Dual Moving Average Pseudo-Labeling for Source-Free Inductive Domain Adaptation

Hao Yan[1]
haoyan6@cmail.carleton.ca
Yuhong Guo[1,2]
yuhong.guo@carleton.ca

[1] Carleton University
Ottawa, Canada
[2] Canada CIFAR AI Chair
Amii, Canada

Abstract

Unsupervised domain adaptation reduces the reliance on data annotation in deep learning by adapting knowledge from a source to a target domain. For privacy and efficiency concerns, source-free domain adaptation extends unsupervised domain adaptation by adapting a pre-trained source model to an unlabeled target domain without accessing the source data. However, most existing source-free domain adaptation methods to date focus on the transductive setting, where the target training set is also the testing set. In this paper, we address source-free domain adaptation in the more realistic inductive setting, where the target training and testing sets are mutually exclusive. We propose a new semi-supervised fine-tuning method named Dual Moving Average Pseudo-Labeling (DMAPL) for source-free inductive domain adaptation. We first split the unlabeled training set in the target domain into a pseudo-labeled confident subset and an unlabeled less-confident subset according to the prediction confidence scores from the pre-trained source model. Then we propose a soft-label moving-average updating strategy for the unlabeled subset based on a moving-average prototypical classifier, which gradually adapts the source model towards the target domain. Experiments show that our proposed method achieves state-of-the-art performance and outperforms previous methods by large margins.

1 Introduction

Training deep models requires large scale datasets with accurate annotations. Data annotation however can be difficult in many real-world domains. Unsupervised domain adaptation (UDA) aims at reducing the dependence on data annotation in a target domain with the help of another already labeled source dataset. Existing UDA methods have attempted to learn a good prediction model for the target domain from both the labeled data in the source domain and the unlabeled data in the target domain by bridging the domain divergence gap through adversarial learning [6, 22], metric minimization [16, 30], or regularized semi-supervised learning [2, 12, 26]. However, in many domains that involve personal or commercial data, organizations usually prefer to release a pre-trained model instead of sharing a labeled source dataset due to data privacy concern. Moreover, storing and transferring large scale source dataset are way less efficient than working with a pre-trained source model. For such privacy and efficiency reasons, source-free domain adaptation (SFDA), which aims at learning a target model based on the pre-trained source model and the unlabeled target data, has recently become an emerging topic. Most existing SFDA methods have adopted a semi-supervised fine-tuning framework, where unlabeled target data are used to fine-tune the pre-trained source model. They exploit the unlabeled data in the target domain by either assigning pseudo-labels to the unlabeled target data [20, 36], or utilizing regularization terms such as mutual information maximization [20] and contrastive learning [10] to learn from the unlabeled target distribution. Some other works have also attempted to generate labeled target data or surrogate source data from the pre-trained source model by using generative models [19] or optimization strategies [35]. Nevertheless, these existing SFDA methods have all focused on a transductive setting, where the unlabeled training set in the target domain is also used as the testing set, and hence a method that overfits the unlabeled target training set can demonstrate good performance but has poor generalizability.

In this paper, we consider a more realistic source-free *inductive* domain adaptation setting, where the unlabeled target training set and the testing set are mutually exclusive. This setting aims to evaluate SFDA methods in terms of their generalization ability on unseen test data, which is important for real-world system deployment. To tackle this inductive SFDA problem, we propose a new semi-supervised fine-tuning method named Dual Moving Average Pseudo-Labeling (DMAPL). In the proposed method, to prevent over adaptation to the unlabeled target training data, we split the unlabeled training set in the target domain into two subsets based on the predictions made by the pre-trained source model: a confident subset of instances with high confident label prediction scores, and a less confident subset with low confident label prediction scores. As the high confident subset is better aligned with the source model, we use this subset with the predicted pseudo-labels as a fixed labeled subset to preserve the generalizable prediction properties of the pre-trained source model and alleviate the potential overfitting to the unlabeled target training data. The less confident subset is then used as an unlabeled subset for semi-supervised model adaptation. We design a soft-label updating strategy to gradually update the soft-labels for the unlabeled subset in a moving average manner using a moving-average based prototypical classifier. The soft-labels are used as the pseudo-labels of the unlabeled subset to fine-tune the prediction model.

To evaluate the proposed method, we conduct source-free inductive domain adaptation experiments on two large domain adaptation benchmark datasets. Our proposed method achieves state-of-the-art performance and outperforms the existing SFDA methods by large margins. The experiments also confirm that the relative performance of some existing UDA and SFDA methods is very different in this inductive domain adaptation setting from the previous reported performance in the transductive domain adaptation setting. This may inspire future works to investigate domain adaptation in the more realistic inductive setting.

