Prototype-Driven Active Domain Adaptation with Density Consideration

Zeyu Zhang[1,2], Chun Shen[1,2,3], Qiang Ma[1,3], Meng Kang[1,2], Shuai Lü[1,2,3*]

[1]Key Laboratory of Symbolic Computation and Knowledge Engineering (Jilin University), Ministry of Education, China
[2]College of Computer Science and Technology, Jilin University, China
[3]College of Software, Jilin University, China
{lus, shenchun}@jlu.edu.cn, {zeyuz22, maqiang21, kangmeng20}@mails.jlu.edu.cn

Abstract

Active domain adaptation (ADA) aims to select a small set of target samples for annotation and use them for training to maximally boost the adaptation performance. However, most existing ADA methods only rely on the original output of the model, without considering the relationship between the source and target domain features, which may lead to selecting uninformative samples. In this paper, we propose an effective ADA framework: Prototype-Driven Active Domain Adaptation with density consideration (PDADA). It selects the most valuable target samples in the presence of domain shift through two criteria: Density-Conscious Domainness (DCD) and Prototype-Driven Informativeness (PDI). Furthermore, considering the class imbalance and cluster looseness issues in sample selection and domain adaptation, we develop a Class Balanced Expansion (CBE) algorithm and the Adversarial Active Domain Adaptation via Protecting Structured Information (AADA-PSI). Extensive experiments demonstrate that under the cooperation of the above components, PDADA outperforms previous methods on several challenging benchmarks and can be generalized to multi-source active domain adaptation setting.

Introduction

Domain adaptation (DA) effectively mitigates the dependence of deep learning on manually labeled data by transferring the knowledge learned in the labeled source domain to the unlabeled target domain. Although DA methods have recently achieved marvelous results, there is still a significant gap in their performance compared to supervised learning methods (Fu et al. 2021). Actually, annotating a small portion of target samples is possible. However, not all target samples have annotation values, such as simple samples or outliers. Due to the limited labeling budget, we hope to select the most valuable target samples for annotation. These samples, annotated by experts, can be used for training to maximize performance in the target domain.

Active learning (AL) aims to select the most valuable samples for annotation (Tang and Huang 2021; Zhan et al. 2021; Ren et al. 2022; Parvaneh et al. 2022). However, AL assumes that the unlabeled and labeled data are subject to the same distribution. Therefore, in the presence of domain shift, traditional AL methods may be ineffective. Therefore, active domain adaptation (ADA) has been proposed. However, most existing ADA methods simply rely on the original output of the model (Su et al. 2020; Xie et al. 2022), or clustering in the target domain (Prabhu et al. 2021) to select target samples, without considering the relationship between the features of the source and target domain, which can lead to selecting the uninformative samples or outliers from the target domain, as shown in Figure 1(a).

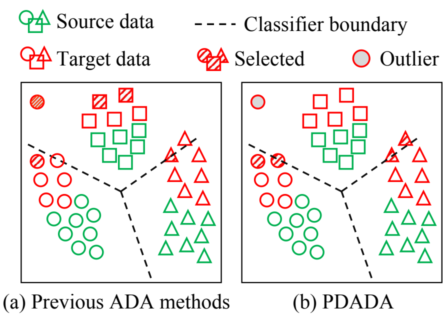


Figure 1: (a) Previous ADA methods may select uninformative samples or outliers from the target domain. (b) PDADA can select the most valuable target samples.

Thus motivated, we propose an effective ADA framework: Prototype-Driven Active Domain Adaptation with density consideration (PDADA). We first design a sample selection strategy that utilizes two criteria to measure the value of target samples: Density-Conscious Domainness (DCD) and Prototype-Driven Informativeness (PDI). DCD can not only identify target domain-specific samples but also avoid selecting outliers with the assistance of density-conscious weight. Then, PDI can further select informative samples that can maximally boost the adaptation performance. In summary, our sample selection strategy can select more valuable samples than existing ADA methods, as shown in Figure 1(b). Furthermore, considering the class imbalance issue during sample selection in previous ADA methods, we develop a Class Balanced Expansion (CBE) algorithm, which expands the labeled target domain in a class-balanced manner. Finally, in response to the cluster loose-

*Corresponding author.

ness issue during domain adaptation, we design the Adversarial Active Domain Adaptation via Protecting Structured Information (AADA-PSI). It incorporates the supervised information from the target domain during training and protects the structured information of the labeled data to alleviate the cluster looseness issue caused by adding labeled target samples to training.

In summary, we make the following contributions:

- We propose an effective ADA framework: Prototype-Driven Active Domain Adaptation with density consideration (PDADA). It can select a compact subset of target samples for annotation and leverages them during training to maximally enhance adaptation performance.
- We design a sample selection strategy that utilizes PDI and DCD criteria to select the most valuable target samples in the presence of domain shift.
- We further develop the CBE algorithm and AADA-PSI to alleviate the class imbalance and cluster looseness issues in sample selection and domain adaptation respectively.
- PDADA significantly outperforms previous ADA methods on multiple benchmarks. Further analysis verifies that PDADA is an effective ADA method and can be generalized to different domain adaptation settings.

Related Work

Domain Adaptation

Most previous methods follow the unsupervised domain adaptation (UDA) paradigm, which assumes that the target domain is completely unlabeled. Methods based on discrepancy metric minimization (Long et al. 2015, 2017) use statistical measures to measure the distribution discrepancy between the source and target domains, and then minimize the distribution discrepancy metrics to address the domain shift. While the methods based on adversarial learning utilize the idea of adversarial training in GAN, utilizing the adversarial between feature extractor and domain discriminator (Ganin and Lempitsky 2015; Long et al. 2018) or bi-classifier (Saito et al. 2018; Li et al. 2021) to learn domain invariant features. Recently proposed semi-supervised domain adaptation (SSDA) (Ao, Li, and Ling 2017; Saito et al. 2019; Jiang et al. 2020; Qiao et al. 2023) allows access to a few labeled target data, but all labeled target data are passively given at the beginning and fixed.

