), second best (*italic blue*). [†: mean values except D↔W]

Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg.	Avg.†
ResNet-50 [25]	78.3	70.4	57.3	93.4	61.5	98.1	76.5	66.9
MinEnt [22]	90.7	89.4	67.1	97.5	65.0	**100.**	85.0	78.1
MCC [30]	92.1	94.0	74.9	98.5	75.3	**100.**	89.1	84.1
BNM [16]	92.2	94.0	74.9	98.5	75.3	**100.**	89.2	84.1
Pseudo-labeling	88.7	89.1	65.8	98.1	66.6	99.6	84.7	77.6
ATDOC-NC	95.2	91.6	74.6	*99.1*	74.7	**100.**	89.2	84.0
ATDOC-NA	94.4	94.3	75.6	98.9	75.2	99.6	89.7	84.9
CDAN+E [48]	94.5	94.2	72.8	98.6	72.2	**100.**	88.7	83.4
+ BSP [9]	94.5	95.0	73.9	98.3	75.7	**100.**	89.6	84.8
+ MCC [30]	94.1	94.7	75.4	99.0	75.7	**100.**	89.8	85.0
+ BNM [16]	94.9	94.3	*75.8*	99.0	75.9	**100.**	*90.0*	*85.2*
+ Pseudo-labeling	91.5	93.1	72.5	97.8	72.7	99.8	87.9	82.4
+ ATDOC-NC	*96.3*	93.6	74.3	*99.1*	75.4	**100.**	89.8	84.9
+ ATDOC-NA	95.4	94.6	**77.5**	98.1	**77.0**	99.7	**90.4**	**86.1**
MixMatch [3]	88.5	84.6	63.3	96.1	65.0	99.6	82.9	75.4
w/ Pseudo-labeling	89.0	86.0	65.8	96.2	65.6	99.6	83.7	76.6
w/ ATDOC-NC	91.3	86.4	66.0	97.4	64.4	99.4	84.1	77.0
w/ ATDOC-NA	92.1	91.0	70.9	98.6	*76.2*	99.6	88.1	82.6
SAFN+ENT [76]	90.7	90.1	73.0	98.6	70.2	99.8	87.1	81.0
CRST [82]	88.7	89.4	72.6	98.9	70.9	**100.**	86.8	80.4
SHOT [45]	94.0	90.1	74.7	98.4	74.3	*99.9*	88.6	83.3
CADA-P [35]	95.6	**97.0**	71.5	**99.3**	73.1	**100.**	89.5	84.3
ATM [41]	**96.4**	*95.7*	74.1	**99.3**	73.5	**100.**	89.8	84.9

DomainNet-126 is a subset of DomainNet [53], by far the largest UDA dataset with 6 distinct domains and approximately 0.6 million images distributed among 345 categories. Following [59], we pick 126 classes in 4 domains i.e. Real (**R**), Clipart (**C**), Painting (**P**), and Sketch (**S**) for evaluation.

Implementation[1] Details. We report the average accuracy over 3 random trials. All the methods including domain alignment methods [48, 9], semi-supervised methods [3], and regularization approaches [38, 22, 8, 30, 16] are implemented based on **PyTorch** and reverse validation [44, 78] is conducted to select hyper-parameters. Note that MixMatch [3] could be considered as a strong domain adaptation baseline [57]. Besides, we select other state-of-the-art UDA approaches [76, 82, 35, 41, 39, 42] and SSDA approaches [59, 29, 40, 32] for further comparison. We adopt a linear rampup scheduler from 0 to λ for all methods, and $\lambda = 0.1$ is fixed for ATDOC-NC, and $\lambda = 0.2, m = 5$ is fixed for ATDOC-NA throughout this paper. We adopt mini-batch SGD to learn the feature encoder by fine-tuning from the ImageNet pre-trained model with the learning rate 0.001, and new layers (bottleneck layer and classification layer) from scratch with the learning rate 0.01. We use the suggested training settings in [48], including learning rate scheduler, momentum (0.9), weight decay (1e^{-3}), bottleneck size (256), and batch size (36).

[1] https://github.com/tim-learn/ATDOC/

Table 2. Per-class accuracy (%) on VisDA-C validation set using a ResNet-101 backbone.