2 Related Works

Unsupervised Domain Adaptation. Unsupervised domain adaptation (UDA) aims at learning a target model given labeled source data and unlabeled target data under cross-domain shift in data distribution. Existing UDA methods can be roughly divided into two categories, alignment-based and regularization-based methods. Alignment-based methods try to reduce domain discrepancy at different levels with various alignment techniques based on the theoretical analysis in [1]. These alignment techniques include maximum mean discrepancy (MMD) [30], adversarial alignment [6], moment matching [3], and optimal transport [16]. Alignment can be performed at feature-level [22], input-level [8] and output-level [29]. Regularization-based approaches usually treat the UDA problem as a semi-supervised learning problem, while adopting pseudo-labeling [17] to assign pseudo-labels to unlabeled target data for self-training [2, 21, 39]. Prediction smoothness losses based on instances and model-weight perturbations [26, 27] have been used as model regularization terms for unsupervised domain adaptation. Some other regularization terms, such as entropy minimization [27, 31], nuclear-norm maximization [4], and prediction consistency loss [5], have also been used to boost target model performance.

Source-Free Domain Adaptation. Source-free domain adaptation (SFDA) aims at learning a prediction model in the target domain given the pre-trained source model and the unlabeled data from the target domain. Existing SFDA methods are mostly based on semi-supervised fine-tuning, which uses different tricks to fine-tune the pre-trained source model with unlabeled target domain data. These methods can be roughly categorized into three groups: self-training based methods, data generation based methods, and regularization based methods. Self-training based methods assign pseudo-labels to the target domain data via various techniques including feature clustering [20, 25, 36], naive pseudo-labeling with instance weighting [13, 18], self-labeling [34], and learning with noisy label [38]. Data generation based methods generate either labeled data for the source distribution [9, 35] or labeled target domain data [15, 19] to facilitate domain adaptation. Regularization-based methods utilize regularization terms to help target model fine-tuning by exploring intrinsic characteristics in the target distribution. This includes mutual information maximization [20], contrastive feature learning [10, 33], instance and feature level mix-up [14]. Some methods also adopt additional model components during source model training, such as weight normalization [20] and domain attention [37]. However, all the existing SFDA methods address the transductive setting, where the unlabeled training set from the target domain is also the testing set. In this setting, a method that overfits the unlabeled target training set can demonstrate good performance but is not able to generalize well on unseen test data. In this paper, we consider an inductive setting, where the target training and testing sets are mutually exclusive, aiming to develop a more realistic experimental setup for SFDA.

3 Proposed Method

This paper addresses the source-free inductive domain adaptation problem, where a pre-trained source model f_S and an unlabeled target training set $\mathcal{X}_T$ are given. The goal is to train a target model f that can generalize well on a target testing set that is unseen during training. To tackle this problem, we propose a new semi-supervised fine-tuning approach to adapt the source model with the unlabeled target training data. To prevent over adaptation, we split the unlabeled target training data into a *pseudo-labeled subset* with high prediction confidence and an *unlabeled subset* with less confident predictions based on the pre-trained source model. The pseudo-labeled subset acts as trusty supervision for the target model fine-tuning. As the pseudo-labels of the unlabeled subset are much more noisy, we propose a soft-label moving-average updating method based on a moving-averaged prototypical classifier. The proposed method is illustrated in Figure 1 and elaborated in the following subsections.

3.1 Target Training Data Splitting

Learning with pseudo-labels is a classical semi-supervised learning strategy, where the predicted labels by the current model are used as pseudo-labels for the unlabeled instances for

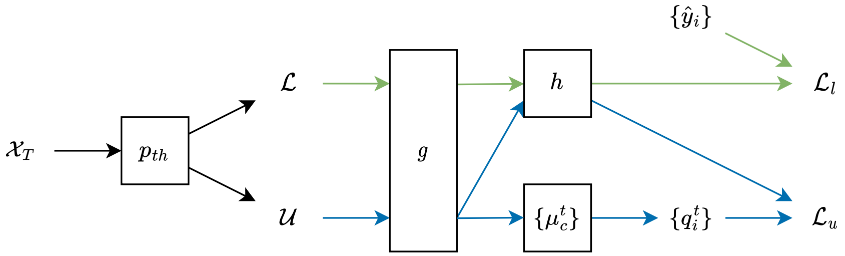

Figure 1: Illustration of the proposed method. The target training data are first split into a pseudo-labeled confident subset ($\mathcal{L}$) and a less-confident unlabeled subset ($\mathcal{U}$) based on the pre-trained source model. The pseudo-labeled confident subset acts as trusty supervision to prevent over adaptation. The unlabeled subset is gradually updated to fine-tune the prediction model ($f = h \circ g$) through dual moving average update.

further model update/fine-tuning. These pseudo-labels are usually noisy if the current model is not trained well, which will cause error accumulation during the iterative training process. In the source-free inductive domain adaptation setting, such gradual pseudo-labeling updates could make the model overfit the unlabeled target training data and hurt its generalization ability. On the other hand, pseudo-labeling with threshold only selects instances with highly confident predictions and ignores those with less confident predictions, which could lead to a loss of valuable data distribution information.