Active Learning

Active learning (AL) aims to select the most valuable samples for annotation (Tang and Huang 2021; Zhan et al. 2021; Ren et al. 2022; Parvaneh et al. 2022). The commonly used sample selection strategies are uncertainty sampling (Wang and Shang 2014; Lewis and Catlett 1994), query-by-committee (Yan et al. 2011), diversity sampling (Sener and Savarese 2018). However, AL assumes that the unlabeled and labeled data are subject to the same distribution. Therefore, in the presence of domain shift, AL may be ineffective.

The closest method to the PDI criterion proposed in this paper is ALFA-Mix (Parvaneh et al. 2022). ALFA-Mix (Parvaneh et al. 2022) proposes to form linear combinations of an unlabelled instance and labeled ones features and devises a method to calculate the interpolation ratio automatically to select the unlabelled samples that have novel features to learn from. Different from it, our PDI criterion considers mixing the target sample features with the source domain prototype features in the presence of domain shift with a fixed mixing ratio, and utilizes the degree of discrepancy in predicted probability distribution before and after mixing to select the most informative target samples. More importantly, PDI also considering the issue that the classifier trained on the source domain may be more sensitive to certain categories or features.

Active Domain Adaptation

Active domain adaptation (ADA) selects the most valuable target samples in the presence of domain shift to further improve the adaptation performance. AADA (Su et al. 2020) uses the output of domain discriminator and the entropy of prediction probability to select target samples. CLUE (Prabhu et al. 2021) proposes uncertainty-weighted clustering, which jointly considers the uncertainty and diversity of target samples. S3VAADA (Rangwani et al. 2021) uses a submodular strategy to select target samples. TQS (Fu et al. 2021) jointly uses transferable committee, transferable uncertainty, and transferable domainness criteria. DBAL (de Mathelin et al. 2022) designs the sample selection strategy under the assumption of Lipschitz function. SDM-AG (Xie et al. 2022) focuses on hard samples, and designs the maximum margin loss and margin sampling. TL-ADA (Han et al. 2023) addresses the transferable issue and low diversity of the existing loss-based sample selection method (Yoo and Kweon 2019) under the domain shift.

However, the above methods only rely on the original output of the model, without considering the relationship between the source and target domain features and may select valueless target samples. In contrast, our PDADA uses PDI and DCD criteria to alleviate the above issue, and it can also alleviate the class imbalance issue during sample selection and the cluster looseness issue during domain adaptation.

Methodology

In this section, we first present the definition of ADA and associated notations, and then introduce the components of the proposed Prototype-Driven Active Domain Adaptation with density consideration (PDADA) in detail. An outline of our PDADA framework is illustrated in Figure 2.

Problem Statement

In ADA, there is a source domain $\mathcal{D}_S = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ with n_s labeled data and a target domain $\mathcal{D}_{UT} = \{x_i^{ut}\}_{i=1}^{n_{ut}}$ with n_{ut} unlabeled data, where $\mathcal{D}_S$ and $\mathcal{D}_{UT}$ are sampled from two different distributions P and Q, respectively. In ADA, sample selection is a reciprocating process in which target samples are selected in $|R|$ rounds, with n_r samples selected in each round. The labeled target samples will be removed from $\mathcal{D}_{UT}$ and form a labeled target domain $\mathcal{D}_{LT} = \{(x_i^{lt}, y_i^{lt})\}_{i=1}^{n_{lt}}$, where $n_{lt} = |R| * n_r$ is the labeling

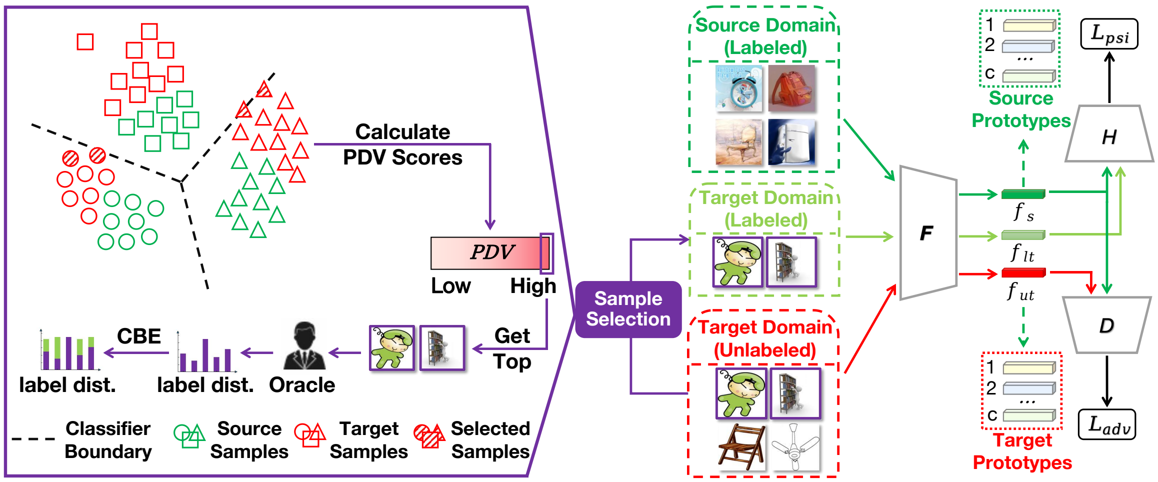


Figure 2: Illustration of PDADA framework. The sample selection process (left) utilizes Prototype-driven Density-conscious Value (PDV) and Class Balanced Expansion (CBE) algorithm to select the most valuable target samples in a class-balanced manner. The training process (right) utilizes all queried data for Adversarial Active Domain Adaptation via Protecting Structured Information (AADA-PSI). These two processes alternate until the labeling budget is exhausted.

budget. The network structure of PDADA includes a feature extractor F, a classifier H, and a domain discriminator D, which are parameterized by θ_f, θ_h and θ_d respectively. Given an image x, $f = F(x) \in \mathbb{R}^d$ is the feature extracted by F, where d is the feature dimension; $p = H(f) \in \mathbb{R}^c$ is the prediction probability, where c is the number of categories. The domain discriminator D is used to determine whether x comes from the source or target domain.

