Research Article

Source-Free Semantic Regularization Learning for Semi-Supervised Domain Adaptation

Xinyang Huang[1]

1. School of Artificial Intelligence, Beijing University of Posts and Telecommunications, China

Semi-supervised domain adaptation (SSDA) has been extensively researched due to its ability to improve classification performance and generalization ability of models by using a small amount of labeled data on the target domain. However, existing methods cannot effectively adapt to the target domain due to difficulty in fully learning rich and complex target semantic information and relationships. In this paper, we propose a novel SSDA learning framework called semantic regularization learning (SERL), which captures the target semantic information from multiple perspectives of regularization learning to achieve adaptive fine-tuning of the source pre-trained model on the target domain. SERL includes three robust semantic regularization techniques. Firstly, semantic probability contrastive regularization (SPCR) helps the model learn more discriminative feature representations from a probabilistic perspective, using semantic information on the target domain to understand the similarities and differences between samples. Additionally, adaptive weights in SPCR can help the model learn the semantic distribution correctly through the probabilities of different samples. To further comprehensively understand the target semantic distribution, we introduce hard-sample mixup regularization (HMR), which uses easy samples as guidance to mine the latent target knowledge contained in hard samples, thereby learning more complete and complex target semantic knowledge. Finally, target prediction regularization (TPR) regularizes the target predictions of the model by maximizing the correlation between the current prediction and the past learned objective, thereby mitigating the misleading of semantic information caused by erroneous pseudo-labels. Extensive experiments on three benchmark datasets demonstrate that our SERL method achieves state-of-the-art performance.

 doi.org/10.32388/4T33D8

I. Introduction

In recent years, deep neural networks (DNN) have brought a series of breakthroughs in many computer vision tasks, such as image classification[1][2][3][4][5][6][7], semantic segmentation[8][9][10][11][12][13]. However, to achieve satisfactory results, the large number of sample labels required for deep neural network training is costly and time-consuming. Therefore, domain adaptation (DA)[14][15][16][17] is proposed by generalizing the knowledge learned from the source domain with rich labels to the target domain with no or few labels. Domain adaptation can be simply divided into unsupervised domain adaptation (UDA)[18][19][20][21][22][23][24][25] and semi-supervised domain adaptation (SSDA)[26][27][28][29][30][31][32][33][34][35][36] according to access to target labels during training. This paper focuses on SSDA, which performs significantly better than UDA when given a small number of labeled target samples. It can utilize a small number of labels on the target domain to expand semantic information and learn semantic knowledge of target samples of the same category to achieve domain alignment.

Due to its advantages of practical significance, SSDA has attracted increasing attention and has been widely studied. However, SSDA also has its specific challenges and issues. First, the training of the supervised model only uses a small number of labeled target samples. The model can only learn the extremely limited target domain knowledge and cannot generate a highly discriminative knowledge representation for the target domain[26][34]. At the same time, due to many labeled source samples, the feature representation learned by the model is biased toward the source domain[37]. To address these issues, existing methods[26][27][28][29][30][31][32][33][34][35] have proposed their solutions to address these challenges and have witnessed significant performance improvements. MCL[31] learns the consistency between samples, but it ignores the learning of target semantic information. ProML[33] utilizes target labels by constructing prototypes, but the semantic information contained in them is very limited due to the scarcity of labeled target samples. Due to the complexity of semantic information between target samples, the knowledge representation learned by existing methods still needs to be improved. This complex semantic information$i.e.$, category knowledge representation on the target domain can better bridge the distribution differences between domains, encouraging the model to generate domain-invariant but differentiated target features when adapting.

In this paper, we present a novel SSDA learning framework, named semantic regularization learning (SERL), which is proposed to tackle the challenges of the SSDA tasks. As shown in Figure 1, different

from the training paradigm of most existing SSDA methods, this paper considers a source-free scenario,$i.e.$, in which target domain adaptation is performed using the source domain pre-trained model[38]. Unlike UDA, SSDA can obtain a small amount of labeled data on the target domain, so it can better adapt to this source-free scenario. SERL provides regularization constraints from different perspectives by fully learning the target semantic information, which can enrich the understanding of the accurate distribution of the target domain and thus better learn the knowledge of the target domain, as shown in Figure 2.

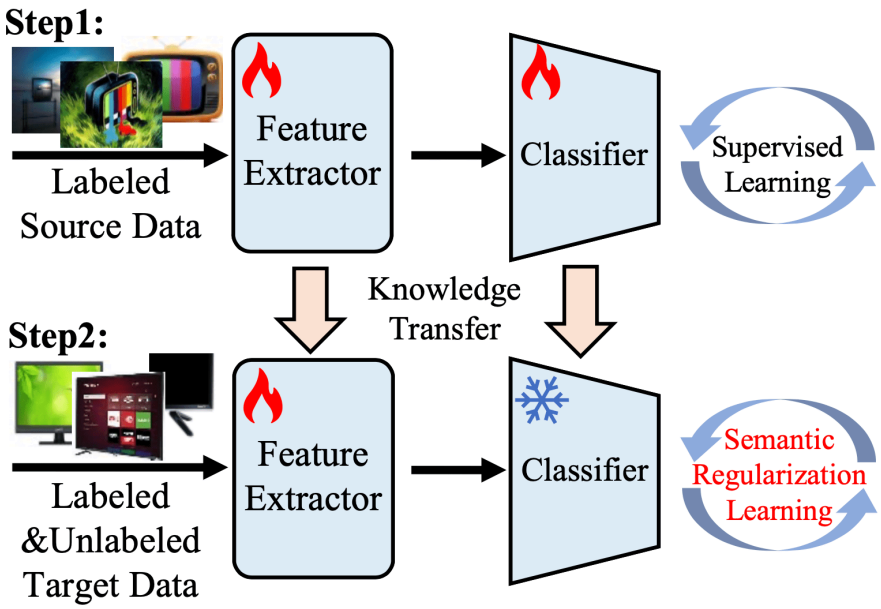

Figure 1. The learning scenario of our SERL framework. Different from the training paradigm of most existing SSDA methods, we adopt a source-free training strategy. The source model comprises a feature extractor and a classifier initialized on the source domain. We focus on improving the target domain adaptation stage of the model. In the target domain adaptation stage, SERL freezes the classifier module and fine-tunes the feature extractor module through semantic regularization learning.

doi.org/10.32388/4T33D8

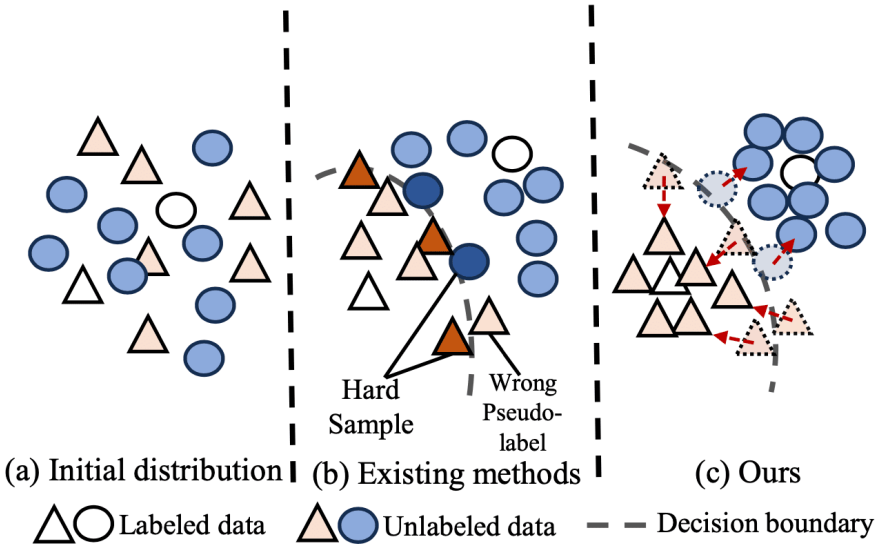

Figure 2. The motivation of our SERL. (a) Due to the scarcity of target semantic labels during training, most existing SSDA methods have shortcomings in target semantic learning, resulting in models only learning limited knowledge (e.g., only the relationships between samples) on the target domain. When more complex relationships exist on the target domain, such as hard and noisy samples, the model may perform poorly due to a lack of understanding of semantic information. (b) Our SERL utilizes the semantic information learned on the target domain from the perspective of semantic regularization to constrain the feature representation of the model further, thereby adapting to more complex target domain distributions.

Specifically, we propose semantic probability contrastive regularization (SPCR), which helps the model aggregate features of similar samples according to the distribution of target semantic information and keep features of heterogeneous samples away from each other. This method forces the model to learn more discriminative semantic knowledge on the target domain from the probability perspective. At the same time, SPCR uses adaptive weights to assign lower weights to low-confidence samples by combining the confidence of contrasting examples to reduce the impact of erroneous semantic information and help the model learn the correct target distribution. Furthermore, hard samples are crucial to fully understand the target semantic distribution[39][40][41][42]. However, existing SSDA methods ignore exploring hard samples due to their complex knowledge distribution. To fill this gap, we further explore the complex target relationships of hard target samples through

hard-sample mixup regularization (HMR). After screening out easy and hard samples using the classifier prototype, we use easy sample guidance to learn these hard samples. Specifically, HMR uses the regularization constraint of mixup[43] to mix easy samples with hard samples. This method further explores the potential knowledge of hard samples through the guidance of easy samples and further helps the model learn more complex target semantic information. Finally, even if we consider the discriminative knowledge representation and hard sample information of the target domain, there will still be bias in semantic learning when there is much noise in the target pseudo-labels. To reduce this misleading semantic information caused by noisy pseudo-labels, we minimize the impact of noisy pseudo-labels from the perspective of target prediction regularization (TPR). Inspired by[44] [45], we use the early prediction of samples to constrain the probability output of the model during the adaptation stage to encourage the model to follow early target sample predictions and alleviate overfitting of erroneous semantic information on the target domain.

In summary, our main contributions are as follows:

- We propose a novel SSDA framework called semantic regularization learning (SERL). The proposed SERL considers fully utilizing and learning semantic knowledge on the target domain to achieve cross-domain adaptation when fine-tuning the source model on the target domain.
- To fully utilize the semantic relationships of the target domain, we propose three regularization methods, $i.e.$, semantic probability contrastive regularization, hard-sample mixup regularization, and target prediction regularization, to constrain the performance of the model on the target domain through semantic regularization strategies and further learn the knowledge of the target domain.
- Extensive experiments conducted on three standard benchmark datasets, including DomainNet[46], Office-Home[47], and Office-31[48], have shown that our method has significant advantages over previous state-of-the-art SSDA methods.

The paper is structured as follows: In Section II, we provide an overview of prior research related to our work. Section IIIintroduces and describes the proposed algorithm for semi-supervised domain adaptation. In Section IV, we conduct comparative experiments to evaluate the performance of the proposed method. Finally, the conclusions of our approach are presented in Section V.