In view of these issues, we propose to split the unlabeled target training data $\mathcal{X}_T$ into a confident subset $\mathcal{L}$ and a less-confident subset $\mathcal{U}$ based on the pre-trained source model and then exploit them in separate ways. Specifically, for each unlabeled target training instance $\mathbf{x}_i \in \mathcal{X}_T$, we use $p(y|\mathbf{x}_i;\theta_{f_S})$ to denote the predicted probability of $\mathbf{x_i}$ belonging to the y-th class by the pre-trained source model f_S. By defining a prediction threshold p_{th}, we then select the instances with maximum prediction probability exceeding the threshold p_{th} to form the confident subset $\mathcal{L}$ and others form the less-confident subset $\mathcal{U}$, i.e.

$$\mathcal{L} = \left\{ \mathbf{x}_i \in \mathcal{X}_T \,|\, \max_y p(y|\mathbf{x}_i;\theta_{f_S}) \geq p_{th} \right\}, \qquad \mathcal{U} = \mathcal{X}_T \backslash \mathcal{L}. \tag{1}$$

Hence for each instance in the confident subset $\mathcal{L}$, its pseudo-label can be assigned with high probability (larger then p_{th}) based on the source model predictions as follows:

$$\hat{y}_i = \arg\max_y \, p(y|\mathbf{x}_i;\theta_{f_S}), \quad \mathbf{x}_i \in \mathcal{L}. \tag{2}$$

All the instances in $\mathcal{L}$ and their corresponding pseudo-labels can then form a pseudo-labeled subset $\{(\mathbf{x}_i, \hat{y}_i) | \mathbf{x}_i \in \mathcal{L}\}$. If the prediction threshold p_{th} is high enough, the high confident subset $\mathcal{L}$ will be well aligned with the pre-trained source model. We hence propose to use this subset $\mathcal{L}$ with the predicted pseudo-labels as a fixed *labeled subset* and use the less confident subset $\mathcal{U}$ as an *unlabeled subset* for semi-supervised model fine-tuning. The unlabeled subset will be used to fine-tune and adapt the source model to the target domain, while the labeled subset will be used as trusty supervision to preserve the generalizable prediction properties of the source model and prevent over adaptation to the unlabeled target subset during the semi-supervised fine-tuning process. Specifically, we consider the following supervised cross-entropy loss on the labeled subset $\mathcal{L}$:

$$\mathcal{L}_l = \mathbb{E}_{\mathbf{x}_i \in \mathcal{L}} \left[-\log(p(\hat{y}_i | \mathbf{x}_i; \theta_f)) \right], \tag{3}$$

where θ_f denotes the parameters of target model f, which is initialized from the source model f_S at the very beginning.

3.2 Dual Moving Average based Model Fine-Tuning

By using $\mathcal{L}$ as a fixed labeled subset, we conduct semi-supervised model fine-tuning with soft pseudo-labeling updates on the unlabeled subset $\mathcal{U}$. Inspired by the partial label learning work [32], we introduce a dual moving average based soft-label updating scheme to assign soft-labels to the instances in the unlabeled subset $\mathcal{U}$. Typical deep prediction models are composed of a convolutional encoder and a linear or MLP classifier head. We hence denote the target model f as a composition of a feature encoder g and a classifier h, i.e. $f = h \circ g$. For each instance $\mathbf{x}_i$, $g(\mathbf{x}_i)$ denotes its feature vector extracted by the encoder g. Based on the extracted feature vectors, we introduce a prototypical classifier with centroids denoted as $\{\mu_c\}_{c=1}^{C}$, where C is the number of classes. Each centroid μ_c denotes the prototype corresponding to the c-th class and it is updated with the feature vectors of the instances belonging to the c-th class in every batch. To calculate the centroids, we first normalize each feature vector $g(\mathbf{x}_i)$ by its L2-norm, resulting in a normalized vector $\mathbf{z}_i = g(\mathbf{x}_i)/\|g(\mathbf{x}_i)\|_2$. Considering the current mini-batch with labeled instances X_l sampled from $\mathcal{L}$ and unlabeled instances X_u sampled from $\mathcal{U}$, the feature mean of the c-th class in the current iteration t can be calculated as,

$$\mathbf{v}_c^t = \frac{\sum_{\mathbf{x}_i \in (X_l \cup X_u)} \mathbb{1}(\bar{y}_i = c) \cdot \mathbf{z}_i}{\sum_{\mathbf{x}_i \in (X_l \cup X_u)} \mathbb{1}(\bar{y}_i = c)}, \tag{4}$$