Prototype-Driven Density-Conscious Sample Selection Strategy

To select the most valuable target samples in the presence of domain shift, we propose the prototype-driven density-conscious sample selection strategy that includes two criteria: Prototype-Driven Informativeness (PDI) and Density-Conscious Domainness (DCD).

Density-conscious domainness. Density-conscious domainness criterion aims to select the samples with strong domainness, which means these samples are target domain-specific. These samples contain the knowledge that can not be learned from the source domain. Following (Su et al. 2020), for an unlabeled target sample x^{ut}, its domainness score can be measured as:

$$\mathrm{DOM}(x^{ut}) = \frac{1 - D(F(x^{ut}))}{D(F(x^{ut}))}. \quad (1)$$

The gap in domainness score for target domain-specific samples is not significant, which makes it difficult to select the target samples with the strongest domainness, resulting in a suboptimal sample selection strategy. Based on the observation that samples with stronger domainness tend to have neighbors with similarly strong domainness, we define the samples that are close to x^{ut} in the feature space as its neighbors, and utilize the domainness scores of the K nearest neighbors of x^{ut} to further enlarge this gap. The enhanced domainness score of x^{ut} can be formulated as:

$$\mathrm{DOM}(x^{ut}) \leftarrow \mathrm{DOM}(x^{ut}) + \frac{1}{K}\sum_{i=1}^{K} \mathrm{DOM}(x_i^{nn}), \quad (2)$$

where $\{x_i^{nn}\}_{i=1}^{K}$ are K nearest neighbors of x^{ut}.

Outliers also contain target domain-specific knowledge, that is, samples with large DOM scores may be outliers. Training with annotated outliers will damage the performance and waste the valuable labeling budget. We propose density-conscious weight to this issue. For each class in the target domain, we attain its prototype. Denote $T \in \mathbb{R}^{c \times d}$ as the global prototypes, which can be formulated as:

$$T = [t_j]_{j=1}^{c},\ t_j = \frac{\sum_{i=1}^{n_{ut}} H(F(x_i^{ut}))F(x_i^{ut})}{\sum_{i=1}^{n_{ut}} H(F(x_i^{ut}))}, \quad (3)$$

where $t_j \in \mathbb{R}^d$ is the prototype of j-th class. The prototypes only participate in sample selection and do not participate in training. Therefore, we only need to calculate prototypes once before sample selection. It is noteworthy that since the number of normal target samples is much greater than outliers, these normal samples will dominate the calculation of prototype, so the impact of outliers can be ignored. The outliers are generally far from the target domain prototypes, thus we consider the similarity between candidate samples and prototypes. The density-conscious weight of x^{ut} can be formulated as:

$$\mathrm{DEN}(x^{ut}) = \max_{j=1}^{c} Sim(F(x^{ut}), t_j), \quad (4)$$

where $Sim(.,.)$ is the cosine similarity. Finally, we integrate the enhanced domainness score $\mathrm{DOM}(x^{ut})$ and density-conscious weight $\mathrm{DEN}(x^{ut})$ into a unified density-conscious domainness score:

$$\mathrm{DCD}(x^{ut}) = \mathrm{DEN}(x^{ut}) \cdot \mathrm{DOM}(x^{ut}). \tag{5}$$

Prototype-driven informativeness. Samples with high DCD scores (i.e. strong domainness), are not always valuable. Some of them have been correctly classified, as shown by the red squares selected in Figure 1(a). Such samples can not provide valuable information to further improve the adaptation performance. To further select the most informative target samples, we propose the prototype-driven informativeness criterion. For each class in the source domain, we attain its global prototype $S = [s_j]_{j=1}^{c}$ similar to Eq. (3).

Then, we mix the target sample features with the source domain prototype features of each class separately and use the consistency of model predictions of mixed and original features to select the most informative target samples. The mixed features are formulated as:

$$f_j^{mix}(x^{ut}) = \lambda s_j + (1-\lambda)F(x^{ut}),\ j = 1, \dots, c, \tag{6}$$

where $\lambda \in (0, 1)$ is the feature mixing ratio. If the classification results of the mixed features and original feature are inconsistent, that is, $\exists j \in \{1, \dots, c\}$, $\arg\max_k H(F(x^{ut}))_k \neq \arg\max_k H(f_j^{mix}(x^{ut}))_k$, the sample is considered as a potential informative sample. Using the degree of discrepancy in predicted probability distribution before and after mixing with the prototype features of different classes as the prototype-driven informativeness score for x^{ut}:

$$\mathrm{PDI}(x^{ut}) = \frac{1}{c}\sum_{j=1}^{c} Div(H(F(x^{ut})), H(f_j^{mix}(x^{ut}))), \tag{7}$$

where $Div(p, q) = \mathbb{E}_i|p_i - q_i|$ is used to measure the divergence between two prediction probabilities. It should be noted that, since the classifier trained on the source domain may be more sensitive to certain categories or features, the traditional product-based prediction probability may lead to selecting pseudo-informative target samples. Therefore, we utilize the similarity-based prediction probability:

$$H(f) = \mathrm{softmax}([Sim(f, w_1), \dots, Sim(f, w_c)]), \tag{8}$$

where $w_j \in \mathbb{R}^d$ $(j = 1, \dots, c)$ is the j-th component of the classifier weights.