Method	aero	bike	bus	car	horse	knife	mbike	person	plant	skbrd	train	truck	Mean
ResNet-101 [25]	67.7	27.4	50.0	61.7	69.5	13.7	85.9	11.5	64.4	34.4	84.2	19.2	49.1
MinEnt [22]	88.6	29.5	82.5	75.8	88.7	16.0	**93.2**	63.4	94.2	40.1	87.3	12.1	64.3
BNM [16]	91.1	69.0	76.7	64.3	89.8	61.2	90.8	74.8	90.9	66.6	88.1	46.1	75.8
MCC [30]	92.2	82.9	76.8	66.6	90.9	78.5	87.9	73.8	90.1	76.1	87.1	41.0	78.7
Pseudo-labeling	90.9	74.6	73.2	55.8	89.6	64.6	86.8	68.7	90.7	64.8	89.5	47.7	74.7
ATDOC-NC	91.1	60.1	78.4	72.2	88.1	*97.6*	86.9	55.9	79.2	64.9	88.4	31.9	74.6
ATDOC-NA	93.7	83.0	76.9	58.7	89.7	95.1	84.4	71.4	89.4	80.0	86.7	55.1	80.3
CDAN+E [48]	94.3	60.8	79.9	72.7	89.5	86.8	92.4	*81.4*	88.9	72.9	87.6	32.8	78.3
+ BSP [9]	93.7	58.1	80.5	69.3	89.7	86.4	92.9	78.4	88.5	74.7	88.4	33.0	77.8
+ BNM [16]	93.6	68.3	78.9	70.3	91.1	82.8	*93.0*	78.7	90.9	76.5	89.1	40.9	79.5
+ MCC [30]	93.5	72.5	72.5	72.9	91.6	88.7	92.1	75.1	92.7	79.4	87.8	53.3	81.0
+ Pseudo-labeling	94.1	70.4	78.1	68.7	90.3	77.9	92.1	78.5	90.6	76.9	88.5	44.5	79.2
+ ATDOC-NC	94.3	64.0	80.1	66.2	89.1	92.1	91.0	75.4	86.9	76.3	87.2	43.4	78.8
+ ATDOC-NA	93.0	77.4	83.4	62.3	91.5	88.4	91.8	77.1	90.9	86.4	85.8	48.2	81.4
MixMatch [3]	93.9	71.8	**93.5**	*82.1*	95.3	0.7	90.8	38.1	94.5	*96.0*	86.3	2.2	70.4
w/ Pseudo-labeling	*95.0*	75.5	*92.5*	79.5	**96.0**	0.4	91.1	23.7	**95.2**	95.1	82.5	0.6	68.9
w/ ATDOC-NC	93.7	77.2	71.6	71.7	92.1	0.1	86.3	52.9	86.7	**96.8**	**92.9**	45.8	72.3
w/ ATDOC-NA	**95.3**	*84.7*	82.4	75.6	*95.8*	**97.7**	88.7	76.6	94.0	91.7	*91.5*	*61.9*	**86.3**
SAFN [76]	93.6	61.3	84.1	70.6	94.1	79.0	91.8	79.6	89.9	55.6	89.0	24.4	76.1
CRST [82]	88.0	79.2	61.0	60.0	87.5	81.4	86.3	78.8	85.6	86.6	73.9	**68.8**	78.1
DTA [39]	93.7	82.2	85.6	**83.8**	93.0	81.0	90.7	**82.1**	*95.1*	78.1	86.4	32.1	81.5
SHOT [45]	94.3	**88.5**	80.1	57.3	93.1	94.9	80.7	80.3	91.5	89.1	86.3	58.2	*82.9*

Table 3. Accuracy (%) on Office-Home for closed-set UDA (ResNet-50).