II. Related Work

A. Unsupervised Domain Adaptation

To solve the problem that traditional supervised learning requires much manual annotation, unsupervised domain adaptation (UDA) aims to transfer knowledge from a fully labeled source domain to an unlabeled target domain. In recent years, various methods have been proposed for UDA, and adequate progress has been achieved. Commonly used methods mainly include maximum mean difference (MMD)[49], whose basic idea is to achieve migration from the source domain to the target domain by minimizing the distance between feature distributions. DANN[50] and JAN[51] further proposed using the MMD criterion to learn transfer networks by cross-region alignment of multiple region-specific layers. CORAL[52] and DUCDA[53] proposed minimizing the domain shift by aligning the second-order statistics of the source and target distributions. Meanwhile, with the development of generative adversarial networks, many recent works[18][54][55][56][57][58][21][59] have used adversarial learning for domain alignment so that knowledge from classifiers trained on labeled source samples can be effectively transferred to the target domain. In addition, considering the perspective of conditional distributions, many related works[60][61][62] have proven that learning conditional distributions is of good help in reducing the differences in the alignment of classification domains, thereby improving the adaptability between domains. Although the UDA method has been successfully used in many practical applications, it takes work to accurately describe the conditional distribution of target features due to the significant differences between some source domains and target domains and the unreachability of target labels. Therefore, the potential of the UDA method in practical applications is limited compared to the SSDA method.

B. Semi-supervised Domain Adaptation

Semi-supervised domain adaptation (SSDA) aims to utilize a small number of labeled samples on the target domain. Compared with UDA, the classification performance and generalization ability of the model on the target domain can be significantly improved due to the access to labeled target samples. At present, SSDA has made much adequate progress, and the methods used in many works can be roughly divided into cross-domain alignment methods, adversarial training methods, and semi-supervised learning methods. In cross-domain alignment, many related works[28][45][63][64] integrate various complementary domain alignment technologies. G-ABC[34] further achieves

semantic alignment by forcing the transfer from labeled source and target data to unlabeled target samples. In addition, IDMNE[35] is proposed to incorporate the label information of labeled samples into the model to learn cross-domain class feature alignment. Utilizing the idea of adversarial training, many related methods[26][65][66][37][27][29] solve the SSDA problem by minimizing the entropy between the class prototype and adjacent unlabeled target domain samples to achieve the effect of adversarial training. To solve SSDA through the idea of semi-supervised learning, MCL[31] and ProML[33] further help the model understand the target domain that lacks a large number of labels through consistency regularization. Unlike most existing methods, DEEM[67] considers a source-free[38] scenario and proposes a self-distillation method to improve entropy minimization and help label propagation of unlabeled samples on the target domain. However, the above existing methods all need to pay more attention to the importance of profoundly exploring the semantic information of the target domain. This paper starts from the perspective of semantic regularization learning and proposes the SERL framework, which helps the model more comprehensively adapt to the actual target domain distribution by standardizing the knowledge representation learned by the model on the target domain.

III. Methodology

A. Preliminaries and Overview

In semi-supervised domain adaptation (SSDA), the model is expected to generalize well on the target domain with fully labeled source samples and a small number of labeled target samples. Specifically, the source domain dataset $\mathcal{S} = \{x_i^s, y_i^s\}_{i=1}^{N_s}$ contains fully labeled data, $\mathcal{L} = \{x_i^l, y_i^l\}_{i=1}^{N_l}$ contains a small amount of labeled data of the target domain, where N_s and N_l are the source domain and target domain dataset size respectively. Here, x_i^s and x_i^l represent the labeled source image and target image data, respectively, and y_i^s and y_i^l represent the corresponding labels. In addition to the labeled data, there is also an unlabeled target image set $\mathcal{U} = \{x_i^u\}_{i=1}^{N_u}$ for adaptation on the target domain, which contains unlabeled target image data, usually $N_u \gg N_l$. The overall objective used to optimize the model can be expressed as a combination of the loss of the base model and the additional loss, as follows:

$$L_{all} = L_{base} + \lambda_{prob} L_{prob} + \lambda_{mix} L_{mix} + \lambda_{pre} L_{pre}, \tag{1}$$

where λ_{prob}, λ_{mix} and λ_{pre} are scalar hyper-parameters of the loss weights and L_{prob}, L_{mix}, and L_{pre} represent semantic probability contrastive regularization, hard-sample mixup regularization and target prediction regularization respectively.

For the model trained on the source domain, we first use the cross-entropy loss to train the feature extractor $g(\cdot)$ and the linear classifier $f(\cdot)$. For the source data $\mathcal{S} = \{x_i^s, y_i^s\}_{i=1}^{N_s}$, we employ the standard cross-entropy objective:

$$L_s = \frac{1}{N_s} \sum_{i=1}^{N_s} L_{CE}\left(p(y_i^s|x_i^s), y_i^s\right). \tag{2}$$

Following [38][67], we freeze $f(\cdot)$ and train $g(\cdot)$ when the model adapts to the target domain. An overview of our SERL framework in the target adaptation stage is illustrated in Figure 3. Following [33], we generate the strong augment view for each unlabeled target sample x_i^u, represented as $\hat{x}_i^u$. The target samples are then fed to the same feature extractor $g(\cdot)$ and classifier $f(\cdot)$ to obtain the probabilistic predictions p_i^u, $\hat{p}_i^u$, and the model is further adapted by the proposed semantic regularization learning. For the labeled target data, we employ the standard cross-entropy objective:

$$L_l = \frac{1}{N_l} \sum_{i=1}^{N_l} L_{CE}\left(p(y_i^l|x_i^l), y_i^l\right), \tag{3}$$

where L_{CE} is the standard cross-entropy loss. For the unlabeled target data, we employ the cross-entropy objective for its pseudo-label:

$$L_u = \frac{1}{N_u} \sum_{i=1}^{N_u} L_{CE}\left(p(y_i^u|x_i^u), y_i^u\right), \tag{4}$$

where $y_i^u = \arg\max p_i^u$ is the pseudo-label of x_i^u. Then, we utilize the mutual information maximization objective to encourage individually certain and globally diverse predictions:

$$L_{mi} = \frac{1}{N_u} \sum_{i=1}^{N_u} \mathcal{H}(p(y_i^u|x_i^u)) - \mathcal{H}(\sum_{i=1}^{N_u} p(y_i^u|x_i^u)), \tag{5}$$

where the entropy metric $\mathcal{H}(p(y|x)) = \sum_{k=1}^{c} p_k \log p_k$ and c is the number of different categories.

Following [67], we use a KNN-based pseudo-label propagation method. In the neighbor graph, we can obtain one-hot pseudo-labels of unlabeled data through global propagation from labeled and low-uncertainty target data. Finally, the base learning objective on the target domain can be derived as follows:

$$L_{base} = L_l + L_u + L_{mi}. \tag{6}$$

On this basis, we will further introduce the proposed learning framework and how the training objective achieves further learning of the target domain through semantic probability contrastive regularization L_{prob}, hard-sample mixup regularization L_{mix}, and target prediction regularization L_{pre}.

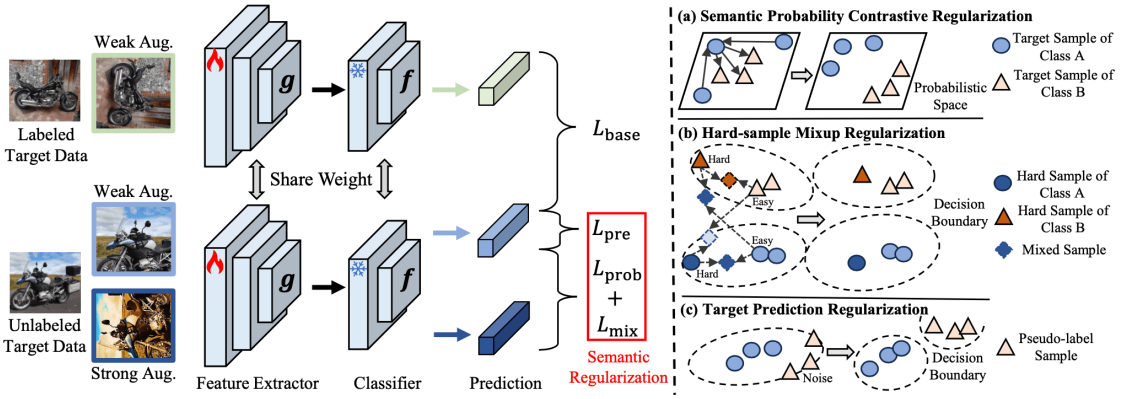

Figure 3. Illustration of our proposed semantic regularization learning (SERL) framework. **Left:** The model initialized on the source domain is adaptively fine-tuned on the target domain. The labeled target data and the strong and weak augmented versions of the unlabeled target data are input to the feature extractor g, then sent to the classifier f, and further learned the target domain knowledge through semantic regularization. The two feature extractors and classifiers used share parameter weight. **Right:** (a) Semantic probability contrastive regularization (SPCR) adaptively learns discriminative features through target semantic information and helps the model obtain a more confident probability output. (b) Hard-sample mixup regularization (HMR) uses the semantic information of easy samples to guide the model in learning the distribution of hard target samples, helping the model learn more complex target domain distributions. (c) Target prediction regularization (TPR) is used to minimize the misleading of erroneous semantic information to the model from the perspective of noise labels and help the model learn the true target domain distribution information.

B. Semantic Probability Contrastive Regularization

After the model is initialized on the source domain, it will be fine-tuned on the target domain, and this process will not access the source domain data so that we can convert this semi-supervised domain adaptive process into a semi-supervised fine-tuning process for the target domain. However, due to domain differences, the model still performs poorly on the target domain, even if it sees rich label information during the initialization of the source domain.

doi.org/10.32388/4T33D8

In recent years, contrastive learning[68][69][70][71][72][73][74][75] has been proven to be an adequate representation learning method, which helps models better understand data and learn helpful knowledge representations in unsupervised or semi-supervised scenarios by constraining sample representation. As a representative work, the self-supervised contrastive loss InfoNCE[68] takes the following format:

$$L_{InfoNCE} = -\sum_{i=1}^{2N_u} \log \frac{\exp(z_i \cdot z_i^+ / \tau)}{\sum_{j=1}^{2N_u} 1(j \neq i) \exp(z_i \cdot z_j / \tau)}, \tag{7}$$

where the z_i^+ represents the positive sample of feature embedding z_i, $1(j \neq i)$ represents the indicator function and $\tau = 0.15$ is the temperature coefficient.