Prototype-driven density-conscious value. We integrate $\mathrm{DCD}(x^{ut})$ and $\mathrm{PDI}(x^{ut})$ into a unified prototype-driven density-conscious value to jointly consider the domainness and informativeness of target samples:

$$\mathrm{PDV}(x^{ut}) = \mathrm{DCD}(x^{ut}) + \mathrm{PDI}(x^{ut}), \tag{9}$$

where $\mathrm{DCD}(x^{ut})$ and $\mathrm{PDI}(x^{ut})$ are normalized using Min-Max Normalization, and we do not impose trade-off between the two criteria. Samples in the top n_r of PDV are selected for annotation. The labeled target samples will be added to $\mathcal{D}_{LT}$ and removed from $\mathcal{D}_{UT}$. The above sample selection process will alternate until the labeling budget is exhausted.

Class Balanced Expansion in $\mathcal{D}_{LT}$

In ADA, the labeling budget in each round is very small (e.g. 1%), and the model will be biased towards frequent classes, which will lead to a class imbalance in $\mathcal{D}_{LT}$. The class imbalance issue will continue to worsen with sample selection and model training. Therefore, we propose a Class Balanced Expansion (CBE) algorithm. After a round of sample selection, the initial number of samples in each class within $\mathcal{D}_{LT}$ is calculated as:

$$\mathcal{N}[j] = \sum_{(x^{lt}, y^{lt}) \in \mathcal{D}_{LT}} \mathbb{1}[y^{lt} = j],\ j = 1, \dots, c. \tag{10}$$

Target samples with unchanged classification results before and after feature mixing and the confidence of prediction also exceeds the threshold are considered as reliable pseudo-labeled samples. When there are insufficient samples of a particular class in $\mathcal{D}_{LT}$, we expand $\mathcal{D}_{LT}$ using the above reliable samples whose pseudo-labels belong to that class. The samples used to expand $\mathcal{D}_{LT}$ constitute $\mathcal{D}_{EXP}$:

$$\begin{aligned}\mathcal{D}_{EXP} = \{x \in \mathcal{D}_{UT} \mid p[\hat{y}] \geq \tau \wedge \mathcal{N}[\hat{y}] < \mathcal{N}^{max} \wedge \\ \forall j \in \{1, \dots, c\}(\hat{y}_j = \hat{y})\},\end{aligned} \tag{11}$$

where $p = H(F(x))$, $\hat{y} = \arg\max_k p_k$ is the pseudo label, τ is the confidence threshold, $\mathcal{N}^{max} = \max_{j=1}^{c} \mathcal{N}[j]$ is the number of samples in the most frequent class, and $\hat{y}_j = \arg\max_k H(f_j^{mix}(x))_k$ is the pseudo label for mixed feature $f_j^{mix}(x)$. Every time a sample with pseudo label $\hat{y}$ is added to $\mathcal{D}_{EXP}$, $N[\hat{y}]$ will increase by 1. Then, we expand $\mathcal{D}_{LT}$ in a class-balanced manner by merging $\mathcal{D}_{LT}$ and $\mathcal{D}_{EXP}$. And $\mathcal{D}_{EXP}$ will be refactored before each training epoch to avoid the accumulation of pseudo label errors.

Adversarial Active Domain Adaptation via Protecting Structured Information

Previous ADA methods may suffer cluster looseness issue when using the labeled target samples for domain adaptation. This is because adding labeled target samples leads to a more diverse of samples within each class, resulting in an increase in within-class scatter. Motivated by this, we propose Adversarial Active Domain Adaptation via Protecting Structured Information (AADA-PSI). AADA-PSI uses the similarity-based prediction probability to calculate the cross entropy loss L_{ce} of labeled data to encourage all features of the same class to be better clustered with corresponding classifier weight. Therefore, the intra-class difference of labeled data can be reduced, the inter-class difference can be increased, that is, the structured information of labeled data can be protected. The supervised loss function used in AADA-PSI is:

$$\begin{aligned}L_{psi} = &\mathbb{E}_{(x_i^s, y_i^s) \sim \mathcal{D}_S}[L_{ce}(y_i^s, H(F(x_i^s)))] + \\ &\mathbb{E}_{(x_j^{lt}, y_j^{lt}) \sim \mathcal{D}_{LT} \cup \mathcal{D}_{EXP}}[L_{ce}(y_j^{lt}, H(F(x_j^{lt})))].\end{aligned} \tag{12}$$

Since the target samples selected in PDADA have strong domainness, utilizing these samples for adversarial domain

Algorithm 1: PDADA

Input: Labeled source data $\mathcal{D}_S$, unlabeled target data $\mathcal{D}_{UT}$, labeled target data $\mathcal{D}_{LT} = \emptyset$ and expand pseudo labeled target data $\mathcal{D}_{EXP} = \emptyset$, epochs M, selection rounds R, labeling budget for each round n_r, mixing ratio λ, neighborhood size K, confidence threshold τ

for $m = 1$ to M **do**
  $\forall x \in \mathcal{D}_{UT}$, calculate $\mathrm{PDV}(x)$ with Eq. (9)
  if $m \in R$ **then**
    $\mathcal{D}_q \leftarrow$ select n_r of PDV with the highest values as active samples for annotation
    $\mathcal{D}_{LT} \leftarrow \mathcal{D}_{LT} \cup \mathcal{D}_q, \mathcal{D}_{UT} \leftarrow \mathcal{D}_{UT} \setminus \mathcal{D}_q$
  end if
  Construct $\mathcal{D}_{EXP}$ with Eq. (11)
  Update F, H, D with Eq. (14)
end for

adaptation may damage the adaptation performance. Thus, we only utilize source and unlabeled target domain data for adversarial domain adaptation in AADA-PSI:

$$\begin{aligned} L_{adv} =& \mathbb{E}_{x_i^s \sim \mathcal{D}_S}[\log(D(F(x_i^s)))] + \\ & \mathbb{E}_{x_j^{ut} \sim \mathcal{D}_{UT}}[\log(1 - D(F(x_j^{ut})))]. \end{aligned} \tag{13}$$

