High-Level Semantic Feature Matters Few-Shot Unsupervised Domain Adaptation

Lei Yu[1], **Wanqi Yang**[1*], **Shengqi Huang**[1], **Lei Wang**[2], **Ming Yang**[1]

[1]School of Computer and Electronic Information, Nanjing Normal University, China
[2]School of Computing and Information Technology, University of Wollongong, Australia
yulei@njnu.edu.cn, yangwq@njnu.edu.cn, huangshengqi@njnu.edu.cn, leiw@uow.edu.au, myang@njnu.edu.cn

Abstract

In few-shot unsupervised domain adaptation (FS-UDA), most existing methods followed the few-shot learning (FSL) methods to leverage the low-level local features (learned from conventional convolutional models, *e.g.,* ResNet) for classification. However, the goal of FS-UDA and FSL are relevant yet distinct, since FS-UDA aims to classify the samples in target domain rather than source domain. We found that the local features are insufficient to FS-UDA, which could introduce noise or bias against classification, and not be used to effectively align the domains. To address the above issues, we aim to refine the local features to be more discriminative and relevant to classification. Thus, we propose a novel task-specific semantic feature learning method (TSECS) for FS-UDA. TSECS learns high-level semantic features for image-to-class similarity measurement. Based on the high-level features, we design a cross-domain self-training strategy to leverage the few labeled samples in source domain to build the classifier in target domain. In addition, we minimize the KL divergence of the high-level feature distributions between source and target domains to shorten the distance of the samples between the two domains. Extensive experiments on *DomainNet* show that the proposed method significantly outperforms SOTA methods in FS-UDA by a large margin (*i.e.*, $\sim 10\%$).

Introduction

Currently, a setting namely few-shot unsupervised domain adaptation (FS-UDA) (Huang et al. 2021)(Yang et al. 2022), which utilizes few labeled data in source domain to train a model to classify unlabeled data in target domain, owns its potential feasibility. Typically, a FS-UDA model could learn general knowledge from base classes during training to guide classification in novel classes during testing. It is known that both insufficient labels in source domain and large domain shift make FS-UDA as a challenging task.

Previous studies, *e.g.,* IMSE (Huang et al. 2021), first followed several few-shot learning (FSL) methods (Li et al. 2019)(Tzeng et al. 2017) to learn the local features by using convolutional models (*e.g.,* ResNet) and then leveraged them to learn image-to-class similarity pattern for classification. However, we wish to clarify that *the goal of FS-UDA and FSL are relevant yet distinct*, since both of them suffer from insufficient labeled training data whereas FS-UDA aims to classify the samples in target domain rather than source domain. As shown in Figure 1, by visualizing the spatial similarity of query images to predicted support classes, we found using local features causes the inaccurate regions of query images to match incorrect classes. This reason might be that few labeled samples and large domain shift between the support and query sets simultaneously result in the conventional local features in FSL to fail in classification. In this sense, the local features are insufficient to FS-UDA, which could introduce noise or bias against the classification in target domain and not be used to effectively align the domains.

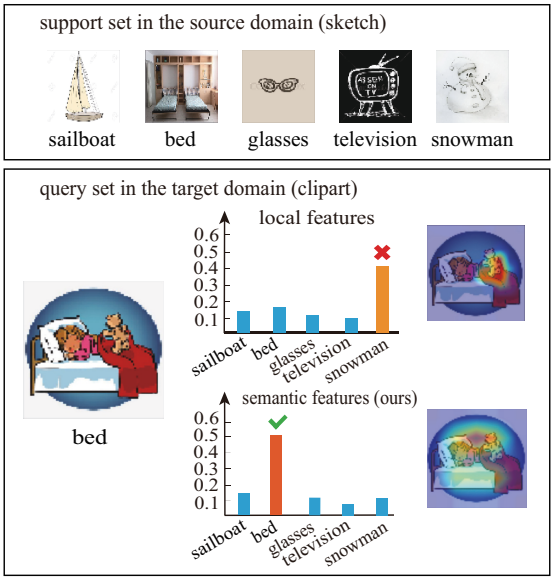

Figure 1: A 5-way 1-shot task for FS-UDA where the support set includes five classes and one sample for each class. The figure shows the similarity of query images to every support classes and the spatial similarity of query images to the predicted support class. We found using local features could cause some inaccurate regions of query images to match the incorrect classes, while our semantic features make the object region in query images similar with their true class, thus achieving correct classification.

*The corresponding author is Wanqi Yang.