In instance-based contrastive learning, two different augmented views from the same sample should be shown to represent similar features. However, the knowledge learned only considering instance-level relationships is limited in complex target domains. SupCon[70] learns more complex inter-sample relationships by introducing semantic information. It is equivalent to applying semantic information regularization constraints to the model, which helps improve the generalization performance of the model on the target domain. However, it is only applied to label-rich supervised learning, and feature-based contrastive learning cannot represent the actual target distribution of feature representations of many unlabeled target samples, which will impair the generalization ability of the classifier on the target domain[72]. Inspired by[72][76], we consider using semantic probability contrastive regularization based on adaptive weights to help the model better adapt to the target domain. Specifically, we consider the following loss:

$$L_{prob} = -\sum_{i=1}^{2N_u} \sum_{k=1}^{2N_u} w_{ik} \log \frac{\exp(p_i^u \cdot p_k^{u+} / \tau)}{\sum_{j=1}^{2N_u} 1_{j \neq i} \exp(p_i^u \cdot p_j^u / \tau)}, \tag{8}$$

where p_k^{u+} represents the predicted probability of the positive target sample k and the adaptive weight w_{ik} is defined as follows:

$$w_{ik} = \begin{cases} 1 & \text{if } k = i, \\ p_i^u \cdot p_k^u & \text{if } \arg\max p_i^u = \arg\max p_k^u, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

The adaptive weights give lower weights to low-confidence samples, which can help mitigate the impact of false constraints and reduce misunderstandings about the proper distribution of the target domain.

Compared with instance-level contrastive learning, we learn a more realistic target domain distribution by constraining the probability distribution of samples with the same semantic information. At the same time, we constrain the similarity of the target samples from a probability perspective. Specifically, for two samples i, j:

$$\begin{aligned} p_i^u \cdot p_j^u = 1 \Leftrightarrow \operatorname{argmax}(p_i^u) = \operatorname{argmax}(p_j^u) \\ \wedge \max(p_i^u) = \max(p_j^u) = 1. \end{aligned} \tag{10}$$

where $p_i^u \cdot p_j^u$ represents the product of two unsupervised samples, $\Leftrightarrow$ represents the equivalence, and $\wedge$ represents the logical AND relationship. Eq. 10 indicates that when optimizing L_{prob}, the model forces the product between similarities to be maximized ($i.e.$, the product is 1), which is equivalent to the probability value corresponding to the predicted category ($i.e.$, $\operatorname{argmax}(p_i^u)$) being 1. It encourages the prediction of the model to be close to the one-shot vector, $i.e.$, to make confident judgments on the target sample with the same semantic label, which helps to capture the semantic information on the target domain more effectively and has unique advantages in improving model performance. Different from [77][71], we do not need a large batch size or sample queue to build comparison relationships, which can further save model memory consumption.

C. Hard-sample Mixup Regularization

Through semantic probability contrastive regularization, the model has been able to have a basic understanding of the sample relationships between target domains. However, when we consider a more complex target domain relationship, $i.e.$, there are a certain number of complex samples on the target domain, which are usually distributed near the decision boundary and have low confidence, making it challenging to learn the complete target distribution further. Existing SSDA methods ignore this problem, which makes them perform poorly in the face of complex target domain distributions. Mixup is proven to reduce the overfitting tendency of the model by introducing a certain degree of regularization [43][78][79][80]. Therefore, we can consider using Mixup to mix the semantic information of samples with different difficulty levels so that the model can better learn the complex semantic feature distribution of the target domain rather than adapt to the easy target distribution.

An important issue is partitioning samples with varying degrees of difficulty through existing models. Previous work [81] revealed that the weight vector of the trained last layer classifier converges to a high-dimensional geometric structure, which maximizes the separation of paired angles for all classifiers. Another work [82] uses the weight vector of the classifier to construct pseudo-source

domain samples to help model learning compensate for the lack of source domain knowledge. Inspired by these works, we use the weight vectors of pre-trained classifiers on the source domain as anchors to divide easy and complex samples. Specifically, we first define the classifier weight vectors $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_c\}$ of each category on the source domain as category anchors, search for and divide a certain number of easy and hard samples on unlabeled target domain data based on their distance from the anchors:

$$x_c^{easy} = \text{argTOPK}(\min(\text{dist}\langle g(x^u), \mathbf{c}_c\rangle), N_u^{easy}), \tag{11}$$

$$x_c^{hard} = \text{argTOPK}(\max(\text{dist}\langle g(x^u), \mathbf{c}_c\rangle), N_u^{hard}), \tag{12}$$

where N_u^{easy} and N_u^{hard} represent the number of easy and hard samples, $argTOPK(\cdot, N)$ means taking the first N numbers, $\min(\cdot)$ and $\max(\cdot)$ mean sorting the objects from small to large/from large to small, $dist\langle\cdot,\cdot\rangle$ represents the cosine distance between samples, and x_c^{easy} and x_c^{hard} represent the set of easy and hard samples for the c-th category.

To further enhance the understanding of semantic information, we connect easy and hard samples with their augmented versions $\hat{x}^{easy}$, $\hat{x}^{hard}$ to construct a vector represented as $X^{easy} = concate(x^{easy}, \hat{x}^{easy})$ and $X^{hard} = concate(x^{hard}, \hat{x}^{hard})$. Furthermore, we will mix X^{easy} and X^{hard} to construct the following mixed training samples:

$$\begin{aligned} X_i^{mix} &= \theta X_i^{easy} + (1-\theta) X_j^{hard}, \\ y_i^{mix} &= \theta y_i^{easy} + (1-\theta) y_j^{hard}, \end{aligned} \tag{13}$$

where θ is the mixup coefficient sampled from a random Beta distribution $Beta(\alpha, \alpha)$, $\alpha = 1$. Following [78], we formulate the hard-sample mixup regularization loss as:

$$L_{mix} = \frac{1}{N_u} \sum_{i=1}^{N_u} \| f(g(X_i^{mix})) - y_i^{mix} \|_2^2, \tag{14}$$

where $\|\cdot\|_2$ represents the l_2 regularization.

Unlike the cross-entropy loss, it is bound and more robust due to the sensitivity to corrupted labels. The guidance of easy samples can help the model reduce the predicted distribution fluctuations between easy and hard samples when complex samples exhibit features more distinct from the source domain distribution[42]. At the same time, this method imposes more complex semantic regularization constraints on the model, reducing the difference in semantic learning between easy and hard target samples, thereby helping the model better adapt to more complex target domain distributions.

D. Target Prediction Regularization

Our method relies on pseudo-labels generated by the model to form semantic information and use this as regularization information on the target domain without accessing source domain data during training. Even if the learning of the model considers the discriminative knowledge of the target domain and hard sample information, the learning is still biased when there is much noise in the target pseudo-label. Therefore, it is necessary to reduce the model from being misled by the semantic information generated by incorrect pseudo-labels. However, existing SSDA methods ignore this impact, which will cause the model to generate noise due to domain shift and mislead the learning of clustering structures[75]. To reduce the misguidance brought by erroneous semantic information to the model, we minimize the impact of erroneous pseudo-labels from the perspective of prediction regularization and further help the model learn correct target domain distribution knowledge.

Inspired by[45][83][84][44], we exploit the early training phenomenon to address the potential spurious label noise problem. Specifically, the early training phenomenon shows that classifiers can predict mislabeled samples with relatively high accuracy in the early adaptation stage before memorizing mislabeled target data. To leverage predictions made during early training, we employ early learning regularization (ELR), encouraging model predictions to adhere to early sample predictions. The regularization term is given by:

$$L_{pre} = \frac{1}{N_u}\sum_{i=1}^{N_u}\log(1-\tilde{y}_i^{ut^\top}p_i^{ut}), \tag{(15)}$$

where p_i^{ut} is the target probability output at epoch t, $\tilde{y}_i^{ut} = \beta\tilde{y}_i^{u(t-1)} + (1-\beta)p_i^{ut}$ is the moving average prediction and $\beta = 0.7$ is the hyper-parameter.

Note that minimizing Eq. 15 forces p_i^{ut} to be close to $\tilde{y}_i^{ut}$. Therefore, Eq. 15 prevents the model from remembering target label noise by forcing the model predictions to stay close to the moving average predictions $\tilde{y}^{ut}$ of these most likely accurate target labels, further reducing the impact of noisy semantic information on the model brought about misguidance. Combined with all the components mentioned above, the whole algorithm of our SERL can be described using Algorithm 1.

Algorithm 1 SERL Framework for SSDA.

Input: Labeled source data $\{x_i^s, y_i^s\}_{i=1}^{N_s}$, labeled target data $\{x_i^l, y_i^l\}_{i=1}^{N_l}$, unlabeled target set $\{x_i^u\}_{i=1}^{N_u}$ and strongly augmented unlabeled target set $\{\hat{x}_i^u\}_{i=1}^{N_u}$. The number of training epochs T. The trade-off hyper-parameters λ_{prob}, λ_{mix}, and λ_{pre}.

Initialization: Freeze the final classifier layer f, and copy the parameters from the source feature extractor to the target feature extractor as initialization.

 # Adaptive on Target Domain
for each $t \in [1, T]$ **do**
  Compute self-supervised pseudo-labels for x_i^u.
  for each target data $x_i^u \in [1, N_u]$ **do**
    # Calculate Losses
    Compute the base loss L_{base} with Eq. 6;
    Compute the probability contrastive loss L_{prob} with Eq. 8;
    Compute the mixup loss L_{mix} with Eq. 14;
    Compute the prediction regularization loss L_{pre} with Eq. 15;
    # Parameter Optimization.
    Update the parameters in target feature extractor g via L_{all} in Eq. 1.
  end for
end for
return The updated parameters of the target feature extractor g.

doi.org/10.32388/4T33D8

SF	Method	R→C	R→P	P→C	C→S	S→P	R→S	P→R	Mean
×	S+T	55.6	60.6	56.8	50.8	56.0	46.3	71.8	56.9
×	DANN[50]	58.2	61.4	56.3	52.8	57.4	52.2	70.3	58.4
×	ENT[85]	65.2	65.9	65.4	54.6	59.7	52.1	75.0	62.6
×	MME[26]	70.0	67.7	69.0	56.3	64.8	61.0	76.1	66.4
×	UODA[65]	72.7	70.3	69.8	60.5	66.4	62.7	77.3	68.5
×	BiAT[66]	73.0	68.0	71.6	57.9	63.9	58.5	77.0	67.1
×	APE[37]	70.4	70.8	72.9	56.7	64.5	63.0	76.6	67.6
×	STar[63]	74.1	71.3	71.0	63.5	66.1	64.1	80.0	70.0
×	DECOTA[64]	79.1	74.9	76.9	65.1	72.0	69.7	79.6	73.9
×	CDAC[27]	77.4	74.2	75.5	67.6	71.0	69.2	80.4	73.6
×	CLDA[86]	76.1	75.1	71.0	63.7	70.2	67.1	80.1	71.9
×	ECACL[28]	75.3	74.1	75.3	65.0	72.1	68.1	79.7	72.8
×	ASDA[29]	77.0	75.4	75.5	66.5	72.1	70.9	79.7	73.9
×	MCL[31]	77.4	74.6	75.5	66.4	74.0	70.7	82.0	74.4
×	ProML[33]	78.5	75.4	77.8	70.2	74.1	72.4	84.0	76.1
×	SLA[32]	79.8	75.6	77.4	68.1	71.7	71.7	80.4	75.0
×	IDMNE[35]	79.6	76.0	79.4	71.7	75.4	73.5	82.1	76.8
×	G-ABC[34]	80.7	76.8	79.3	72.0	75.0	73.2	83.4	77.5
✓	DEEM[67]	79.7	78.1	77.0	71.9	77.7	76.7	85.4	78.1
✓	SERL (Ours)	**90.5**	**88.8**	**90.2**	**89.1**	**90.1**	**87.1**	**93.3**	**89.9**