Overall, the final objective can be stated as:

$$\min_{\theta_f, \theta_h} \max_{\theta_d} L_{psi} + L_{adv}. \tag{14}$$

AADA-PSI can not only alleviate the domain shift, but also assist DCD and PDI in selecting the most valuable samples from the remaining target samples. We summarize the proposed framework in Algorithm 1.

Experiments

Experiment Setup

Datasets. We evaluate PDADA on four datasets: *Office-31* (Saenko et al. 2010) contains 31 categories from three domains: Amazon (A), Webcam (W), DSLR (D). *Office-Home* (Venkateswara et al. 2017) contains 65 categories from four domains: Artistic (A), Clipart (C), Product (P), Real-World (R). *VisDA* (Peng et al. 2017) contains 12 categories from two domains: Syntatic (S) and Real (R). *MiniDomainNet* (Zhou et al. 2021) is a subset of DomainNet that maintains the complexity of DomainNet and reduces the requirements for computing resources. It contains 126 categories from four domains: Clipart (C), Painting (P), Real (R), Sketch (S).

Implementation details. To ensure the fairness of the experiment, similar to (Rangwani et al. 2021; Han et al. 2023), we use ResNet-50 (He et al. 2016) which pre-trained on ImageNet (Krizhevsky, Sutskever, and Hinton 2017) as backbone network. We use SGD optimizer (Bottou 2010) with momentum 0.9, weight decay 5e-4, and learning rate 0.01. We set the labeling budget to 5%, the sample selection rounds $|R|$ to 5, the mixing ratio λ in Eq. (6) to 0.1, and the confidence threshold τ in Eq. (11) to 0.9 for all datasets. The neighborhood size K in Eq. (2) is 9 for Office-31, 20 for Office-Home, 5 for VisDA and MiniDomainNet. The batch size is 64 for VisDA and 32 for other datasets. In each adaptation scenario, we report the average accuracy over 3 trials. More implementation details can be seen in the Appendix. Code is available at https://github.com/jluzeyuz/PDADA.

Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg
Source Only	81.5	75.0	63.1	95.2	65.7	99.4	80.0
Random	87.1	84.1	75.5	98.1	75.8	99.6	86.7
Entropy	91.0	89.2	76.1	99.7	77.7	**100.0**	88.9
CoreSet	82.5	81.1	70.3	96.5	72.4	99.6	83.7
BADGE	90.8	89.1	79.8	99.6	79.6	**100.0**	89.8
AADA	89.2	87.3	78.2	99.5	78.7	**100.0**	88.8
CLUE	92.0	87.3	79.0	99.2	79.6	99.8	89.5
S3VAADA	93.0	93.7	75.9	99.4	78.2	**100.0**	90.0
TQS	92.8	92.2	80.6	**100.0**	80.4	**100.0**	91.1
DBAL	88.2	88.9	75.2	99.4	77.0	**100.0**	88.1
SDM-AG	94.8	93.5	81.9	**100.0**	81.9	**100.0**	92.0
TL-ADA	**96.6**	96.8	79.9	99.8	81.7	99.8	92.2
PDADA	95.4	**97.0**	**83.2**	**100.0**	**84.6**	**100.0**	**93.4**

Table 1: Comparison results (Accuracy: %) on Office-31 with 5% labeling budget.

Comparative Results

We compare PDADA with two baseline methods, including Source Only (ResNet-50 trained with source data only) and Random (samples are selected randomly from the unlabeled target domain); several traditional AL methods, including Entropy (Wang and Shang 2014), CoreSet (Sener and Savarese 2018), and BADGE (Ash et al. 2020); and several ADA methods, including AADA (Su et al. 2020), CLUE (Prabhu et al. 2021), S3VAAD (Rangwani et al. 2021), TQS (Fu et al. 2021), DBAL (de Mathelin et al. 2022), SDM-AG (Xie et al. 2022), and TL-ADA (Han et al. 2023).

Office-31. Results on Office-31 are shown in Table 1. We can observe that PDADA outperforms the existing AL and ADA methods on most tasks and average accuracy. Especially for some difficult tasks, PDADA achieves +1.3% improvement in D→A and +2.7% in W→A.

Office-Home. Results on Office-Home are shown in Table 2. Compared with SDM-AG (Xie et al. 2022) and TL-ADA (Han et al. 2023), PDADA achieves +5.1% and +2.6% improvement in average accuracy, respectively. Specifically, PDADA outperforms existing methods on some difficult adaptation scenarios, such as A→C and P→C. It demonstrates that PDADA is effective in ADA issues and suitable for tasks with large domain shift.

VisDA. As shown in the first column of Table 2, PDADA is still significantly superior to existing AL and ADA methods on VisDA. It demonstrates that PDADA is also suitable for large-scale datasets.