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.
ResNet-50 [25]	44.9	66.3	74.3	51.8	61.9	63.6	52.4	39.1	71.2	63.8	45.9	77.2	59.4
MinEnt [22]	51.0	71.9	77.1	61.2	69.1	70.1	59.3	48.7	77.0	70.4	53.0	81.0	65.8
BNM [16]	56.7	77.5	81.0	67.3	76.3	77.1	65.3	55.1	82.0	73.6	57.0	84.3	71.1
MCC [30]	56.3	77.3	80.3	67.0	77.1	77.0	66.2	55.1	81.2	73.5	57.4	84.1	71.0
Pseudo-labeling	54.1	74.1	78.4	63.3	72.8	74.0	61.7	51.0	78.9	71.9	56.6	81.9	68.2
ATDOC-NC	54.4	77.6	80.8	66.5	75.6	75.8	65.9	51.9	81.1	72.7	57.0	83.5	70.2
ATDOC-NA	58.3	**78.8**	**82.3**	**69.4**	*78.2*	**78.2**	67.1	56.0	*82.7*	72.0	58.2	*85.5*	*72.2*
CDAN+E [48]	54.6	74.1	78.1	63.0	72.2	74.1	61.6	52.3	79.1	72.3	57.3	82.8	68.5
+ BSP [9]	57.1	73.4	77.5	64.2	71.8	74.3	64.0	56.7	81.0	73.4	59.1	83.3	69.6
+ BNM [16]	58.1	77.2	81.1	67.5	75.3	77.2	65.5	*56.8*	82.6	74.1	*59.9*	84.6	71.7
+ MCC [30]	*58.9*	77.6	80.7	67.0	75.1	77.1	65.8	*56.8*	82.2	73.9	59.8	84.5	71.6
+ Pseudo-labeling	57.3	76.6	79.2	66.6	74.0	76.6	66.1	53.6	81.0	*74.3*	58.9	84.2	70.7
+ ATDOC-NC	55.9	76.3	80.3	63.8	75.7	76.4	63.9	53.7	81.7	71.6	57.7	83.3	70.0
+ ATDOC-NA	**60.2**	77.8	*82.2*	*68.5*	**78.6**	77.9	**68.4**	**58.4**	**83.1**	**74.8**	**61.5**	**87.2**	**73.2**
SAFN [76]	52.0	71.7	76.3	64.2	69.9	71.9	63.7	51.4	77.1	70.9	57.1	81.5	67.3
CADA-P [35]	56.9	76.4	80.7	61.3	75.2	75.2	63.2	54.5	80.7	73.9	**61.5**	84.1	70.2
DCAN [42]	54.5	75.7	81.2	67.4	74.0	76.3	*67.4*	52.7	80.6	74.1	59.1	83.5	70.5
SHOT [45]	57.1	*78.1*	81.5	68.0	*78.2*	*78.1*	*67.4*	54.9	82.2	73.3	58.8	84.3	71.8

4.2. Results

▷ **Results of Closed-set Unsupervised DA (UDA)**. We use three datasets as introduced above for vanilla UDA tasks, with results shown in Tables 1∼3. On the small-sized Office-31 dataset, we first study different SSL regularization approaches when integrated with the source classification loss only. It is obvious that both BNM [16] and MCC [30] consistently perform better than instance-wise regularization methods like MinEnt [22], which verifies the importance of local diversity. Besides, ATDOC-NC achieves competitive results against BNM, outperforming MinEnt and Pseudo-labeling. ATDOC-NA outperforms BNM in 4 out of 6 tasks, obtaining the best average accuracy. When combined with one popular UDA method i.e. CDAN+E [48], the average accuracy increases accordingly, and ATDOC-NA still performs the best. Since Office-31 is relatively small, MixMatch [3]

Table 4. Accuracy (%) on DomainNet-126 for Semi-supervised DA (SSDA) using a ResNet-34 backbone.

Method	C → S		P → C		P → R		R → C		R → P		R → S		S → P		Average	
	1-shot	3-shot	1-shot	3-shot	1-shot	3-shot	1-shot	3-shot	1-shot	3-shot	1-shot	3-shot	1-shot	3-shot	1-shot	3-shot
ResNet-34 [25]	54.8	57.9	59.2	63.0	73.7	75.6	61.2	63.9	64.5	66.3	52.0	56.0	60.4	62.2	60.8	63.6
MinEnt [22]	56.3	61.5	67.7	71.2	76.0	78.1	66.1	71.6	68.9	70.4	60.0	63.5	62.9	66.0	65.4	68.9
MCC [30]	56.8	60.5	62.8	66.5	75.3	76.5	65.5	67.2	66.9	68.1	57.6	59.8	63.4	65.0	64.0	66.2
BNM [16]	58.4	62.6	69.4	72.7	77.0	79.5	69.8	73.7	69.8	71.2	61.4	65.1	64.1	67.6	67.1	70.3
Pseudo-labeling	62.5	64.5	67.6	70.7	78.3	79.3	70.9	72.9	69.2	70.7	62.0	64.8	67.0	68.6	68.2	70.2
ATDOC-NC	58.1	62.2	65.8	70.2	76.9	78.7	69.2	72.3	69.8	70.6	60.4	65.0	65.5	68.1	66.5	69.6
ATDOC-NA	**65.6**	**66.7**	*72.8*	74.2	**81.2**	**81.2**	**74.9**	**76.9**	**71.3**	**72.5**	*65.2*	64.6	**68.7**	**70.8**	**71.4**	**72.4**
MixMatch [3]	59.3	62.7	66.7	68.7	74.8	78.8	69.4	72.6	67.8	68.8	62.5	65.6	66.3	67.1	66.7	69.2
w/ Pseudo-labeling	59.6	62.6	67.5	69.6	74.8	78.6	70.0	73.0	68.6	69.3	63.2	65.9	66.6	67.3	67.2	69.5
w/ ATDOC-NC	60.2	63.4	65.2	69.5	75.0	78.9	68.4	73.0	68.7	70.1	60.9	64.5	65.3	67.1	66.2	69.5
w/ ATDOC-NA	*64.6*	*65.9*	70.7	72.2	*80.3*	*80.8*	*74.0*	75.2	70.2	71.2	**65.7**	*67.7*	*68.5*	*69.4*	*70.6*	*71.8*
MME [59]	56.3	61.8	69.0	71.7	76.1	78.5	70.0	72.2	67.7	69.7	61.0	61.9	64.8	66.8	66.4	68.9
BiAT [29]	57.9	61.5	71.6	*74.6*	77.0	78.6	73.0	74.9	68.0	68.8	58.5	62.1	63.9	67.5	67.1	69.7
Meta-MME [40]	-	62.8	-	72.8	-	79.2	-	73.5	-	70.3	-	63.8	-	68.0	-	70.1
APE [32]	56.7	63.1	**72.9**	**76.7**	76.6	79.4	70.4	*76.6*	*70.8*	*72.1*	63.0	*67.8*	64.5	66.1	67.6	71.7