Table I. Accuracy (%) on *DomainNet* under the settings of 1-shot and ResNet-34 as backbone networks.

doi.org/10.32388/4T33D8

SF	Method	R→C	R→P	P→C	C→S	S→P	R→S	P→R	Mean
×	S+T	60.0	62.2	59.4	55.0	59.5	50.1	73.9	60.0
×	DANN[50]	59.8	62.8	59.6	55.4	59.9	54.9	72.2	60.7
×	ENT[85]	71.0	69.2	71.1	60.0	62.1	61.1	78.6	67.6
×	MME[26]	72.2	69.7	71.7	61.8	66.8	61.9	78.5	68.9
×	UODA[65]	75.4	71.5	73.2	64.1	69.4	64.2	80.8	71.2
×	BiAT[66]	74.9	68.8	74.6	61.5	67.5	62.1	78.6	69.7
×	APE[37]	76.6	72.1	76.7	63.1	66.1	67.8	79.4	71.7
×	STar[63]	77.1	73.2	75.8	67.8	69.2	67.9	81.2	73.2
×	DECOTA[64]	80.4	75.2	78.7	68.6	72.7	71.9	81.5	75.6
×	CDAC[27]	79.6	75.1	79.3	69.9	73.4	72.5	81.9	76.0
×	CLDA[86]	77.7	75.7	76.4	69.7	73.7	71.1	82.9	75.3
×	ECACL[28]	79.0	77.3	79.4	70.6	74.6	71.6	82.4	76.4
×	ASDA[29]	79.4	76.7	78.3	70.2	74.2	72.1	82.3	76.2
×	MCL[31]	79.4	76.3	78.8	70.9	74.7	72.3	83.3	76.5
×	ProML[33]	80.2	76.5	78.9	72.0	75.4	73.5	84.8	77.4
×	SLA[32]	81.6	76.0	80.3	71.3	73.5	73.5	82.5	76.9
×	IDMNE[35]	80.8	76.9	80.3	73.2	75.4	73.9	82.8	77.5
×	G-ABC[34]	82.1	76.7	81.6	73.7	76.3	74.3	83.9	78.2
✓	DEEM[67]	80.5	79.0	77.5	74.9	80.0	75.9	88.5	79.5
✓	SERL (Ours)	**91.8**	**89.1**	**91.9**	**89.9**	**92.1**	**87.5**	**94.3**	**90.9**

Table II. Accuracy (%) on *DomainNet* under the settings of 3-shot and ResNet-34 as backbone networks.

SF	Method	R→C	R→P	R→A	P→R	P→C	P→A	A→P	A→C	A→R	C→R	C→A	C→P	Mean
×	S+T	39.5	75.3	61.2	71.6	37.0	52.0	63.6	37.5	69.5	64.5	51.4	65.9	57.4
×	DANN[50]	52.0	75.7	62.7	72.7	45.9	51.3	64.3	44.4	68.9	64.2	52.3	65.3	60.0
×	ENT[85]	23.7	77.5	64.0	74.6	21.3	44.6	66.0	22.4	70.6	62.1	25.1	67.7	51.6
×	MME[26]	49.1	78.7	65.1	74.4	46.2	56.0	68.6	45.8	72.2	68.0	57.5	71.3	62.7
×	UODA[65]	49.6	79.8	66.1	75.4	45.5	58.8	72.5	43.3	73.3	70.5	59.3	72.1	63.9
×	DECOTA[64]	47.2	80.3	64.6	75.5	47.2	56.6	71.1	42.5	73.1	71.0	57.8	72.9	63.3
×	ASDA[29]	51.6	80.9	66.9	75.9	49.7	60.5	71.0	44.9	73.2	70.6	58.7	72.8	64.7
×	IDMNE[35]	52.6	81.8	67.5	77.3	50.7	59.7	73.7	49.6	72.6	71.4	62.5	76.2	66.3
✓	DEEM[67]	62.5	82.1	68.5	79.0	62.1	65.4	76.5	60.3	76.1	74.6	63.3	75.4	70.5
✓	SERL (Ours)	**74.4**	**92.8**	**78.0**	**89.4**	**70.6**	**72.2**	**86.7**	**74.7**	**86.1**	**84.3**	**72.7**	**86.8**	**80.6**

Table III. Accuracy (%) on Office-Home under the settings of 1-shot using VGGNet-16 as the backbone network.

doi.org/10.32388/4T33D8

SF	Method	R→C	R→P	R→A	P→R	P→C	P→A	A→P	A→C	A→R	C→R	C→A	C→P	Mean
×	S+T	49.6	78.6	63.6	72.7	47.2	55.9	69.4	47.5	73.4	69.7	56.2	70.4	62.9
×	DANN[50]	56.1	77.9	63.7	73.6	52.4	56.3	69.5	50.0	72.3	68.7	56.4	69.8	63.9
×	ENT[85]	48.3	81.6	65.5	76.6	46.8	56.9	73.0	44.8	75.3	72.9	59.1	77.0	64.8
×	MME[26]	56.9	82.9	65.7	76.7	53.6	59.2	75.7	54.9	75.3	72.9	61.1	76.3	67.6
×	UODA[65]	57.6	83.6	67.5	77.7	54.9	61.0	77.7	55.4	76.7	73.8	61.9	78.4	68.9
×	APE[37]	56.0	81.0	65.2	73.7	51.4	59.3	75.0	54.4	73.7	71.4	61.7	75.1	66.5
×	DECOTA[64]	59.9	83.9	67.7	77.3	57.7	60.7	78.0	54.9	76.0	74.3	63.2	78.4	69.3
×	ASDA[29]	59.3	83.6	68.0	78.3	56.8	61.8	78.6	55.7	75.3	74.0	63.3	78.9	69.5
×	IDMNE[35]	60.2	84.4	69.3	77.9	59.2	62.6	77.7	58.2	76.7	74.9	64.6	79.3	70.4
✓	DEEM[67]	69.3	86.6	69.8	79.3	66.3	64.0	80.1	64.0	77.8	75.6	63.7	78.3	72.9
✓	SERL (Ours)	**79.6**	**92.8**	**78.4**	**90.0**	**78.3**	**72.8**	**90.1**	**78.4**	**86.8**	**89.6**	**74.2**	**91.5**	**83.5**

Table IV. Accuracy (%) on Office-Home under the settings of 3-shot using VGGNet-16 as the backbone network.

SF	Method	1-shot			3-shot		
		D→A	W→A	Mean	D→A	W→A	Mean
×	S+T	50.0	50.4	50.2	62.4	61.2	61.8
×	DANN[50]	54.5	57.0	55.8	65.2	64.4	64.8
×	ENT[85]	50.0	50.7	50.4	66.2	64.0	65.1
×	MME[26]	55.8	57.2	56.5	67.8	67.3	67.6
×	BiAT[66]	54.6	57.9	56.3	68.5	68.2	68.3
×	APE[37]	-	-	-	67.6	69.0	68.3
×	CLDA[86]	62.7	64.6	63.6	72.5	70.5	71.5
×	CDAC[27]	62.8	63.4	63.1	70.0	70.1	70.0
×	STar[63]	56.8	59.8	58.3	69.0	69.1	69.1
×	IDMNE[35]	-	-	-	71.3	71.0	71.2
×	G-ABC[34]	65.7	67.9	66.8	73.1	71.0	72.0
✓	DEEM[67]	75.7	76.6	76.2	76.8	78.5	77.7
✓	SERL (Ours)	**79.0**	**81.1**	**80.1**	**82.1**	**82.5**	**82.3**

Table V. Accuracy (%) on Office-31 under the settings of 1-shot and 3-shot using AlexNet as the backbone network.

IV. Experiment

A. Datasets

We evaluate our proposed method on three widely used datasets, including DomainNet[87], Office-Home[47], and Office-31[48]. For fairness of comparison, we have one or three samples on the target domain during training for each category in different datasets.

DomainNet is a significant benchmark dataset designed to evaluate multi-source domain adaptation methods composed of 345 classes, six domains: Clipart, Infographics, Painting, Real, Sketch, and Quickdraw, and each domain contains 126 image categories. Similar to MME[26], we use a subset of the DomainNet as one of our evaluation benchmarks. We only select four domains: Real (R), Clipart (C), Painting (P), and Sketch (S), because other domains and categories may contain excessive sample noise. Following MME[26], we conduct adaptation experiments on seven scenarios on these four domains.

Office-Home is a medium-sized SSDA benchmark dataset with many challenging object recognition domain adaptation scenarios. It consists of four domains: Art (A), Clipart (C), Products (P), and Real (R). The dataset contains images of 65 object classes typically constructed in office and home environments for each domain. We consider 12 domain adaptation scenarios compared with previous SSDA methods to achieve a fair comparison.

Office-31 is a small dataset containing three domains: Amazon (A), DSLR (D), and Webcam (W), with 31 categories on each domain. Following MME[26], we choose Amazon (A) as the target domain because compared to Webcam (W) and DSLR (D), each category in Amazon has sufficiently rich samples. Therefore, we only consider two adaptation scenarios on this small SSDA dataset: "W→A" and "D→A."

B. Implementation Details

We select three feature extraction backbones, including AlexNet[1], VGGNet-16[88], and ResNet-34[2] with pre-trained weights on ImageNet[1]. Similar to[38][67], for AlexNet and VGGNet-16, we add a bottleneck layer after the last layer of the feature extractor. We then use a classifier with a normalized, fully connected layer. For ResNet-34, we remove the last layer of the feature extractor, add a bottleneck layer like the previous backbone network, and use a classifier with fully connected layers. We randomly select three mini-batches from N_s, N_l, and N_u during each iteration. For batch sizes, they are 64, 32, and 64 for AlexNet, 32, 16, and 32 for VGGNet-16, and 48, 24, and 48 for ResNet-34. The learning rates of the feature extractor, bottleneck layer, and classifier are set to 0.001, 0.01, and 0.01, respectively, and the weight decay is 0.0005. The loss weights λ_{prob}, λ_{mix}, and λ_{pre} are specified as 0.3, 60, and 3, respectively. The number of easy samples N_u^{easy} and hard samples N_u^{hard} are 15. We adopt the widely used Randaugmnt[89] as the strong data augmentation strategy. Our

experiments were implemented using Pytorch[90] and run on an RTX 3090 GPU. We use three different randomized seeds to obtain fairer experimental results.