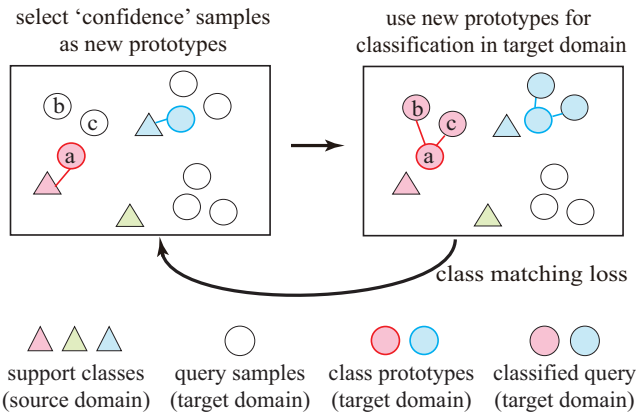

Figure 2: Illustration of the process for cross-domain self-training in TSECS. Different shapes represent different domains. We first select the 'confidence' target samples (*e.g.,* a) that are very similar to support classes, and then regard them as the new class prototypes to further classify the other target samples (*e.g.,* b, c). This process is executed iteratively with using class matching loss to narrow the distance of query images and their most similar support classes.

To address this issue, we aim to refine the low-level local features to be more discriminative and relevant to classification, *i.e., high-level semantic features*, and meanwhile align the semantic features for domain adaptation. Therefore, we propose a novel task-specific semantic feature method (TSECS) that learns the semantic features for each task by clustering the local features of support set and query set. To obtain the related semantics from previous tasks, the cluster centroids of the current task are then fused by cross-attention with that of the previous task to generate high-level semantic features to boost classification performance.

Moreover, for the domain shift between source and target domains, many domain adaptation methods (Saito et al. 2018)(Tzeng et al. 2017)(Tzeng et al. 2014) reduced the distribution discrepancy between domains by using a discriminator to adverse against feature embedding. However, this way could fail in aligning the samples of the same class between domains due to label missing in target domain, which could make the classes of two domains mismatched and thus affect the classification. Therefore, we aim to align the high-level semantic features by minimizing the KL divergence of the semantic feature distributions between domains, and meanwhile design a cross-domain self-training strategy to train the classifier in target domain.

We hypothesis that there are usually several 'confidence' samples in target domain that could be classified correctly by support set in source domain, in other words, they are very similar to their class prototypes in source domain. Meanwhile, the target domain samples in the same class are more similar to each other than that of other classes. Based on this, we regard these 'confidence' samples in the target domain as new prototypes of the classes, which replace those from the support set of source domain. As shown in Figure 2, several 'confidence' samples (*e.g.,* a) can be selected as prototypes of their similar classes for classification (*e.g.,* b and c) in target domain. Moreover, the process is conducted iteratively by using class matching loss for better domain alignment.

In sum, we propose the novel method, namely TSECS, for FS-UDA. It refines the local features of convolutional network to generate specific semantic features of each task, and meanwhile perform cross-domain self-training to transport labels from support set in the source domain to query set in the target domain to effectively classify the samples in target domain. Our contributions can be summarized as:

(1) A novel solution for FS-UDA. TSECS aims to learn high-level semantic features for classification and domain alignment, which could be regarded as a more effective and efficient way than using local features.

(2) Task-specific semantic embedding for few-shot setting. It can be seamlessly add to existing FSL/FS-UDA models, which could alleviate the bias of classification.

(3) Cross-domain self-training for domain alignment. It is designed to bring the samples of the same class close, which could guide effective domain alignment.

We conduct extensive experiments on *DomainNet*. Our method significantly outperforms SOTA methods in FS-UDA by a large margin up to $\sim 10\%$.

Related Works

Unsupervised domain adaptation. The conventional UDA methods aim to reduce discrepancy between source domain and target domain in the feature space and utilize sufficiently labeled source domain data to classify data from target domain. The difference between unsupervised domain adaptation methods often lies in the evaluation of domain discrepancy and the objective function of model training. Several researchers (Long et al. 2015)(Tzeng et al. 2014) minimize the feature discrepancy by using maximum mean discrepancy to measure the discrepancy between the distribution of domains. Moreover, adversarial training (Tzeng et al. 2017)(Ganin et al. 2016) to learn domain-invariant features is usually used to tackle domain shift. Several methods (Tang, Chen, and Jia 2020)(Zou et al. 2019)(Zou et al. 2018)(Kim et al. 2021)train the classifier in both source domain and target domain and utilize pseudo-labels from target domain to calculate classification loss. Overall, these UDA methods all require sufficiently labeled source domain data to realize domain alignment and classification, but they perform poor when labeled source domain data are insufficient.