MiniDomainNet. Results on MiniDomainNet further demonstrate the effectiveness of PDADA. As shown in Ta-

Method	VisDA	Office-Home												
	Syn→Real	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg
Source Only	44.7	42.1	66.3	73.3	50.7	59.0	62.6	51.9	37.9	71.2	65.2	42.6	76.6	58.3
Random	78.1	52.5	74.3	77.4	56.3	69.7	68.9	57.7	50.9	75.8	70.0	54.6	81.3	65.8
Entropy	82.7	58.0	78.4	79.1	60.5	73.0	72.6	60.4	54.2	77.9	71.3	58.0	83.6	68.9
CoreSet	81.9	51.8	72.6	75.9	58.3	68.5	70.1	58.8	48.8	75.2	69.0	52.7	80.0	65.1
BADGE	84.3	58.2	79.7	79.9	61.5	74.6	72.9	61.5	56.0	78.3	71.4	60.9	84.2	69.9
AADA	80.8	56.6	78.1	79.0	58.5	73.7	71.0	60.1	53.1	77.0	70.6	57.0	84.5	68.3
CLUE	85.2	58.0	79.3	80.9	68.8	77.5	76.7	66.3	57.9	81.4	75.6	60.8	86.3	72.5
S3VAADA	77.7	57.3	73.9	76.6	60.3	76.5	71.1	57.6	56.0	78.7	71.4	63.1	83.3	68.8
TQS	83.1	58.6	81.1	81.5	61.1	76.1	73.3	61.2	54.7	79.7	73.4	58.9	86.1	70.5
DBAL	82.6	58.7	77.3	79.2	61.7	73.8	73.3	62.6	54.5	78.1	72.4	59.9	84.3	69.6
SDM-AG	80.3	61.2	82.2	82.7	66.1	77.9	76.1	66.1	58.4	81.0	76.0	62.5	87.0	73.1
TL-ADA	86.8	63.7	83.9	82.5	69.7	82.7	**81.4**	70.3	61.2	84.6	77.4	63.4	85.9	75.6
PDADA	**87.3**	**68.3**	**84.4**	**84.2**	**70.3**	**85.0**	81.2	**71.8**	**68.7**	**84.8**	**79.1**	**70.7**	**90.3**	**78.2**

Table 2: Comparison results (Accuracy: %) on VisDA and Office-Home with 5% labeling budget.

Method	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
Source Only	52.1	63.0	49.4	55.9	73.0	51.1	56.8	61.0	50.0	54.0	48.9	60.3	56.3
Random	61.6	78.7	61.6	64.0	78.7	63.7	60.5	64.3	61.1	64.8	58.7	75.2	66.1
AADA	62.4	77.5	61.7	61.9	79.7	61.1	65.6	66.0	60.8	65.1	62.1	80.0	67.0
CLUE	57.6	77.5	58.6	58.9	76.8	65.9	66.3	60.2	60.5	66.2	58.7	76.0	65.3
TQS	**67.8**	**82.0**	65.4	67.5	**84.8**	66.1	63.8	67.2	62.5	71.1	64.4	**81.6**	70.4
DBAL	62.9	79.2	60.8	64.6	78.1	62.5	65.6	65.2	59.2	66.3	61.3	80.3	67.2
PDADA	67.0	81.6	**69.6**	**72.4**	83.3	**67.0**	**78.3**	**72.7**	**70.7**	**77.5**	**66.9**	81.2	**74.0**

Table 3: Comparison results (Accuracy: %) on MiniDomainNet with 5% labeling budget.

ble 3, PDADA is superior to existing ADA methods on most tasks, especially for some difficult tasks: R→C and R→S. It demonstrates that PDADA is effective on large-scale dataset with significant domain shift.

Further Analysis

Ablation studies. To illustrate the effect of each component of PDADA, we conduct ablation studies on Office-Home, the results are shown in Table 4. When both PDI and DCD are removed, we use the random sample selection strategy instead. Removing PSI means not using similarity-based prediction probability to calculate the cross entropy loss and utilizing $\mathcal{D}_{LT}$ for adversarial domain adaptation. By comparing (g) with (f), the class imbalance in $\mathcal{D}_{LT}$ will damage performance, thereby in turn emphasizing the need for CBE. Note that to eliminate the influence of $\mathcal{D}_{EXP}$ and more accurately analyze the effectiveness of other components, we removed the class balance strategy in other variants. In short, PDADA outperforms all of the variants by a large margin on average accuracy, demonstrating the effectiveness of PDI, DCD, AADA-PSI, and CBE. More experiments and analyses can be seen in the Appendix.

	PDI	DCD	AADA-PSI	CBE	Office-Home
(a)					72.2
(b)			✓		74.5
(c)	✓	✓			76.0
(d)	✓		✓		77.2
(e)		✓	✓		73.8
(f)	✓	✓	✓		77.6
(g)	✓	✓	✓	✓	**78.2**

Table 4: Ablation study results on Office-Home.

Sensitivity to hyperparameter. To show the sensitivity of PDADA to the mixing ratio λ, neighborhood size K, and confidence threshold τ, we conduct experiments on VisDA. The results are shown in Figure 3, we can observe that PDADA is insensitive to λ in a reasonable range. When λ is abnormally small or large, it causes an inappropriate proportion of prototype features in mixed features, resulting in many uninformative target samples being selected. Therefore, the performance will decrease. Within a wide range of K, the performance of PDADA is stable. Besides, when τ is abnormally high or low, it causes insufficient samples with credible pseudo label or selects some samples with error pseudo label, resulting in damaged performance.