where $\mathbb{1}(\cdot)$ denotes an indicator function; and $\bar{y}_i$ denotes the pseudo-label for instance $\mathbf{x}_i$. For instances from the labeled subset, we always use the pre-assigned pseudo-labels by the initial source model, i.e., $\bar{y}_i = \hat{y}_i$ for $\mathbf{x}_i \in \mathcal{L}$. For instances from unlabeled subset, we use the pseudo-labels predicted by the current model, i.e., $\bar{y}_i = \arg\max_y p(y|\mathbf{x}_i; \theta_f)$for $\mathbf{x}_i \in \mathcal{U}$. We then calculate the centroid μ_c^t for the current iteration t as the weighted average of the centroid μ_c^{t-1} from the previous iteration and the feature mean $\mathbf{v}_c^t$ in he current iteration:

$$\mu_c^t = \text{Normalize}(\alpha \mu_c^{t-1} + (1-\alpha)\mathbf{v}_c^t), \tag{5}$$

where Normalize$(\cdot)$ denotes the L2-normalization function and $\alpha \in (0,1)$ is a moving average coefficient hyperparameter. Starting with the initial centroid $\mu_c^0 = \mathbf{0}$, the centroid vector will be updated by gradually discounting the previous vector with a factor α and adding the current feature mean vector $\mathbf{v}_c^t$ with a weight $(1-\alpha)$ in each iteration.

The prototypical classifier assigns a new one-hot label vector $\tilde{\mathbf{y}}_i^t$ to each unlabeled instance $\mathbf{x}_i \in X_u$ by placing 1 in the j-th entry of the label vector such that the maximum inner-product value is produced between the normalized feature vector $\mathbf{z}_i$ and the centroid vector μ_j^t among all centroids $\{\mu_1^t, \cdots, \mu_C^t\}$, such that

$$(\tilde{\mathbf{y}}_i^t)_j = \begin{cases} 1, & j = \arg\max_{c \in \{1,\cdots,C\}} \mathbf{z}_i^\top \mu_c^t, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

where $(\tilde{\mathbf{y}}_i^t)_j$ denotes the j-th element of the vector $\tilde{\mathbf{y}}_i^t$. Note as both vectors are L2-normalized, the inner-product above actually measures the cosine-similarity between the feature vector and the centroid. This newly assigned pseudo-label vector however will not be used directly to train the target model as it is still very noisy. Instead, we use the pseudo-label vectors produced by the prototypical classifier to update the soft-labels of the unlabeled subset gradually in a moving average manner. We denote the soft-label vector for an unlabeled instance $\mathbf{x}_i$ in the current iteration t as $\mathbf{q}_i^t$, while $\mathbf{q}_i^0 = \mathbf{0}$. It is updated as the weighted average of the soft-label vector $\mathbf{q}_i^{t-1}$ from the previous iteration and the one-hot pseudo-label vector $\tilde{\mathbf{y}}_i^t$ from the current prototypical classifier:

$$\mathbf{q}_i^t = \beta \mathbf{q}_i^{t-1} + (1-\beta)\tilde{\mathbf{y}}_i^t, \tag{7}$$

where $\beta \in (0,1)$ is another coefficient hyperparameter. As the one-hot labels from the prototypical classifier in the early iterations typically would be more noisy, the moving average update in Eq.(7) could gradually discount the previous predictions with a discount factor β. Different from the work [32], we do not set the initial soft-label vectors as uniform vectors as that will introduce more noise into the soft-labels—all labels except one for each instance would be wrong labels. This makes the soft-label vector not a probability vector as all the elements do not add up to 1. It is easy to show that the L1-norm of $\mathbf{q}_i^t$ will always be smaller than 1 within finite iterations, while approaching 1 when $t \rightarrow \infty$:

$$|\mathbf{q}_i^t|_1 = \beta |\mathbf{q}_i^{t-1}|_1 + (1-\beta)|\tilde{\mathbf{y}}_i^t| = (1-\beta)(1+\beta+\beta^2+\cdots+\beta^{t-1}) = 1-\beta^t \tag{8}$$

The soft-label vectors for the instances in the unlabeled subset $\mathcal{U}$ are then further used to fine-tune the target model f by minimizing the following *soft cross-entropy loss* in the t-th iteration:

$$\mathcal{L}_u = \mathbb{E}_{\mathbf{x}_i \in \mathcal{U}} \left[\sum_y -(\mathbf{q}_i^t)_y \log p(y|\mathbf{x}_i;\theta_f) \right] \tag{9}$$

where $(\mathbf{q}_i^t)_y$ denotes the y-th element of the vector $\mathbf{q}_i^t$ and it indicates the weight of the cross-entropy loss computed using label y. If the prototypical classifier keeps assigning the instance $\mathbf{x}_i$ to the y-th class across iterations, $(\mathbf{q}_i^t)_y$ will gradually increase to enforce the importance of the pseudo-labeled pair $(\mathbf{x}_i, y)$ for model fine-tuning. Meanwhile, using soft-label vectors allows the system to consider different label assignment options in a weighted manner, which can alleviate dramatic oscillations from hard label assignments and produce stable model fine-tuning.