C. Comparison With State-of-the-Arts

In this section, we compare the classification performance of our proposed SERL method with previous state-of-the-art SSDA algorithms, including S+T, DANN[50], ENT[85], MME[26], UODA[65], BiAT[66], APE[37], STar[63], DECOTA[64], ECACL[28], ASDA[29], MCL[31], SLA[32], CLDA[86], CDAC[27], ProML[33], DEEM[67], IDMNE[35], G-ABC[34]. Note that S+T refers to the method of training an adaptive model by supervising only labeled samples from two domains, DANN[50] applies standard cross-entropy loss to SSDA by using it to some labeled samples on the target domain. SLA[32] is a plug-and-play SSDA method, and we consider combining it with CDAC[27], the best result reported in their paper.

1. Results on DomainNet

Tables I and II present the quantitative comparison results of our proposed method with numerous existing alternatives on the DomainNet benchmark. For the large dataset DomainNet, we use 1-shot and 3-shot settings and ResNet-34 with a relatively deep network structure as the corresponding backbone network. It can be seen from the results that our method outperforms all previous methods in all scenarios on 1-shot and 3-shot settings and achieves enormous advantages. Specifically, SERL improves the previous best-performing SSDA algorithm DEEM in the 1-shot and 3-shot settings of all adaptive scenarios, respectively, with the average accuracy increased by 11.8% and 11.4%. It is worth noting that DEEM is also based on the source-free SSDA method, but our performance is better, which is all attributed to our semantic regularization learning method. Most of the existing methods use the source-with training paradigm. Compared with them, we have improved the average accuracy of G-ABC by 12.4% and 12.7% in the 1-shot and 3-shot settings, respectively. By comparing the two tables, we can find that the performance in the 1-shot setting is slightly inferior to the improvement in the 3-shot setting. This means that our method requires more supervision to realize its potential better since more labeled target examples help better to learn the semantic information of the target domain.

2. Results on Office-Home

To further validate the feasibility of the proposed SERL framework in SSDA scenarios, Tables III and IV present the quantitative results and comparison of our method in benchmark Office-Home compared to previous methods. We conducted experiments on the dataset using VGGNet-16 as the backbone network in 1-shot and 3-shot settings and all 12 Office-Home adaptation scenarios. It is worth noting that our method outperforms all existing methods in all scenarios and significantly outperforms the source-free SSDA method DEEM by 10.1% for 1-shot and 10.6% for 3-shot and the source-with method IDMNE by 14.3% for 1-shot and 13.1% for 3-shot in terms of average accuracy, further demonstrating the superiority of our method.

3. Results on Office-31

Table V shows the results of our comparison with existing methods on Office-31. Office-31 is a small dataset, and in order to maintain consistency with existing methods, we use AlexNet with a relatively small number of layers to conduct experiments under 1-shot and 3-shot. It can be seen from the results that the average accuracy of our method under the 1-shot setting is 80.1%, and the average accuracy under the 3-shot setting is 82.3%, respectively surpassing the existing state-of-the-art SSDA method DEEM 3.9% and 4.6%. Compared with DomainNet and Office-Home, its performance improvement is relatively limited. This is because Office-31 contains a few images and is a relatively simple SSDA dataset. In contrast, DomainNet and Office-Home have richer image data, providing more challenging environments and room for improvement. This shows that our method is more capable of handling more complex domain adaptation scenarios than existing methods, proving the superiority of the proposed method on SSDA tasks.

D. Ablation Study

1. Each Main Component

We conducted ablation studies on the main components in 1-shot and 3-shot settings for DomainNet R$\rightarrow$C and R$\rightarrow$P, as shown in Table VI. Rows 2-4 show that each component can produce significant improvements. Rows 5-7 show that each combination still improves performance, indicating the versatility of the proposed module. At the same time, the SPCR module and the TPR module can bring more significant improvements to the model than the HMR module. This is because, in the SPCR and

TPR modules, the model has learned good feature representations for most samples on the target domain, resulting in a limited number of potentially hard samples, so the improvement is relatively limited. The best performance is achieved when all components of the model are activated.

Num.	L_{base}	L_{prob}	L_{mix}	L_{pre}	**R→C**	**R→P**	**Mean**
1	✓				79.0	77.8	78.4
2	✓	✓			86.4	83.9	85.2
3	✓		✓		81.1	79.8	80.5
4	✓			✓	83.7	81.9	82.8
5	✓	✓	✓		87.4	85.8	86.6
6	✓	✓		✓	88.9	87.6	88.3
7	✓		✓	✓	84.3	83.1	83.7
8	✓	✓	✓	✓	**90.5**	**88.8**	**89.7**

Table VI. Accuracy (%) of ablation study on DomainNet under the settings of 1-shot with the ResNet-34 backbone.

2. *Source-Free Learning Framework*

To prove the importance of the source-free training framework, we show the ablation experimental results in different cross-domain scenarios of Office-Home in Figure 4. When source-free training strategies and label propagation methods are not considered, the performance of the model will drop to the lowest point. This shows that the source-free training framework can allow the model to focus on learning a more accurate target domain distribution, thereby significantly improving the performance of the model. \addedSince the target domain has only a small amount of labeled data, while the source domain has a large amount of labeled data for supervision. The number of this part of supervision signals creates a strong contrast between the source domain and the target domain. When there are only a few labeled data, the model can easily rely on the characteristics of the source domain to make decisions. When considering either alone, the performance of the model drops significantly compared to the performance of the complete model. In particular, the source-free training method

can bring more significant performance improvement to the model. This is because source-free only considers fine-tuning the source model on the target domain, which can reduce the impact of source domain samples on the adaptation of the target domain during training, allowing the model to focus more on learning semantic information on the target domain.

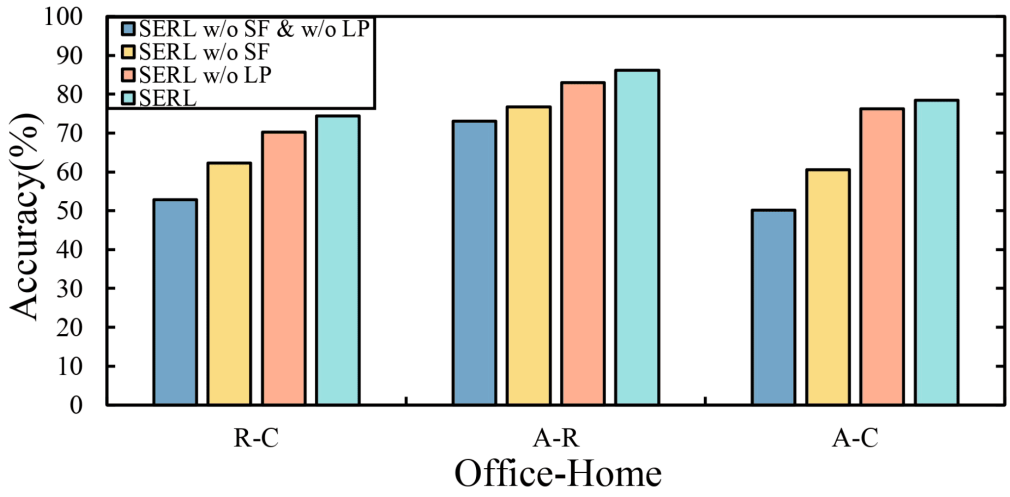

Figure 4. The impact of source-free learning frameworks on performance. The experiments were conducted in three scenarios of *Office-Home* under the 1-shot setting. *SF* stands for source-free training paradigm, and *LP* stands for label propagation.

3. Probability Contrast and Adaptive Weight in SPCR

We investigated specific techniques mentioned in SPCR to prove the effectiveness of our SPCR further, as shown in Table VII. It is worth noting that when nothing is considered, the model degrades to the InfoNCE loss, as shown in Eq. 7. When considering learning discriminative features from the probability space, the model performance improves significantly because the model is forced to output more confident representation information and can be combined with the knowledge learned by the classifier to allow the feature extractor to learn more compact target representation clusters. When considering adding adaptive weights, the model can adaptively learn relevant target representations for objects of the same category with different confidence levels, thereby achieving the best performance.

doi.org/10.32388/4T33D8

Probability Contrast	Adaptive Weight	Office-Home R→P	DomainNet C→S	Mean
		84.0	83.1	83.6
✓		88.6	87.6	88.1
✓	✓	**92.8**	**89.1**	**91.0**

Table VII. Accuracy (%) of ablation study for Probability Contrast and Adaptive Weight in SPCR with 1-shot setting.

E. Further Analysis

1. Sensitivity of λ_{prob}, λ_{mix} and λ_{pre}

We show the impact of the loss balance parameters λ_{prob}, λ_{mix} and λ_{pre} on the classification accuracy under the Office-Home C→A scenario in Figure 5. It can be observed that when $\lambda_{prob} = 0.1$, $\lambda_{mix} = 60$, and $\lambda_{pre} = 3$, the trained model achieves the highest performance in image classification.

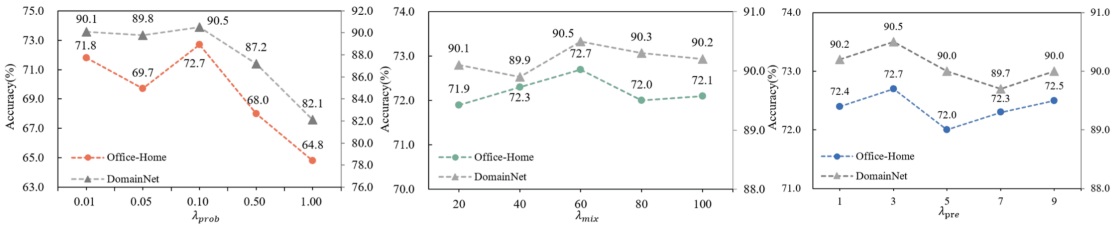

Figure 5. The effect of different loss balance parameters λ_{prob}, λ_{mix}, and λ_{pre} on the model classification accuracy in the *Office-Home* C→A and *DomainNet* R→C scenario under the 1-shot setting.