Table 5. Accuracy (%) on Office-Home for Partial-set UDA (PDA) using a ResNet-50 backbone.

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.
ResNet-50 [25]	43.5	67.8	78.9	57.5	56.2	62.2	58.1	40.7	74.9	68.1	46.1	76.3	60.9
MinEnt [22]	45.7	73.3	81.6	64.6	66.2	73.0	66.0	52.4	78.7	74.8	56.7	80.8	67.8
MCC [30]	54.1	75.3	79.5	63.9	66.3	71.8	63.3	55.1	78.0	70.4	55.7	76.7	67.5
BNM [16]	54.6	77.2	81.1	64.9	67.9	72.8	62.6	55.7	79.4	70.5	54.7	77.6	68.2
Pseudo-labeling	51.9	70.7	77.5	61.7	62.4	67.8	62.9	54.1	73.8	70.4	56.7	75.0	65.4
ATDOC-NC	59.5	**80.3**	*83.8*	*71.8*	*71.6*	79.7	70.6	*59.4*	82.2	**78.4**	**61.1**	81.5	*73.3*
ATDOC-NA	*60.1*	76.9	**84.5**	**72.8**	71.2	**80.9**	**73.9**	**61.8**	83.8	77.3	*60.4*	80.4	**73.7**
ETN [6]	59.2	77.0	79.5	62.9	65.7	75.0	68.3	55.4	*84.4*	75.7	57.7	*84.5*	70.5
SAFN [76]	58.9	76.3	81.4	70.4	*73.0*	77.8	*72.4*	55.3	80.4	75.8	*60.4*	79.9	71.8
RTNet$_{adv}$ [12]	**63.2**	*80.1*	80.7	66.7	69.3	77.2	71.6	53.9	**84.6**	*77.4*	57.9	**85.5**	72.3

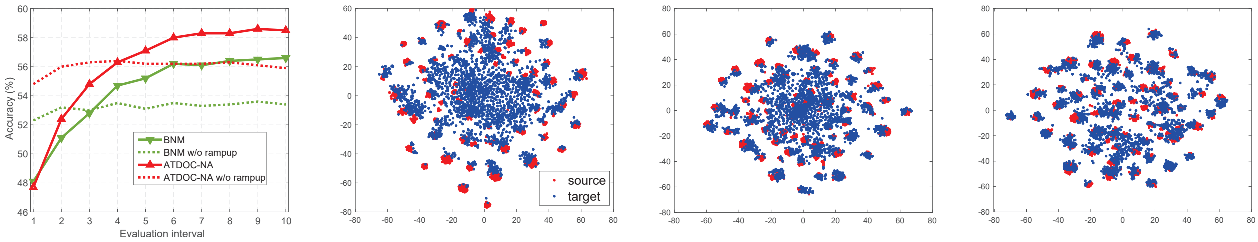
performs worse than CDAN+E. Using pseudo labels provided by ATDOC-NC and ATODC-NA, MixMatch obtains boosted performance. Besides, ATDOC-NA achieves competitive performance with state-of-the-art UDA methods like ATM [41] *without any explicit feature-level alignment*.