By taking both subsets $\mathcal{L}$ and $\mathcal{U}$ into consideration, the overall loss minimization for the proposed semi-supervised fine-tuning method is shown as follows,

$$\min_{\theta_f} \mathcal{L}_u + \lambda \mathcal{L}_l, \tag{10}$$

where λ is the trade-off parameter. Here we associate the trade-off parameter with the labeled cross-entropy loss instead of the soft-labeled cross-entropy loss to avoid interactive influence between parameters β and λ.

4 Experiments

In this section, we conduct experiments to evaluate the proposed source-free inductive domain adaptation method and perform ablation study to investigate the separate contributions from the two components of the proposed method. As most exiting datasets for domain adaptation do not split training and testing sets, we select two large domain adaptation datasets for experiments such that the training and testing sets will both have sufficient data.

4.1 Experimental Setting

Datasets. We evaluate our proposed method on two large domain adaptation datasets, *DomainNet* [24] and *VisDA2017* [23]. *DomainNet* is a large-scale classification dataset for domain adaptation. It has already split the data in each domain into training and testing sets. The original *DomainNet* dataset has about 0.6 millions images distributed among 345 categories from 6 domains. Here we choose 4 domains with decent oracle performance to use. The chosen domains are clipart (c), painting (p), real (r) and sketch (s). There are totally 12 domain adaptation tasks among them. *VisDA2017* is a large-scale synthetic to real domain adaptive classification dataset. There are two domains, a synthetic domain with 152,397 simulation images and a real domain with 55,388 real-word images. Both domain contain images from 12 categories and we conduct experiments on the synthetic to real domain adaptation task. As the original *VisDA2017* dataset does not split training and testing sets, we split each domain into training and testing sets with a splitting ratio of $8:2$. To preserve the original class distribution, we split the data from each class separately and randomly according to the same ratio. We name this split *VisDA2017* dataset as *VisDA2017Split*.

Implementation details. For fair comparisons, we follow the previous domain adaptation methods and employ the pre-trained ResNet-101 [7] as the backbone module for both *DomainNet* and *VisDA2017Split* datasets. For the same reason, we replace the final linear layer of the ResNet-101 model with a bottleneck module and a linear layer. The bottleneck module is composed of one linear layer and a 1-dimensional batch normalization layer followed by a ReLU layer. It is used to reduce the feature dimension from 2048 to 256. For all model training and fine-tuning, we use SGD optimizer with momentum set as 0.9 and weight decay set as 10^{-3}. The learning rate is scheduled as a cosine decaying function $\eta_i = \eta_1 + 0.5(\eta_0 - \eta_1)(1 + \cos(t\pi/N))$, where η_0, η_1, t, N are the initial learning rate, stopping learning rate, iteration step and total number of iterations respectively. This decaying function decreases the learning rate from η_0 to η_1 slowly at the starting and stopping iterations, and rapidly at the intermediate iterations. We set $\eta_0 = 10^{-2}$ and $\eta_1 = 10^{-3}$ for the parameters of bottleneck and linear layers. As the backbone is pre-trained on ImageNet, we set $\eta_0 = 10^{-3}$ and $\eta_1 = 10^{-4}$ for the parameters of the backbone module. For source model pre-training, we train the model for 20 epochs on the training set and use the testing set to find the best source model by early-stopping. For the target model fine-tuning, we set the prediction threshold $p_{th} = 0.9$ for both datasets to select high confident pseudo-labels. For soft-label updating, we set two coefficient parameters as $\alpha = 0.9$ and $\beta = 0.9$. In the overall objective, the trade-off parameter λ is set to 1 for *DomainNet* dataset and 10^{-2} for *VisDA2017Split* dataset. Our code is available online[1].

4.2 Results of Source-Free Inductive Domain Adaptation

To evaluate the proposed Dual Moving Average Pseudo-Labeling (DMAPL) method, we conduct source-free inductive domain adaptation experiments on the two datasets, *DomainNet* and *VisDA2017Split*. The results are reported in Table 1 and Table 2. With the pre-trained source model, we first evaluate it on the unlabeled target testing set and produce the source-only result on each task, which demonstrates the performance of the source model without any adaptation process and is used as a comparison baseline. We also report the

[1] https://github.com/cnyanhao/dmapl

Table 1: Test accuracy (%) on DomainNet dataset (ResNet-101). SF means source-free.