2. Sensitivity of N_u^{easy} and N_u^{hard} in HMR

Regarding the number of easy and hard samples we mentioned in HMR, *i.e.*, N_u^{easy} and N_u^{hard}, we further analyze its impact on model performance, as shown in Figure 6. It can be seen from the results that the blue part is mainly concentrated in areas with a small number of samples, which shows that the model performs poorly when the number of mixed samples is small. The model performance

doi.org/10.32388/4T33D8

improves when the number of mixed samples is gradually increased. This proves that the model can learn new target domain semantic representations by gradually adding the number of target samples.

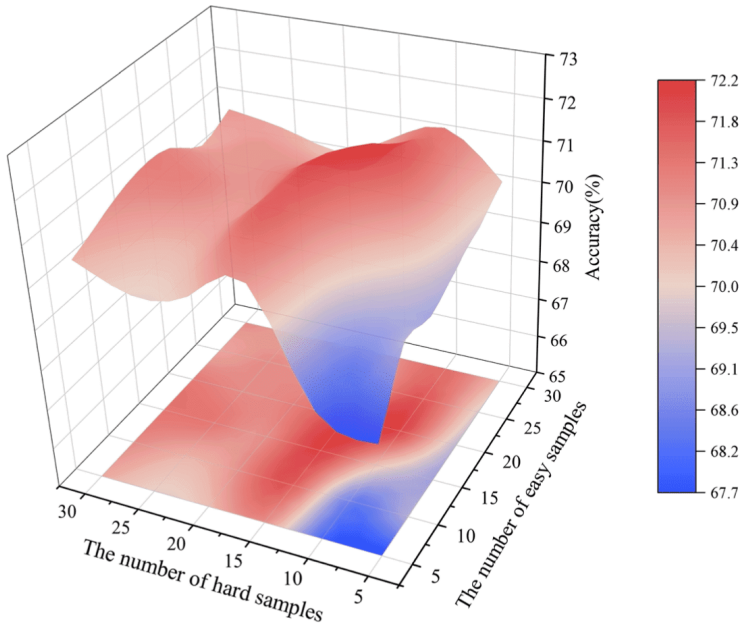

Figure 6. Variation in model performance for different numbers of easy and hard samples $N_u^{easy}, N_u^{hard} \in \{5, 10, 15, 20, 25, 30\}$ for the 1-shot setting in the P→A scenario of the Office-Home dataset.

3. Sensitivity of Labeled Samples

In Figure 7, we show histograms comparing our method with existing methods under different labeled samples. Our method still maintains optimal performance even with more labeled data. At the same time, as the number of labels increases, the improvement of methods gradually decreases. This phenomenon suggests diminishing returns to more labels, eventually leading to a fully supervised learning situation.

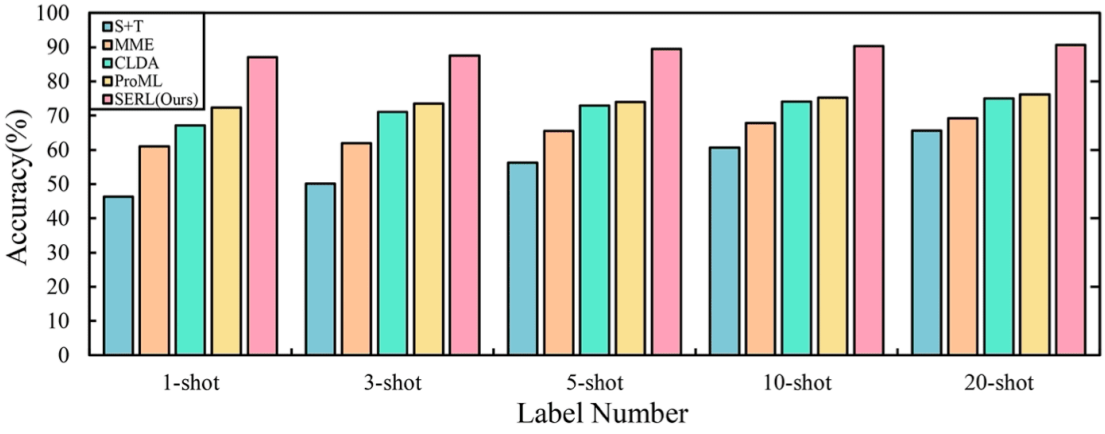

Figure 7. Histogram of quantitative comparison under different number of labeled samples settings on DomainNet R→S.

4. Spectral Analysis

To further analyze the discriminability of the learned features, following[91][92], we perform singular value decomposition (SVD) analysis on the feature matrices extracted under the 1-shot setting for the Office-31 W→A and D→A scenario. The results are shown in Figure 8. Relative to SERL, the largest singular values of the feature matrices of S+T and DEEM are significantly larger than the other singular values, greatly weakening the information signal of the feature vectors corresponding to smaller singular values. Such a sharp distribution of singular values implies a deterioration of distinguishability. However, the singular values of the feature matrices learned by our proposed SERL successfully reduce the large difference between the largest value and the remaining values while maintaining higher values, which implies that more dimensions corresponding to smaller singular values positively affect the classification and intuitively improve the discriminability of the features.

doi.org/10.32388/4T33D8

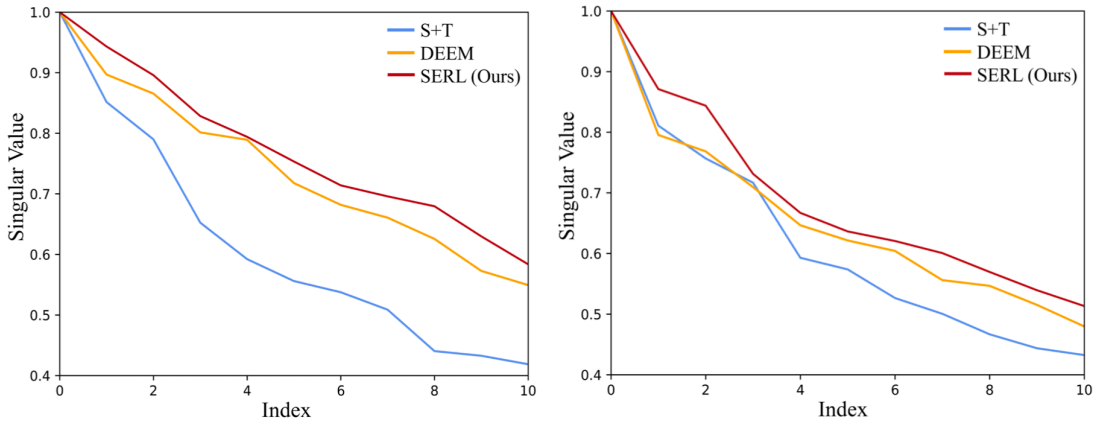

Figure 8. The SVD analysis of feature matrices obtained by different methods in different 1-shot scenarios.

F. Feature Visualization

1. Feature Aggregation

As shown in Figure 9, we use t-SNE[93] to visualize the changes in deep features during training. For DomainNet R→S, where the domain difference is relatively small, the model trained only on the source domain can better aggregate most of the same features. However, it performs poorly in Office-Home C→P, where the domain difference is relatively significant. However, as training proceeds, learned features from different domains belonging to the same class are mapped nearby and clustered together, while those from different classes are clearly separated, and the clusters are more evenly distributed. The results show that using the proposed SERL can produce domain-invariant and differentiated target features, helping the model perform well on the target domain.

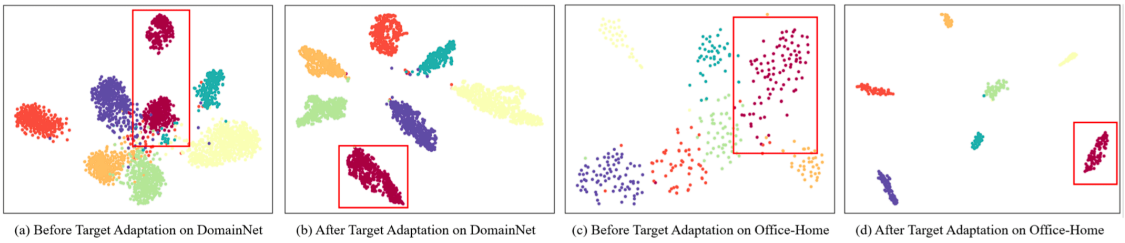

Figure 9. Feature visualization using t-SNE[93]. We randomly selected seven categories and assigned them different colors for the 3-shot scenes of *DomainNet* R→S and *Office-Home* C→P. The red box shows obvious differences.

2. *Attention Visualization*

In Figure 10, we use the Grad-CAM[94] to visualize the attention maps of the model for different categories of target samples in the DomainNet dataset after target domain adaptation. Whether the model faces easy samples with relatively easy backgrounds or hard samples with relatively complex backgrounds, the model can capture the key information of the target samples, which is due to the assistance of our SERL for the model to learn the semantic information on the target domain.

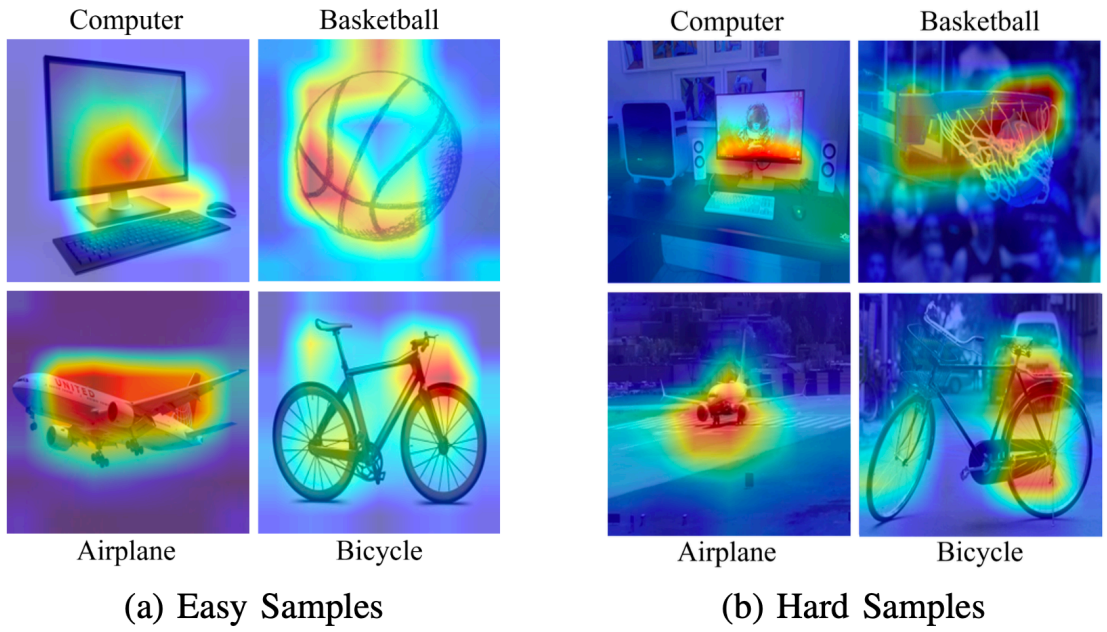

Figure 10. The Grad-CAM[94] visualization of the features generated by our SERL for different samples in the *DomainNet* dataset.

doi.org/10.32388/4T33D8

V. Conclusion

This paper proposes a novel SSDA learning framework called semantic regularization learning (SERL), which provides regularization constraints by learning semantic information from the target data, thereby better learning the representation distribution of the target domain. This paper considers fine-tuning the feature extractor on the target domain based on the source pre-trained model. Firstly, semantic probability contrastive regularization helps the model learn more discriminative feature representations, using semantic information on the target domain to understand the similarities and differences between samples. At the same time, it encourages the model to make confident judgments, helping to capture the semantic information on the target domain more fully. Then, hard-sample mixup regularization is proposed to learn more complex target domains by reducing the fluctuation of predictive distributions between easy and hard samples through a guidance strategy for easy samples. Finally, target prediction regularization corrects erroneous target predictions by maximizing the correlation between the prediction output and the early learned target, reducing the misleading of false semantic information. Extensive experiments and comprehensive analysis with good performance on three benchmark datasets demonstrate the superiority of our method, which significantly surpasses existing methods and achieves impressive results.