As shown in Table 2, ATDOC-NA clearly performs better than BNM and MCC w.r.t. mean accuracy for both situations. Note, ATDOC-NA combined with MixMatch obtains the state-of-the-art mean accuracy 86.3% for VisDA-C, which outperforms recent UDA methods [76, 82, 39, 45]. Taking a closer look at Table 3, we observe similar results for Office-Home that ATDOC-NA beats ATDOC-NC and BNM in terms of the average accuracy. Integrated with the feature-level method—CDAN+E, ATDOC-NA clearly beats the state-of-the-art approach, SHOT [45].

▷ **Results of Semi-supervised DA (SSDA)**. We follow the settings in MME [59] and evaluate SSDA methods on the DomainNet-126 dataset. There exist two SSDA protocols in which each class in the target domain has one or three labeled data points, respectively. As shown in Table 4, ATDOC-NA outperforms both BNM and MCC for both protocols, and MixMatch also benefits from the incorporation of ATDOC. Comparing the results of ATDOC-NA under 1-shot and 3-shot, we find the difference between them is relatively small, implying that ATDOC-NA can fully exploit the unlabeled data to compensate for the scarcity of labeled data. Moreover, compared with prior state-of-the-art SSDA results reported in APE [32], both ATDOC-NA and its combination with MixMatch achieve better performance for both protocols.

▷ **Results of Partial-set UDA (PDA)**. We adopt the standard partial-set UDA setting as [6, 46] on the Office-Home dataset, in which only data of the first 25 classes exist in the target domain. As can be seen from Table 5, ATDOC-NC behaves competitively against ATDOC-NA, this may be because class centroids are somewhat global-level resisting the noise. Faced with the challenging asymmetric label space in PDA, BNM does not clearly outperform MinEnt anymore, but the adaptation results of ATDOC are still fairly promising. Compared with a recent PDA approach, RTNet$_{adv}$ [12], ATDOC-NA obtains a better average accuracy.

▷ **Results of Scare-labeled SSL**. We further study ATDOC in a special SSL case without the domain shift where labeled samples are very scarce. For simplicity, we adopt the same three-shot setting in the aforementioned SSDA, but take labeled target data as the labeled set and unlabeled target data as the unlabeled set, forming the scarce-labeled SSL task. As shown in Table 6, ATDOC-NA performs the best on both Office-Home and DomainNet-126. For such a scarce-labeled SSL task, MixMatch performs badly. The reason may be that labeled data are quite scarce, resulting in low-



(a) Convergence

(b) t-SNE visualizations of features learned by Source-only (ResNet-50), Pseudo-labeling, and ATDOC-NA

Figure 2. For Ar→Cl task on Office-Home, (a) shows the convergence and (b) shows the t-SNE visualizations. (red: Ar, blue: Cl)

Table 6. Accuracy (%) on Office-Home and DomainNet-126 for scarce-labeled SSL (ResNet-50).

Dataset	Office-Home					DomainNet-126				
Method	Ar	Cl	Pr	Re	Avg.	C	P	R	S	Avg.
ResNet-50 [25]	48.7	42.1	68.9	66.6	56.6	41.6	46.2	66.4	33.3	46.9
MinEnt [22]	51.7	44.5	72.4	68.9	59.4	43.8	48.6	68.8	35.4	49.2
MCC [30]	58.9	*47.7*	*77.4*	74.3	64.6	45.7	49.5	70.9	38.5	51.2
BNM [16]	*59.0*	46.0	76.5	71.5	63.2	44.8	47.2	69.9	35.5	49.4
Pseudo-labeling	47.3	41.4	71.4	66.1	56.6	41.0	46.3	72.5	33.1	48.2
ATDOC-NC	56.0	43.4	76.6	72.9	62.2	45.6	51.5	72.2	35.2	51.1
ATDOC-NA	**59.1**	46.6	**78.4**	**75.9**	**65.0**	**54.7**	**60.0**	**75.5**	*38.6*	**57.2**
MixMatch [3]	52.2	41.9	73.1	69.1	59.1	41.2	38.7	64.3	34.2	44.6
w/ Pseudo-labeling	53.4	42.5	72.6	69.5	59.5	40.4	39.1	64.7	34.1	44.6
w/ ATDOC-NC	54.6	44.4	72.7	71.0	60.7	41.2	38.3	63.4	34.4	44.3
w/ ATDOC-NA	56.4	**48.3**	74.7	*75.2*	*63.6*	*49.9*	*51.4*	*72.8*	**40.8**	*53.7*

quality pseudo labels and thus bringing much noise in the following MixUp step. ATDOC-NA improves the quality of pseudo labels and significantly boosts the performance when replacing the label guessing process in MixMatch. Benefited from a large amount of unlabeled data, ATDOC-NA outperforms MCC for SSL tasks on DomainNet-126 with a larger margin than that on Office-Home.