Methods	SF	c→p	c→r	c→s	p→c	p→r	p→s	r→c	r→p	r→s	s→c	s→p	s→r	Avg.
ResNet-101 [7]	-	37.9	53.4	44.2	44.1	57.0	38.6	50.9	48.8	37.7	52.8	37.3	47.6	45.9
AdaMatch [2]	✗	45.3	56.0	60.2	35.3	47.6	42.9	46.5	48.1	49.1	46.5	41.0	42.4	46.7
MCC [12]	✗	37.7	55.7	42.6	45.4	59.8	39.9	54.4	53.1	37.0	58.1	46.3	56.2	48.9
CDAN [11, 22]	✗	40.4	56.8	46.1	45.1	58.4	40.5	55.6	53.6	43.0	57.2	46.4	55.7	49.9
CDAN+SDAT [26]	✗	41.5	57.5	47.2	47.5	58.0	41.8	56.7	53.6	43.9	58.7	48.1	57.1	51.0
SHOT [20]	✓	45.6	63.4	49.1	35.1	64.1	21.0	57.1	51.1	44.0	61.2	47.6	62.0	48.4
SSFT-SSD [35]	✓	41.9	57.5	46.5	47.6	59.6	42.6	55.4	51.9	42.0	58.4	45.2	55.7	50.4
DMAPL (Ours)	✓	**46.0**	**63.7**	**49.1**	**53.2**	**64.2**	**46.0**	**61.6**	**55.4**	**47.8**	**64.1**	**50.3**	**63.5**	**55.4**
Oracle	-	71.1	83.4	70.0	78.4	83.4	70.0	78.4	71.1	70.0	78.4	71.1	83.4	75.7

Table 2: Test accuracy (%) on VisDA2017Split dataset (ResNet-101). SF means source-free.

Methods	SF	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Macro	Micro
ResNet-101 [7]	-	76.7	23.9	48.1	68.0	67.8	6.5	86.0	20.6	71.8	23.9	85.0	8.4	48.9	54.1
CDAN [11, 22]	✗	92.7	73.5	80.0	46.4	90.2	93.2	86.1	78.4	83.8	87.3	83.2	38.3	77.8	73.7
MCC [12]	✗	92.2	79.4	79.0	71.7	92.1	93.0	89.9	79.0	88.2	91.0	82.1	50.8	82.4	80.0
SHOT [20]	✓	77.7	**85.8**	80.2	54.2	90.2	63.4	**82.1**	73.5	88.9	80.5	83.1	54.8	76.2	73.8
SSFT-SSD [35]	✓	94.5	84.9	**80.9**	49.9	91.2	66.8	77.0	75.4	81.3	86.2	**89.4**	50.4	77.3	73.6
DMAPL (Ours)	✓	**95.6**	84.5	78.9	**58.7**	**92.4**	**96.6**	80.8	**82.5**	**90.3**	**88.6**	87.8	**59.1**	**83.0**	**79.1**
Oracle	-	98.2	94.7	89.5	88.0	98.7	96.4	93.6	92.8	98.0	96.5	93.4	72.6	92.7	91.5

target-supervised results as an *oracle* reference, where the model is trained on the labeled target training set and evaluated on the target testing set. To compare with the existing domain adaptation methods, when there are existing inductive results, we cite the results directly; e.g., the test results on the *DomainNet* dataset for CDAN [11, 22], MCC [12], SDAT [26] and AdaMatch [2]. Otherwise, we run the methods in the inductive setting to produce fair comparisons. As there are no reported SFDA results in the inductive setting, we selected two SFDA methods with online code, SHOT and SSFT-SSD, to compare with our proposed method in the same inductive setting. We produce the comparison results on the *VisDA2017Split* dataset in a similar way.

Table 1 reports the domain adaptation test accuracy results for the 12 domain adaptation tasks and the task average results on the *DomainNet* dataset. This table comprises two parts of results. The upper part includes the *source-only* results denoted as *ResNet-101* and the results of the several UDA methods that use both the labeled source data and unlabeled target data. The lower part reports the results of our proposed method (DMAPL) and two existing source-free domain adaptation methods—the results are produced in the inductive SFDA setting. The bottom line reports the target supervised results denoted as *oracle*. First, we can see our proposed method improves the testing accuracy over source-only by a large margin on every task, and on average a 9.5 percentage points (pp) gain can be observed. This is a remarkable improvement on this challenging dataset. Second, comparing with the several unsupervised domain adaptation methods, our method outperforms all of them even though the source-free setting is more restrictive than the UDA setting. This shows the effectiveness of the proposed semi-supervised fine-tuning framework on exploiting the unlabeled target training data. Similar results have been widely observed in the previous source-free domain adaptation works as well [18, 20]. Third, our method also outperforms both of the two SFDA comparison methods by large margins. Our method achieves the state-of-the-art performance on all the 12 tasks. Comparing with the oracle results, we can see there is still a large space for further improvement. In addition, there are also some other interesting observations. For example, the two UDA methods, MCC and AdaMatch, have been previously shown to perform better than CDAN in the transductive domain adaptation setting [2, 12], while here CDAN achieves better results than MCC and AdaMatch in the

Table 3: Ablation study: Average accuracy (%)

Ablation	Split target	DomainNet	VisDA2017 (Micro)	VisDA2017 (Macro)
Source-only	✗	45.9	54.1	48.9
Naive pseudo-labeling [17]	✗	47.8	61.2	55.7
Soft-label updating	✗	52.4	78.3	82.2
DMAPL	✓	55.3	79.1	83.0

inductive setting. The two source-free domain adaptation comparison methods, SHOT and SSFT-SSD, have been shown to outperform the source-only baseline by large margins in the transductive setting in the literature, while here they can only achieve small performance gains in the inductive setting. These observations might inspire future works to shift the focus of domain adaptation from the transductive setting to the inductive setting.