Variation with different labeling budget. We further analyze the performance on the target domain as the labeling budget gradually increases from 1% to 10%. The results are shown in Figure 4. We can observe that PDADA achieves the

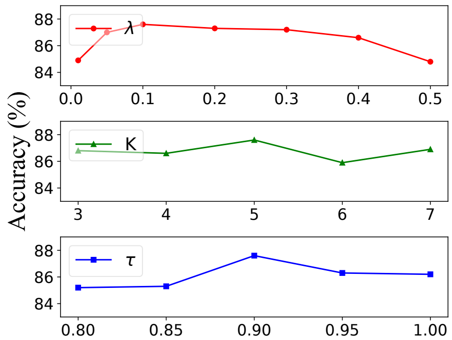
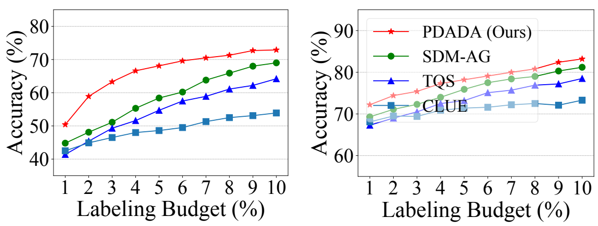


Figure 3: Hyper-parameter sensitivity analysis of λ, K and τ on VisDA with 5% labeling budget.



(a) P→C (b) R→A

Figure 4: Performance variation with different labeling budget (1% to 10%) on Office-Home.

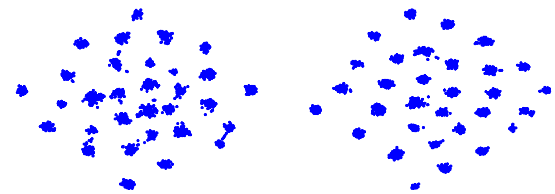
best results regardless of the labeling budget, and the performance of PDADA can increase steadily with the increase of the labeling budget. It demonstrates that PDADA is effective across a wide range of labeling budgets.

Expanding to Multi-source Active Domain Adaptation (MADA) Setting. In addition to using PDADA for standard setting, we also extend PDADA to MADA setting, where multiple source domains are available during training. For simplicity, when using PDADA for MADA setting, we only merge multiple source domains into one, while the sample selection and domain adaptation process remain unchanged. We conduct experiments on Office-31 and compare PDADA with the first and latest MADA method: D3AAMDA (Liu et al. 2023). The experimental results are shown in Table 5. We can observe that PDADA is superior to D3AAMDA without specifically designing for the MADA setting. It demonstrates that PDADA is a universal method that can be well generalized to other settings.

Effect of Protecting Structured Information (PSI). To further investigate the effect of PSI in adversarial active domain adaptation, we visualize the representations with and without PSI in task D→A on Office-31 by t-SNE in Figure 5. We can observe that the feature distribution when trained using PSI has smaller intra-class difference and larger inter-class difference. It demonstrates that PSI can effectively reduce the within-class scatter, which means that PSI can protect the structured information of data during training.

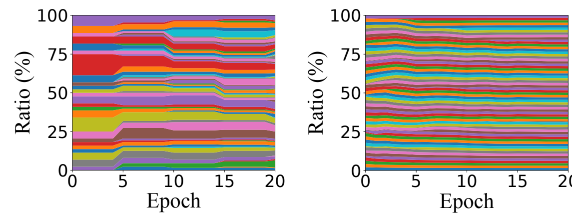
Method	D, W→A	A, W→D	A, D→W	Avg
D3AAMDA	81.3	**100.0**	**100.0**	93.8
PDADA	**83.9**	99.9	99.8	**94.5**

Table 5: Comparison results (Accuracy: %) on Office-31 with 5% labeling budget under MADA setting.



(a) PDADA w/o PSI (b) PDADA w/ PSI

Figure 5: T-SNE of representations with and without PSI in task D→A on Office-31.



(a) PDADA w/o CBE (b) PDADA w/ CBE

Figure 6: Ratio of samples per class in the labeled target domain with and without CBE in task P→C on Office-Home.

Effect of Class Balanced Expansion (CBE). To further investigate the effect of CBE in PDADA, we visualize the ratio of samples per class in the labeled target domain with and without CBE in task P→C on Office-Home. The results are shown in Figure 6, and we can observe that the labeled target domain used for training is dominated by major classes without using CBE. On the contrary, when CBE is used, the class distribution will be more balanced. Moreover, training the model in a class-balanced manner is more conducive to improving the adaptation performance, as can be seen from (g) outperforming (f) in Table 4.

Conclusion

In this paper, we introduce an effective active domain adaptation framework, named Prototype-Driven Active Domain Adaptation with density consideration (PDADA). We design a prototype-driven density-conscious sample selection strategy that utilizes PDI and DCD criteria to select the most valuable target samples in the presence of domain shift. In addition, to alleviate the class imbalance issue during sample selection and the cluster looseness issue during adversarial active domain adaptation, we develop a Class Balanced Expansion (CBE) algorithm and a domain adaptation strategy: AADA-PSI, respectively. Extensive experiments and analyses demonstrate the superiority of PDADA.

Acknowledgments

This work was supported by the Natural Science Research Foundation of Jilin Province of China under Grant No. 20220101106JC, and the National Natural Science Foundation of China under Grant No. 62407010.

References

Ao, S.; Li, X.; and Ling, C. X. 2017. Fast Generalized Distillation for Semi-Supervised Domain Adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2017)*, 1719–1725.

Ash, J. T.; Zhang, C.; Krishnamurthy, A.; Langford, J.; and Agarwal, A. 2020. Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds. In *Proceedings of the International Conference on Learning Representations (ICLR 2020)*.

Bottou, L. 2010. Large-Scale Machine Learning with Stochastic Gradient Descent. In *Proceedings of the International Conference on Computational Statistics (COMPSTAT 2010)*, 177–186.

de Mathelin, A.; Deheeger, F.; Mougeot, M.; and Vayatis, N. 2022. Discrepancy-Based Active Learning for Domain Adaptation. In *Proceedings of the International Conference on Learning Representations (ICLR 2022)*.