4.3. Model Analysis

We study the convergence of ATDOC-NA and the ramp-up of λ, and make comparisons with BNM in Fig. 2(a). Comparing both methods with or without the ramp-up, it is easy to verify the effectiveness of linear ramp-up. Since the pseudo labels or original classifier outputs in the early stage are not reliable enough, using a ramp-up to progressively increase the regularization weight is desirable for both ATDOC-NA and BNM. Besides, with the iteration number increasing, the accuracy of ATDOC-NA grows up and converges at last. Furthermore, we employ the t-SNE visualization [50] in Fig. 2(b) to show whether features from different domains are well aligned even without explicit domain alignment. Compared with source-only (ResNet-50) and pseudo-labeling, features from both domains learned by ATDOC-NA are semantically aligned and more favorable.

We further conduct ablation on Office-31 and VisDA-C for UDA and show average accuracy in Table 7. Comparing results in the first three rows, we find both weighting and sharpening strategies are effective. Besides, we study the neighborhood size m for ATDOC-NA and find a larger value of m can bring better performance. In particular, on the small Office-31 dataset, using $m = 1$ is quite risky and achieves worse results. Regarding another parameter λ, we discover $\lambda = 0.2$ is a suitable choice for both datasets. For the large-scale VisDA-C dataset, the learned pseudo labels are more reliable, so a large value of λ is beneficial. In addition, we investigate the effects of different parameters λ for ATDOC-NC. For a large-scale dataset, increasing λ is a better choice while decreasing λ is suitable for small-scale datasets. At last, we study the incorporation of source features in the memory bank and find it always degrades the performance, verifying the effectiveness of the target-oriented classifier.

5. Conclusion

We presented ATDOC, a new regularization approach to address the dataset shift for DA tasks. Despite the simplicity, extensive experiments demonstrated that ATDOC-NA outperforms both feature-level domain alignment methods and other regularization methods with consistent margins on UDA, SSDA, PDA, and even scarce-labeled SSL tasks. In the future, we would like to extend ATDOC to other challenging transfer tasks like universal DA [60, 77] and dense labeling tasks like semantic segmentation [67, 11].

Table 7. Ablation study.

Method/ Dataset	Office-31	VisDA-C
ATDOC-NA (***default***, $T = 0.5, m = 5, \lambda = 0.2$)	89.7 (–)	80.3 (–)
ATDOC-NA w/o weight $\tilde{q}_{i,\tilde{y}_i}$	89.4 (↓)	79.2 (↓)
ATDOC-NA w/ temperature $T = 1$	85.8 (↓)	65.6 (↓)
ATDOC-NA (w/ **source memory**)	89.4 (↓)	80.9 (↑)
ATDOC-NA w/ neighborhood size $m = 1$	84.7 (↓)	79.9 (↓)
ATDOC-NA w/ neighborhood size $m = 3$	87.9 (↓)	80.0 (↓)
ATDOC-NA w/ parameter $\lambda = 0.1$	90.2 (↑)	78.8 (↓)
ATDOC-NA w/ parameter $\lambda = 0.3$	89.2 (↓)	80.9 (↑)
ATDOC-NC (***default***, $\lambda = 0.1$)	89.2 (–)	74.6 (–)
ATDOC-NC w/ parameter $\lambda = 0.2$	88.6 (↓)	76.4 (↑)
ATDOC-NC w/ parameter $\lambda = 0.3$	88.3 (↓)	77.0 (↑)
ATDOC-NC (w/ **source memory**)	87.4 (↓)	66.6 (↓)

References

[1] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.*, 7(Nov):2399–2434, 2006. 3

[2] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Mach. Learn.*, 79(1-2):151–175, 2010. 1, 2

[3] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Proc. NeurIPS*, pages 5049–5059, 2019. 1, 3, 4, 5, 6, 7, 8

[4] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *Proc. NeurIPS*, pages 343–351, 2016. 2

[5] Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. Partial adversarial domain adaptation. In *Proc. ECCV*, pages 135–150, 2018. 2

[6] Zhangjie Cao, Kaichao You, Mingsheng Long, Jianmin Wang, and Qiang Yang. Learning to transfer examples for partial domain adaptation. In *Proc. CVPR*, pages 2985–2994, 2019. 7

[7] Woong-Gi Chang, Tackgeun You, Seonguk Seo, Suha Kwak, and Bohyung Han. Domain-specific batch normalization for unsupervised domain adaptation. In *Proc. CVPR*, pages 7354–7362, 2019. 2