Table 2 reports the test accuracy results on the 12 categories, the macro accuracy and the micro accuracy for the synthetic-to-real domain adaptation task on the *VisDA2017Split* dataset. Here the macro accuracy is the test accuracy averaged over the 12 classes and the micro accuracy is the test accuracy averaged over all test instances. First, comparing with the source-only baseline, our proposed method improves the macro accuracy by 34.1 pp and improves the micro accuracy by 25 pp. Second, our proposed method also outperforms all the UDA methods, though UDA has the source data available. Finally, our method outperforms the two SFDA comparison methods by large margins in terms of both macro accuracy and micro accuracy under the inductive setting. Overall, the proposed method achieves state-of-the-art performance on 8 out of the 12 categories, as well as in terms of macro accuracy and micro accuracy. But still there is large improvement space for future works by comparing with the oracle results.

4.3 Ablation Study and Hyper-Parameters Analysis

Ablation study. We investigate the contributions of the two components of the proposed method: the target training data split and the dual moving average based soft pseudo-labeling. We compare the proposed method, DMAPL, with the Source-only baseline and two variants: *naive pseudo-labeling* and *soft-label updating*. *Naive pseudo-labeling* does not split the target training data, but assigns pseudo-labels to the unlabeled target training data at the beginning of each epoch and uses them for target model fine-tuning; *Soft-label updating* drops the target training data splitting step from the proposed method and uses all target training data for soft pseudo-label based fine-tuning. As shown in Table 3, comparing with the complete method DMAPL, *soft-labeling updating* without target data split degrades the performance on both datasets. This verifies the usage of the selected target training instances as fixed labeled data during model fine-tuning. Meanwhile, the proposed method DMAPL outperforms *naive pseudo-labeling* by large margins on both datasets. This shows the effectiveness of the proposed soft-label updating method.

Confidence threshold. The left figure of Figure 2 shows the confident subset splitting *Ratio* ($|\mathcal{L}|/|\mathcal{X}_T|$), accuracy of the pseudo-labels for the confident subset $\mathcal{L}$ (*PL accu*), and test set accuracy (*Trg acc*) values for different threshold (p_{th}) choices on the task p → c of *DomainNet*. We can observe that higher threshold makes the pseudo-labels for the high confident subset cleaner (higher *PL acc*) at the expense of less instances in the high confident subset (lower Ratio). However, similar testing accuracy values (*Trg acc*) can be observed with different threshold choices. The reason might be credited to the effectiveness of the

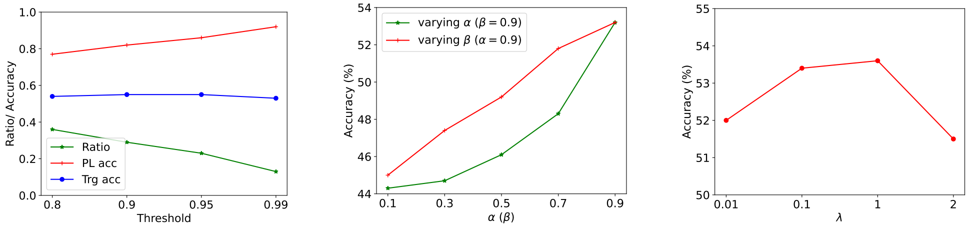

Figure 2: Hyper-parameters analysis. Experiments are conducted on task p → c of DomainNet dataset. Left: Confident subset splitting *Ratio*, accuracy of the pseudo-labels for the confident subset (*PL acc*) and test set accuracy (*Trg acc*) for different thresholds (p_{th}). Middle: Test set accuracy for different choices of coefficient parameters α and β. Right: Test set accuracy for different choices of the trade-off parameter λ.

proposed soft-label updating approach. This indicates that our proposed method is not very sensitive to the p_{th} value as long as it is reasonably large ($0.8 \leq p_{th} < 1$).

Coefficient parameters. The middle figure of Figure 2 illustrates the test accuracy values against different choices of the coefficient parameters α and β on the task p → c of *DomainNet*, where α and β control the updating degrees for centroid update and soft-label update respectively—lower values indicate larger updating degrees. It is obviously that higher values achieve better performance for both variables. This indicates that slower updates of the centroid vectors and soft-labels are more beneficial for the proposed method, due to better training stability. This coincides with results from other moving average cases [28] as well.

Trade-off parameter. The right figure of Figure 2 demonstrates the test set accuracy values for different choices of the trade-off parameter λ on the task p → c of *DomainNet*. We can observe that the proposed method is not very sensitive to this trade-off parameter within a reasonable range of values $[0.1, 1]$, while the optimal λ value is 1.