References

1. a, b, c *Krizhevsky A, Sutskever I, Hinton GE (2012). "Imagenet classification with deep convolutional neural networks." Advances in Neural Information Processing Systems.* **25**.
2. a, b *He K, Zhang X, Ren S, Sun J (2016). "Deep residual learning for image recognition." In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778.*
3. ^ *Rastegari M, Ordonez V, Redmon J, Farhadi A (2016). "Xnor-net: Imagenet classification using binary convolutional neural networks." In: European conference on computer vision. pp. 525–542. Springer.*
4. ^ *Krizhevsky A, Sutskever I, Hinton GE (2017). "Imagenet classification with deep convolutional neural networks." Communications of the ACM.* **60** *(6): 84–90.*
5. ^ *Qiao L, Shi Y, Li J, Wang Y, Huang T, Tian Y (2019). "Transductive episodic-wise adaptive metric for few-shot learning." In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3603–3612.*

6. ^Yan Y, Nie F, Li W, Gao C, Yang Y, Xu D (2016). "Image classification by cross-media active learning wi th privileged information." IEEE Transactions on Multimedia. **18** (12): 2494–2502.
7. ^Wang J, Wang W, Wang R, Gao W (2016). "Csps: An adaptive pooling method for image classification." IEEE Transactions on Multimedia. **18**(6): 1000–1010.
8. ^Long J, Shelhamer E, Darrell T (2015). "Fully convolutional networks for semantic segmentation." In: P roceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440.
9. ^Gao G, Xu G, Li J, Yu Y, Lu H, Yang J (2022). "Fbsnet: A fast bilateral symmetrical network for real-time semantic segmentation." IEEE Transactions on Multimedia.
10. ^Yan B, Niu X, Bare B, Tan W (2019). "Semantic segmentation guided pixel fusion for image retargetin g." IEEE Transactions on Multimedia. **22** (3): 676–687.
11. ^Minaee S, Boykov Y, Porikli F, Plaza A, Kehtarnavaz N, Terzopoulos D (2021). "Image segmentation us ing deep learning: A survey." IEEE Transactions on Pattern Analysis and Machine Intelligence. **44** (7): 3 523–3542.
12. ^Li Z, Ye W, Terven J, Bennett Z, Zheng Y, Jiang T, Huang T (2023). "Muva: A new large-scale benchmar k for multi-view amodal instance segmentation in the shopping scenario." In: Proceedings of the IEEE/ CVF International Conference on Computer Vision. pp. 23504–23513.
13. ^Wang X, Zhang X, Cao Y, Wang W, Shen C, Huang T (2023). "Seggpt: Segmenting everything in contex t." arXiv preprint arXiv:2304.03284. Available from: https://arxiv.org/abs/2304.03284.
14. ^Pan SJ, Tsang IW, Kwok JT, Yang Q (2010). "Domain adaptation via transfer component analysis." IEE E Transactions on Neural Networks. **22** (2): 199–210.
15. ^Patel VM, Gopalan R, Li R, Chellappa R (2015). "Visual domain adaptation: A survey of recent advance s." IEEE Signal Processing Magazine. **32** (3): 53–69.
16. ^Mei K, Zhu C, Zou J, Zhang S. "Instance adaptive self-training for unsupervised domain adaptation." I n: Computer Vision--ECCV 2020: 16th European Conference, Glasgow, UK, August 23--28, 2020, Proce edings, Part XXVI 16. Springer; 2020. p. 415-430.
17. ^Zhang C, Li Z, Liu J, Peng P, Ye Q, Lu S, Huang T, Tian Y (2021). "Self-guided adaptation: Progressive r epresentation alignment for domain adaptive object detection". IEEE Transactions on Multimedia. **24**: 2 246–2258.
18. a, bGanin Y, Lempitsky V (2015). "Unsupervised domain adaptation by backpropagation." In: Internatio nal conference on machine learning. PMLR. pp. 1180–1189.

19. ^Deng W, Zhao L, Liao Q, Guo D, Kuang G, Hu D, Pietikäinen M, Liu L (2021). "Informative feature disen tanglement for unsupervised domain adaptation." IEEE Transactions on Multimedia. **24**: 2407–2421.
20. ^Wang R, Wu Z, Weng Z, Chen J, Qi GJ, Jiang YG. "Cross-domain contrastive learning for unsupervised d omain adaptation." IEEE Transactions on Multimedia. 2022.
21. a, bJing M, Meng L, Li J, Zhu L, Shen HT. "Adversarial mixup ratio confusion for unsupervised domain ad aptation." IEEE Transactions on Multimedia. 2022.
22. ^Lu Y, Li D, Wang W, Lai Z, Zhou J, Li X (2021). "Discriminative invariant alignment for unsupervised do main adaptation." IEEE Transactions on Multimedia. **24**: 1871–1882.
23. ^Zhao S, Yue X, Zhang S, Li B, Zhao H, Wu B, Krishna R, Gonzalez JE, Sangiovanni-Vincentelli AL, Seshi a SA, et al. (2020). "A review of single-source deep unsupervised visual domain adaptation." IEEE Tran sactions on Neural Networks and Learning Systems. **33** (2): 473–493.
24. ^Zuo Y, Yao H, Zhuang L, Xu C (2023). "Dual structural knowledge interaction for domain adaptation." IEEE Transactions on Multimedia. (99): 1–15.
25. ^Ding F, Li J, Tian W, Zhang S, Yuan W (2023). "Unsupervised domain adaptation via risk-consistent est imators." IEEE Transactions on Multimedia.
26. a, b, c, d, e, f, g, h, i, j, k, l, mSaito K, Kim D, Sclaroff S, Darrell T, Saenko K (2019). "Semi-supervised domai n adaptation via minimax entropy." In: Proceedings of the IEEE/CVF International Conference on Comp uter Vision. pp. 8050–8058.
27. a, b, c, d, e, f, g, hLi J, Li G, Shi Y, Yu Y. "Cross-domain adaptive clustering for semi-supervised domain ad aptation." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 20 21:2505-2514.
28. a, b, c, d, e, fLi K, Liu C, Zhao H, Zhang Y, Fu Y. "Ecacl: A holistic framework for semi-supervised domain adaptation." In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021. p. 857 8-8587.
29. a, b, c, d, e, f, g, hQin C, Wang L, Ma Q, Yin Y, Wang H, Fu Y (2022). "Semi-supervised domain adaptive st ructure learning." IEEE Transactions on Image Processing. **31**: 7179–7190.
30. a, bXu H-M, Liu L, Bian Q, Yang Z (2022). "Semi-supervised semantic segmentation with prototype-ba sed consistency regularization." Advances in Neural Information Processing Systems.
31. a, b, c, d, e, f, gYan Z, Wu Y, Li G, Qin Y, Han X, Cui S (2022). "Multi-level consistency learning for semi-s upervised domain adaptation." arXiv preprint arXiv:2205.04066. Available from: https://arxiv.org/abs/2205.04066.

32. a, b, c, d, e, f Yu Y-C, Lin H-T. "Semi-supervised domain adaptation with source label adaptation." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023. p. 24100-24109.

33. a, b, c, d, e, f, g, h Huang X, Zhu C, Chen W (2023). "Semi-supervised domain adaptation via prototype-based multi-level learning." arXiv preprint arXiv:2305.02693. Available from: https://arxiv.org/abs/2305.02693.

34. a, b, c, d, e, f, g, h Li J, Li G, Yu Y. "Adaptive betweenness clustering for semi-supervised domain adaptation." IEEE Transactions on Image Processing. 2023.

35. a, b, c, d, e, f, g, h, i Li J, Li G, Yu Y. "Inter-domain mixup for semi-supervised domain adaptation." Pattern Recognition. **146**: 110023, 2024.

36. ^ Chen T, Guo Y, Hao S, Hong R (2023). "Semi-supervised domain adaptation for major depressive disorder detection." IEEE Transactions on Multimedia.

37. a, b, c, d, e, f, g Kim T, Kim C (2020). "Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation." In: European conference on computer vision. pp. 591–607. Springer.

38. a, b, c, d Liang J, Hu D, Feng J (2020). "Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation." In: International Conference on Machine Learning. PMLR. pp. 6028–6039.

39. ^ Xuan H, Stylianou A, Liu X, Pless R. "Hard negative examples are hard, but useful." In: Computer Vision--ECCV 2020: 16th European Conference, Glasgow, UK, August 23--28, 2020, Proceedings, Part XIV 16. Springer; 2020. p. 126-142.

40. ^ Zuo L, Jing M, Li J, Zhu L, Lu K, Yang Y (2021). "Challenging tough samples in unsupervised domain adaptation." Pattern Recognition. **110**: 107540.

41. ^ Liu Y, Ge H, Sun L, Hou Y (2022). "Complementary attention-driven contrastive learning with hard-sample exploring for unsupervised domain adaptive person re-id." IEEE Transactions on Circuits and Systems for Video Technology. **33** (1): 326–341.

42. a, b Xiong Y, Chen H, Lin Z, Zhao S, Ding G (2023). "Confidence-based visual dispersal for few-shot unsupervised domain adaptation." In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 11621–11631.

43. a, b Zhang H, Cisse M, Dauphin YN, Lopez-Paz D (2017). "mixup: Beyond empirical risk minimization." arXiv preprint arXiv:1710.09412. Available from: https://arxiv.org/abs/1710.09412.

44. ^a, ^bLiu S, Niles-Weed J, Razavian N, Fernandez-Granda C (2020). "Early-learning regularization prevents memorization of noisy labels." Advances in Neural Information Processing Systems. 33: 20331–20342.