[8] Minghao Chen, Hongyang Xue, and Deng Cai. Domain adaptation for semantic segmentation with maximum squares loss. In *Pro. ICCV*, pages 2090–2099, 2019. 1, 3, 4, 5

[9] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *Proc. ICML*, pages 1081–1090, 2019. 5, 6

[10] Yanbei Chen, Xiatian Zhu, and Shaogang Gong. Semi-supervised deep learning with memory. In *Proc. ECCV*, pages 268–283, 2018. 3

[11] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proc. CVPR*, pages 1791–1800, 2019. 8

[12] Zhihong Chen, Chao Chen, Zhaowei Cheng, Boyuan Jiang, Ke Fang, and Xinyu Jin. Selective transfer with reinforced transfer network for partial domain adaptation. In *Proc. CVPR*, pages 12706–12714, 2020. 7

[13] Jaehoon Choi, Minki Jeong, Taekyung Kim, and Changick Kim. Pseudo-labeling curriculum for unsupervised domain adaptation. In *Proc. BMVC*, 2019. 3

[14] Safa Cicek and Stefano Soatto. Unsupervised domain adaptation via regularized conditional alignment. In *Proc. ICCV*, pages 1416–1425, 2019. 1, 2

[15] Gabriela Csurka. A comprehensive survey on domain adaptation for visual applications. In *Domain adaptation in computer vision applications*, pages 1–35. Springer, 2017. 2

[16] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Liang Li, Qingming Huang, and Qi Tian. Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In *Proc. CVPR*, pages 3941–3950, 2020. 1, 3, 4, 5, 6, 7, 8

[17] Zhijie Deng, Yucen Luo, and Jun Zhu. Cluster alignment with a teacher for unsupervised domain adaptation. In *Proc. ICCV*, pages 9944–9953, 2019. 3

[18] Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *Proc. ICLR*, 2018. 1

[19] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proc. ICML*, pages 1180–1189, 2015. 2, 3

[20] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, 2016. 1

[21] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Proc. CVPR*, pages 2066–2073, 2012. 1

[22] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Proc. NeurIPS*, pages 529–536, 2005. 2, 4, 5, 6, 7, 8

[23] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J Smola. A kernel method for the two-sample-problem. In *Proc. NeurIPS*, pages 513–520, 2007. 2

[24] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *Proc. ICML*, pages 1321–1330, 2017. 4

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, pages 770–778, 2016. 5, 6, 7, 8

[26] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *Proc. ICML*, pages 1989–1998, 2018. 3

[27] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016. 1

[28] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proc. CVPR*, pages 5070–5079, 2019. 3

[29] Pin Jiang, Aming Wu, Yahong Han, Yunfeng Shao, Meiyu Qi, and Bingshuai Li. Bidirectional adversarial training for semi-supervised domain adaptation. In *Proc. IJCAI*, pages 934–940, 2020. 1, 5, 7

[30] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *Proc. ECCV*, pages 464–480, 2020. 1, 3, 4, 5, 6, 7, 8

[31] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proc. CVPR*, pages 4893–4902, 2019. 1

[32] Taekyung Kim and Changick Kim. Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation. In *Proc. ECCV*, pages 591–607, 2020. 5, 7

[33] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proc. ICLR*, 2017. 3

[34] Wouter Marco Kouw and Marco Loog. A review of domain adaptation without target labels. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 1–1, 2019. 2

[35] Vinod Kumar Kurmi, Shanu Kumar, and Vinay P Namboodiri. Attending to discriminative certainty for domain adaptation. In *Proc. CVPR*, pages 491–500, 2019. 5, 6

[36] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *Proc. ICLR*, 2016. 3

[37] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *Proc. CVPR*, pages 10285–10295, 2019. 1

[38] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, 2013. 1, 3, 4, 5

[39] Seungmin Lee, Dongwan Kim, Namil Kim, and Seong-Gyun Jeong. Drop to adapt: Learning discriminative features for unsupervised domain adaptation. In *Proc. ICCV*, pages 91–100, 2019. 5, 6, 7

[40] Da Li and Timothy Hospedale. Online meta-learning for multi-source and semi-supervised domain adaptation. In *Proc. ECCV*, pages 382–403, 2020. 1, 5, 7

[41] Jingjing Li, Erpeng Chen, Ding Zhengming, Lei Zhu, Ke Lu, and Heng Tao Shen. Maximum density divergence for domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 1–1, 2020. 3, 5, 7

[42] Shuang Li, Chi Harold Liu, Qiuxia Lin, Binhui Xie, Zhengming Ding, Gao Huang, and Jian Tang. Domain conditioned adaptation network. In *Proc. AAAI*, pages 11386–11393, 2020. 5, 6