5 Conclusion

This paper addresses a newly proposed source-free inductive domain adaptation problem. We proposed a new semi-supervised fine-tuning method based on dual moving average pseudo-labeling. To prevent over adaptation of the source model to the unlabeled target training data, we proposed to split the unlabeled target training data into a pseudo-labeled subset with high prediction confidence and an unlabeled subset with less confident predictions based on the pre-trained source model. The model is fine-tuned in a semi-supervised manner by using the pseudo-labeled subset as trusty supervision, and using the unlabeled subset with soft-labels produced by dual moving average pseudo-labeling. The experimental results show that our method achieves state-of-the-art performance and outperforms previous methods by large margins for source-free inductive domain adaptation. The experiments also indicate that some previous domain adaptation methods that are effective in the transductive setting can achieve less performance gains in the inductive setting. This is expected to motivate future works in the more realistic inductive domain adaptation settings.

References

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

[2] David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alex Kurakin. Adamatch: A unified approach to semi-supervised learning and domain adaptation. In *ICLR*, 2022.

[3] Chao Chen, Zhihang Fu, Zhihong Chen, Sheng Jin, Zhaowei Cheng, Xinyu Jin, and Xian-Sheng Hua. Homm: Higher-order moment matching for unsupervised domain adaptation. In *AAAI*, 2020.

[4] 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 *CVPR*, 2020.

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

[6] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[8] 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 *ICML*, 2018.

[9] Yunzhong Hou and Liang Zheng. Visualizing adapted knowledge in domain transfer. In *CVPR*, 2021.

[10] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. In *NeurIPS*, 2021.

[11] Junguang Jiang, Chen, Baixu, Bo Fu, and Mingsheng Long. Transfer-learning-library. https://github.com/thuml/Transfer-Learning-Library, 2020.

[12] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *ECCV*, 2020.

[13] Jogendra Nath Kundu, Naveen Venkat, R Venkatesh Babu, et al. Universal source-free domain adaptation. In *CVPR*, 2020.

[14] Jogendra Nath Kundu, Akshay R Kulkarni, Suvaansh Bhambri, Deepesh Mehta, Shreyas Anand Kulkarni, Varun Jampani, and Venkatesh Babu Radhakrishnan. Balancing discriminability and transferability for source-free domain adaptation. In *ICML*, 2022.

[15] Vinod K Kurmi, Venkatesh K Subramanian, and Vinay P Namboodiri. Domain impression: A source data free domain adaptation method. In *WACV*, 2021.

[16] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *CVPR*, 2019.

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

[18] Jonghyun Lee, Dahuin Jung, Junho Yim, and Sungroh Yoon. Confidence score for source-free unsupervised domain adaptation. In *ICML*, 2022.

[19] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *CVPR*, 2020.

[20] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *ICML*, 2020.

[21] Hong Liu, Jianmin Wang, and Mingsheng Long. Cycle self-training for domain adaptation. In *NeurIPS*, 2021.

[22] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, 2018.

[23] 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.

[24] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, 2019.

[25] Zhen Qiu, Yifan Zhang, Hongbin Lin, Shuaicheng Niu, Yanxia Liu, Qing Du, and Mingkui Tan. Source-free domain adaptation via avatar prototype generation and adaptation. In *IJCAI*, 2021.

[26] Harsh Rangwani, Sumukh K Aithal, Mayank Mishra, Arihant Jain, and Venkatesh Babu Radhakrishnan. A closer look at smoothness in domain adversarial training. In *ICML*, 2022.

[27] Rui Shu, Hung Bui, Hirokazu Narui, and Stefano Ermon. A DIRT-T approach to unsupervised domain adaptation. In *ICLR*, 2018.

[28] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017.

[29] 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 *CVPR*, 2018.

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

[31] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019.

[32] Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen, and Junbo Zhao. Pico: Contrastive label disambiguation for partial label learning. In *ICLR*, 2022.

[33] Haifeng Xia, Handong Zhao, and Zhengming Ding. Adaptive adversarial network for source-free domain adaptation. In *ICCV*, 2021.

[34] Hao Yan, Yuhong Guo, and Chunsheng Yang. Augmented self-labeling for source-free unsupervised domain adaptation. In *NeurIPS Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.

[35] Hao Yan, Yuhong Guo, and Chunsheng Yang. Source-free unsupervised domain adaptation with surrogate data generation. In *BMVC*, 2021.

[36] Shiqi Yang, Joost van de Weijer, Luis Herranz, Shangling Jui, et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. In *NeurIPS*, 2021.

[37] Shiqi Yang, Yaxing Wang, Joost van de Weijer, Luis Herranz, and Shangling Jui. Generalized source-free domain adaptation. In *ICCV*, 2021.

[38] Haojian Zhang, Yabin Zhang, Kui Jia, and Lei Zhang. Unsupervised domain adaptation of black-box source models. In *BMVC*, 2021.

[39] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *ICCV*, 2019.