45. ^a, ^b, ^cYi L, Xu G, Xu P, Li J, Pu R, Ling C, McLeod AI, Wang B (2023). "When source-free domain adaptation meets learning with noisy labels." arXiv preprint arXiv:2301.13381. Available from: https://arxiv.org/abs/2301.13381.

46. ^^Pei Z, Cao Z, Long M, Wang J (2018). "Multi-adversarial domain adaptation." In: Thirty-second AAAI conference on artificial intelligence.

47. ^a, ^bVenkateswara H, Eusebio J, Chakraborty S, Panchanathan S (2017). "Deep hashing network for unsupervised domain adaptation." In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5018–5027.

48. ^a, ^bSaenko K, Kulis B, Fritz M, Darrell T (2010). "Adapting visual category models to new domains." In: Computer Vision--ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11. Springer. pp. 213–226.

49. ^^Gretton A, Borgwardt KM, Rasch MJ, Schölkopf B, Smola A (2012). "A kernel two-sample test." The Journal of Machine Learning Research. **13** (1): 723–773.

50. ^a, ^b, ^c, ^d, ^e, ^f, ^g, ^hGanin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F, Marchand M, Lempitsky V (2016). "Domain-adversarial training of neural networks." The Journal of Machine Learning Research. **17** (1): 2096–2030.

51. ^^Long M, Zhu H, Wang J, Jordan MI (2017). "Deep transfer learning with joint adaptation networks." In: International Conference on Machine Learning. PMLR. pp. 2208–2217.

52. ^^Sun B, Feng J, Saenko K (2017). "Correlation alignment for unsupervised domain adaptation." Domain adaptation in computer vision applications. pp. 153–171.

53. ^^Zhuo J, Wang S, Zhang W, Huang Q (2017). "Deep unsupervised convolutional domain adaptation." In: Proceedings of the 25th ACM international conference on Multimedia, pp. 261–269.

54. ^^Tzeng E, Hoffman J, Saenko K, Darrell T (2017). "Adversarial discriminative domain adaptation." In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7167–7176.

55. ^^Zhang B, Chen T, Wang B, Li R (2021). "Joint distribution alignment via adversarial learning for domain adaptive object detection." IEEE Transactions on Multimedia. **24**: 4102–4112.

56. ^^Xie S, Zheng Z, Chen L, Chen C (2018). "Learning semantic representations for unsupervised domain adaptation." In: International Conference on Machine Learning. PMLR. pp. 5423–5432.

57. ^Shen J, Qu Y, Zhang W, Yu Y (2018). "Wasserstein distance guided representation learning for domain adaptation." In: Proceedings of the AAAI Conference on Artificial Intelligence. 32.

58. ^Ge P, Ren CX, Xu XL, Yan H (2023). "Unsupervised domain adaptation via deep conditional adaptation network." Pattern Recognition. 134: 109088.

59. ^Shermin T, Lu G, Teng SW, Murshed M, Sohel F (2020). "Adversarial network with multiple classifiers f or open set domain adaptation." IEEE Transactions on Multimedia. 23: 2732–2744.

60. ^Chen C, Xie W, Huang W, Rong Y, Ding X, Huang Y, Xu T, Huang J (2019). "Progressive feature alignme nt for unsupervised domain adaptation." In: Proceedings of the IEEE/CVF Conference on Computer Visio n and Pattern Recognition, pp. 627–636.

61. ^Pan Y, Yao T, Li Y, Wang Y, Ngo C-W, Mei T (2019). "Transferrable prototypical networks for unsupervi sed domain adaptation." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern R ecognition, pp. 2239–2247.

62. ^Zhong L, Fang Z, Liu F, Lu J, Yuan B, Zhang G (2021). "How does the combined risk affect the performa nce of unsupervised domain adaptation approaches?" in Proceedings of the AAAI Conference on Artifici al Intelligence. 35: 11079–11087.

63. a, b, c, d, eSingh A, Doraiswamy N, Takamuku S, Bhalerao M, Dutta T, Biswas S, Chepuri A, Vengatesan B, Natori N. "Improving semi-supervised domain adaptation using effective target selection and seman tics." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021. p. 2709-2718.

64. a, b, c, d, e, fYang L, Wang Y, Gao M, Shrivastava A, Weinberger KQ, Chao W-L, Lim S-N (2021). "Deep co -training with task decomposition for semi-supervised domain adaptation." In: Proceedings of the IEE E/CVF International Conference on Computer Vision. pp. 8906–8916.

65. a, b, c, d, e, fQin C, Wang L, Ma Q, Yin Y, Wang H, Fu Y (2021). "Contradictory structure learning for semi -supervised domain adaptation." In: Proceedings of the 2021 SIAM International Conference on Data M ining (SDM). SIAM. pp. 576–584.

66. a, b, c, d, eJiang P, Wu A, Han Y, Shao Y, Qi M, Li B (2020). "Bidirectional adversarial training for semi-s upervised domain adaptation." In: IJCAI. pp. 934–940.

67. a, b, c, d, e, f, g, h, i, jMa N, Bu J, Lu L, Wen J, Zhou S, Zhang Z, Gu J, Li H, Yan X (2022). "Context-guided e ntropy minimization for semi-supervised domain adaptation." Neural Networks. 154: 270–282.

68. a, bA. v. d. Oord, Y. Li, and O. Vinyals (2018). "Representation learning with contrastive predictive codin g." arXiv preprint arXiv:1807.03748. Available from: https://arxiv.org/abs/1807.03748.

69. ^Grill J-B, Strub F, Altché F, Tallec C, Richemond P, Buchatskaya E, Doersch C, Avila Pires B, Guo Z, Ghes hlaghi Azar M, et al. (2020). "Bootstrap your own latent-a new approach to self-supervised learning." Advances in Neural Information Processing Systems. 33: 21271–21284.

70. a, bKhosla P, Teterwak P, Wang C, Sarna A, Tian Y, Isola P, Maschinot A, Liu C, Krishnan D (2020). "Sup ervised contrastive learning." Advances in Neural Information Processing Systems. 33: 18661–18673.

71. a, bChen T, Kornblith S, Norouzi M, Hinton G (2020). "A simple framework for contrastive learning of vi sual representations." In: International Conference on Machine Learning. PMLR. pp. 1597–1607.

72. a, b, cLi J, Zhang Y, Wang Z, Tu K (2021). "Probabilistic contrastive learning for domain adaptation." arX iv preprint arXiv:2111.06021. Available from: https://arxiv.org/abs/2111.06021.

73. ^Huo X, Xie L, Wei L, Zhang X, Chen X, Li H, Yang Z, Zhou W, Li H, Tian Q (2021). "Heterogeneous contr astive learning: Encoding spatial information for compact visual representations." IEEE Transactions on Multimedia. 24: 4224–4235.

74. ^Zhang Y, Zhang X, Li J, Qiu R, Xu H, Tian Q (2022). "Semi-supervised contrastive learning with similar ity co-calibration." IEEE Transactions on Multimedia. 2022.

75. a, bHuang X, Zhu C, Zhang B, Zhang S (2024). "Learning from different samples: A source-free framewo rk for semi-supervised domain adaptation."

76. ^Yang F, Wu K, Zhang S, Jiang G, Liu Y, Zheng F, Zhang W, Wang C, Zeng L. "Class-aware contrastive se mi-supervised learning." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern R ecognition. 2022. p. 14421-14430.

77. ^He K, Fan H, Wu Y, Xie S, Girshick R (2020). "Momentum contrast for unsupervised visual representati on learning." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9729–9738.

78. a, bBerthelot D, Carlini N, Goodfellow I, Papernot N, Oliver A, Raffel CA (2019). "Mixmatch: A holistic ap proach to semi-supervised learning." Advances in Neural Information Processing Systems. 32.

79. ^Zhang L, Deng Z, Kawaguchi K, Ghorbani A, Zou J (2020). "How does mixup help with robustness and generalization?" arXiv preprint arXiv:2010.04819. Available from: https://arxiv.org/abs/2010.04819.

80. ^Carratino L, Cissé M, Jenatton R, Vert J-P (2022). "On mixup regularization." The Journal of Machine L earning Research. 23 (1): 14632–14662.

81. ^Papyan V, Han X, Donoho DL (2020). "Prevalence of neural collapse during the terminal phase of deep learning training." Proceedings of the National Academy of Sciences. 117 (40): 24652–24663.

82. ^Ding Y, Sheng L, Liang J, Zheng A, He R (2023). "Proxymix: Proxy-based mixup training with label refi nery for source-free domain adaptation." Neural Networks. 167: 92–103.

83. ^Bai Y, Yang E, Han B, Yang Y, Li J, Mao Y, Niu G, Liu T (2021). "Understanding and improving early sto pping for learning with noisy labels." Advances in Neural Information Processing Systems. 34: 24392–2 4403.

84. ^Song H, Kim M, Park D, Lee JG (2019). "Prestopping: How does early stopping help generalization agai nst label noise?"

85. a, b, c, d, e, fGrandvalet Y, Bengio Y (2004). "Semi-supervised learning by entropy minimization." Advan ces in Neural Information Processing Systems. 17.

86. a, b, c, dSingh A. "Clda: Contrastive learning for semi-supervised domain adaptation." In: Ranzato M, Be ygelzimer A, Dauphin Y, Liang P, Vaughan JW, editors. Advances in Neural Information Processing Syste ms. Vol. 34. Curran Associates, Inc.; 2021. p. 5089-5101.

87. ^Peng X, Bai Q, Xia X, Huang Z, Saenko K, Wang B (2019). "Moment matching for multi-source domain adaptation." In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1406–1415.

88. ^Simonyan K, Zisserman A (2014). "Very deep convolutional networks for large-scale image recognitio n." arXiv preprint arXiv:1409.1556. Available from: https://arxiv.org/abs/1409.1556.

89. ^Cubuk ED, Zoph B, Shlens J, Le QV (2020). "Randaugment: Practical automated data augmentation wi th a reduced search space." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 702–703.

90. ^Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, et a l. (2019). "Pytorch: An imperative style, high-performance deep learning library." Advances in Neural I nformation Processing Systems. 32.

91. ^Chen X, Wang S, Long M, Wang J (2019). "Transferability vs. discriminability: Batch spectral penalizati on for adversarial domain adaptation." In: International Conference on Machine Learning. PMLR. pp. 1 081–1090.

92. ^Xie B, Li S, Lv F, Liu CH, Wang G, Wu D (2022). "A collaborative alignment framework of transferable k nowledge extraction for unsupervised domain adaptation." IEEE Transactions on Knowledge and Data Engineering.

93. a, bVan der Maaten L, Hinton G (2008). "Visualizing data using t-sne." Journal of Machine Learning Re search. 9 (11).

94. a, b *Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017). "Grad-cam: Visual explanations from deep networks via gradient-based localization." In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626.*

Declarations

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.