[43] Jian Liang, Ran He, Zhenan Sun, and Tieniu Tan. Distant supervised centroid shift: A simple and efficient approach to visual domain adaptation. In *Proc. CVPR*, pages 2975–2984, 2019. 4

[44] Jian Liang, Ran He, Zhenan Sun, and Tieniu Tan. Exploring uncertainty in pseudo-label guided unsupervised domain adaptation. *Pattern Recognit.*, 96:106996, 2019. 5

[45] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *Proc. ICML*, pages 6028–6039, 2020. 2, 4, 5, 6, 7

[46] Jian Liang, Yunbo Wang, Dapeng Hu, Ran He, and Jiashi Feng. A balanced and uncertainty-aware approach for partial domain adaptation. In *Proc. ECCV*, pages 123–140, 2020. 7

[47] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *Proc. ICML*, pages 97–105, 2015. 1, 2

[48] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Proc. NeurIPS*, pages 1640–1650, 2018. 1, 2, 3, 4, 5, 6

[49] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Proc. NeurIPS*, pages 136–144, 2016. 2

[50] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *J. Mach. Learn. Res.*, 9(Nov):2579–2605, 2008. 8

[51] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(8):1979–1993, 2018. 3

[52] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *Proc. CVPR*, pages 3764–3773, 2020. 4

[53] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proc. ICCV*, pages 1406–1415, 2019. 5

[54] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017. 1, 5

[55] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset shift in machine learning*. 2009. 1

[56] Artem Rozantsev, Mathieu Salzmann, and Pascal Fua. Beyond sharing weights for deep domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(4):801–814, 2018. 2

[57] Danila Rukhovich and Danil Galeev. Mixmatch domain adaptaion: Prize-winning solution for both tracks of visda 2019 challenge. *arXiv preprint arXiv:1910.03903*, 2019. 1, 5

[58] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Proc. ECCV*, pages 213–226, 2010. 5

[59] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *Proc. ICCV*, pages 8050–8058, 2019. 1, 5, 7

[60] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, and Kate Saenko. Universal domain adaptation through self supervision. In *Proc. NeruIPS*, 2020. 3, 8

[61] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *Proc. CVPR*, pages 8503–8512, 2018. 3

[62] Weiwei Shi, Yihong Gong, Chris Ding, Xiaoyu Ma, Zhiheng Tao, and Nanning Zheng. Transductive semi-supervised deep learning using min-max features. In *Proc. ECCV*, pages 299–315, 2018. 3

[63] Rui Shu, Hung Bui, Hirokazu Narui, and Stefano Ermon. A dirt-t approach to unsupervised domain adaptation. In *Proc. ICLR*, 2018. 2

[64] Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *Proc. NeurIPS*, pages 2440–2448, 2015. 3

[65] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proc. CVPR*, pages 2818–2826, 2016. 5

[66] Tatiana Tommasi, Martina Lanzi, Paolo Russo, and Barbara Caputo. Learning the roots of visual domain shift. In *Proc. ECCV*, pages 475–482, 2016. 1

[67] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proc. CVPR*, pages 7472–7481, 2018. 1, 8

[68] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proc. CVPR*, pages 7167–7176, 2017. 2, 3

[69] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014. 2

[70] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Mach. Learn.*, 109(2):373–440, 2020. 2

[71] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proc. CVPR*, pages 5018–5027, 2017. 5

[72] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *Proc. ECCV*, pages 499–515, 2016. 2, 3

[73] Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation. *ACM Trans. Intell. Syst. Technol.*, 11(5), 2020. 2

[74] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proc. CVPR*, pages 3733–3742, 2018. 3

[75] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *Proc. ICML*, pages 5423–5432, 2018. 2, 3

[76] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *Proc. ICCV*, pages 1426–1435, 2019. 5, 6, 7

[77] Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Universal domain adaptation. In *Proc. CVPR*, pages 2720–2729, 2019. 8

[78] Erheng Zhong, Wei Fan, Qiang Yang, Olivier Verscheure, and Jiangtao Ren. Cross validation framework to choose amongst models and datasets for transfer learning. In *Proc. ECML-PKDD*, pages 547–562. Springer, 2010. 5

[79] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *Proc. CVPR*, pages 598–607, 2019. 3

[80] Xiaojin Zhu. Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2005. 2, 3

[81] Xiaojin Zhu and Zoubin Ghahramani. Learning from labels and unlabeled data with label propagation. 2002. 3

[82] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *Proc. ICCV*, pages 5982–5991, 2019. 3, 5, 6, 7

[83] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proc. ECCV*, pages 289–305, 2018. 3