Fu, B.; Cao, Z.; Wang, J.; and Long, M. 2021. Transferable Query Selection for Active Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, 7272–7281.

Ganin, Y.; and Lempitsky, V. S. 2015. Unsupervised Domain Adaptation by Backpropagation. In *Proceedings of the International Conference on Machine Learning (ICML 2015)*, 1180–1189.

Han, K.; Kim, Y.; Han, D.; Lee, H.; and Hong, S. 2023. TL-ADA: Transferable Loss-based Active Domain Adaptation. *Neural Networks*, 161: 670–681.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, 770–778.

Jiang, P.; Wu, A.; Han, Y.; Shao, Y.; Qi, M.; and Li, B. 2020. Bidirectional Adversarial Training for Semi-Supervised Domain Adaptation. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2020)*, 934–940.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2017. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84–90.

Lewis, D. D.; and Catlett, J. 1994. Heterogeneous Uncertainty Sampling for Supervised Learning. In *Proceedings of the International Conference on Machine Learning (ICML 1994)*, 148–156.

Li, S.; Lv, F.; Xie, B.; Liu, C. H.; Liang, J.; and Qin, C. 2021. Bi-Classifier Determinacy Maximization for Unsupervised Domain Adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2021)*, 8455–8464.

Liu, L.; Zhou, B.; Zhao, Z.; and Liu, Z. 2023. Dynamic Domain Discrepancy Adjustment for Active Multi-Domain Adaptation. *arXiv preprint arXiv:2307.14068*.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. I. 2015. Learning Transferable Features with Deep Adaptation Networks. In *Proceedings of the International Conference on Machine Learning (ICML 2015)*, 97–105.

Long, M.; Cao, Z.; Wang, J.; and Jordan, M. I. 2018. Conditional Adversarial Domain Adaptation. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS 2018)*, 1647–1657.

Long, M.; Zhu, H.; Wang, J.; and Jordan, M. I. 2017. Deep Transfer Learning with Joint Adaptation Networks. In *Proceedings of the International Conference on Machine Learning (ICML 2017)*, 2208–2217.

Parvaneh, A.; Abbasnejad, E.; Teney, D.; Haffari, R.; van den Hengel, A.; and Shi, J. Q. 2022. Active Learning by Feature Mixing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, 12227–12236.

Peng, X.; Usman, B.; Kaushik, N.; Hoffman, J.; Wang, D.; and Saenko, K. 2017. VisDA: The Visual Domain Adaptation Challenge. *arXiv preprint arXiv:1710.06924*.

Prabhu, V.; Chandrasekaran, A.; Saenko, K.; and Hoffman, J. 2021. Active Domain Adaptation via Clustering Uncertainty-weighted Embeddings. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV 2021)*, 8485–8494.

Qiao, Z.; Luo, X.; Xiao, M.; Dong, H.; Zhou, Y.; and Xiong, H. 2023. Semi-supervised Domain Adaptation in Graph Transfer Learning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2023)*, 2279–2287.

Rangwani, H.; Jain, A.; Aithal, S. K.; and Babu, R. V. 2021. S3VAADA: Submodular Subset Selection for Virtual Adversarial Active Domain Adaptation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV 2021)*, 7496–7505.

Ren, P.; Xiao, Y.; Chang, X.; Huang, P.; Li, Z.; Gupta, B. B.; Chen, X.; and Wang, X. 2022. A Survey of Deep Active Learning. *ACM Computing Surveys*, 54(9): 180:1–180:40.

Saenko, K.; Kulis, B.; Fritz, M.; and Darrell, T. 2010. Adapting Visual Category Models to New Domains. In *European Conference on Computer Vision (ECCV 2010)*, 213–226.

Saito, K.; Kim, D.; Sclaroff, S.; Darrell, T.; and Saenko, K. 2019. Semi-Supervised Domain Adaptation via Minimax Entropy. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV 2019)*, 8049–8057.

Saito, K.; Watanabe, K.; Ushiku, Y.; and Harada, T. 2018. Maximum Classifier Discrepancy for Unsupervised Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, 3723–3732.

Sener, O.; and Savarese, S. 2018. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *Proceedings of the International Conference on Learning Representations (ICLR 2018)*.

Su, J.; Tsai, Y.; Sohn, K.; Liu, B.; Maji, S.; and Chandraker, M. 2020. Active Adversarial Domain Adaptation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV 2020)*, 728–737.

Tang, Y.; and Huang, S. 2021. Dual Active Learning for Both Model and Data Selection. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2021)*, 3052–3058.

Venkateswara, H.; Eusebio, J.; Chakraborty, S.; and Panchanathan, S. 2017. Deep Hashing Network for Unsupervised Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, 5385–5394.

Wang, D.; and Shang, Y. 2014. A new active labeling method for deep learning. In *International Joint Conference on Neural Networks (IJCNN 2014)*, 112–119.

Xie, M.; Li, Y.; Wang, Y.; Luo, Z.; Gan, Z.; Sun, Z.; Chi, M.; Wang, C.; and Wang, P. 2022. Learning Distinctive Margin toward Active Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, 7983–7992.

Yan, Y.; Rosales, R.; Fung, G.; and Dy, J. G. 2011. Active Learning from Crowds. In *Proceedings of the International Conference on Machine Learning (ICML 2011)*, 1161–1168.

Yoo, D.; and Kweon, I. S. 2019. Learning Loss for Active Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, 93–102.

Zhan, X.; Liu, H.; Li, Q.; and Chan, A. B. 2021. A Comparative Survey: Benchmarking for Pool-based Active Learning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2021)*, 4679–4686.

Zhou, K.; Yang, Y.; Qiao, Y.; and Xiang, T. 2021. Domain Adaptive Ensemble Learning. *IEEE Transactions on Image Processing*, 30: 8008–8018.