Few-shot learning. Few-shot learning has two main streams, metric-based and optimization-based approaches. Optimization-based methods (Bertinetto et al. 2019)(Finn, Abbeel, and Levine 2017)(Ravi and Larochelle 2017) usually train a meta learner over auxiliary dataset to learn a general initialization model, which can fine-tune and adapt to new tasks very soon. The main purpose of metric-based methods (Li et al. 2019)(Snell, Swersky, and Zemel 2017)(Vinyals et al. 2016)(Ye et al. 2020) is that learn a generalizable feature embedding for metric learning, which can immediately adapt to new tasks without any fine-tune and retraining. Typically, ProtoNet (Snell, Swersky, and Zemel

2017) learns the class prototypes in the support set and classifies the query images based on the maximum similarity to these prototypes. Other than these metric-based methods on feature maps, many methods on local features have appeared. DN4 (Li et al. 2019) utilizes large amount of local features to measure the similarity between support and query sets instead of flattening the feature map into a long vector. Based on local features, DeepEMD (Zhang et al. 2020) adopts Earth Mover's Distance distance to measure the relationship between query and support sets. Furthermore, a few recent works focus on the issue of cross-domain FSL in which domain shift exists between data of meta tasks and new tasks. The baseline models (Chen et al. 2019) are used to do cross-domain FSL. LFT (Tseng et al. 2020) performs adaptive feature transformation to tackle the domain shift.

Few-shot unsupervised domain adaptation. Compared with UDA, FS-UDA is to deal with many UDA tasks by leveraging few labeled source domain samples for each. And compared with cross-domain FSL, FS-UDA are capable of handling the circumstances of no available labels in the target domain, and large domain gap between the support and query sets in every task. For the one-shot UDA (Luo et al. 2020), it deals with the case that only one unlabeled target sample is available, but does not require the source domain to be few-shot, which is different from ours. Recently, there are a few attempts in FS-UDA. PCS (Yue et al. 2021) performs prototype self-supervised learning in cross-domain, but they require enough unlabeled source samples to learn prototypes and ignore task-level transfer, which is also different from ours. meta-FUDA (Yang et al. 2022) leverages meta learning-based optimization to perform task-level transfer and domain-level transfer jointly. IMSE (Huang et al. 2021) utilizes local features to learn similarity patterns for cross-domain similarity measurement. However, they did not consider that local features could bring the noise or bias to affect classification and domain alignment. Thus, we propose task-specific semantic features to solve this problem.

Methodology

Problem Definition

A N-way, K-shot FS-UDA task. The FS-UDA setting includes two domains: a source domain S and a target domain T. A N-way, K-shot FS-UDA task includes a support set X_S from S and a query set Q_T from T. The support set X_S contains N classes and K samples per class in the source domain. The query set Q_T contains the same N classes as in X_S and N_q target domain samples per class. To classify query images in Q_T to the correct class in X_S, it is popular to train a general model from base classes to adapt to handle new N-way, K-shot FS-UDA tasks for testing.

Auxiliary dataset and episodic training. As in (Huang et al. 2021), the base classes are collected from an auxiliary dataset D^{aux} to perform episodic training to learn the general model. Note that the base classes in D^{aux} are completely different from new classes in testing tasks, which are unseen during episodic training. Moreover, D^{aux} includes labeled source domain data and unlabeled target domain data for FS-UDA.

The flowchart of our method. Figure 3 illustrates our method for 5-way, 1-shot FS-UDA tasks. In each episode, a support set (X_S) and two query sets (Q_S and Q_T) are first through the convolution network (*e.g., ResNet*) to extract their local features. Then, the task-specific semantic embedding module refines the local features to generate semantic features, which is computational efficient due to dimension reduction. Also, based on semantic features of Q_S and Q_T, we leverage their similarity patterns (Huang et al. 2021) to calculate *image-to-class* similarity for classification with the loss $\mathcal{L}_{cls}$. To improve its performance, cross-domain self-training module is performed to introduce the class prototypes of target domain and train a target domain classifier with a class matching loss $\mathcal{L}_{clm}$. In addition, the semantic features and similarity patterns from both domains are further aligned by calculating their alignment losses $\mathcal{L}_{sfa}$ and $\mathcal{L}_{spa}$, respectively. Finally, the losses above are back-propagated to update our model. After episodic training over all episodes, we utilize the learned model to test new FS-UDA tasks. Then, we calculate the averaged classification accuracy on these tasks for performance evaluation.

Task-specific Semantic Feature Learning

Most FSL methods and FS-UDA methods learned local features from convolutional networks for classification. However, we found that the local features could introduce noise or bias that is valid for classification and domain alignment. Thus, we aim to refine the local features to generate high-level semantic features for each task. In the following, we will introduce our semantic feature embedding module.

First of all, in each episode, all local features $L \in \mathbb{R}^{(|X_S|+|Q_S|+|Q_T|)HW\times d}$ are extracted from the convolutional network, where $|\cdot|$ is the number of samples in a set, and H, W and d are the height, width, and channel of the feature map, respectively. Then, we cluster the local features to generate different semantic clusters for support set and query set, respectively, since clustering the two sets together could result in the clusters that relate to the domains due to the presence of large domain gap. For simplification, we adopt K-means for clustering, and meanwhile utilize the singular value decomposition (SVD) to adaptively take the number of eigenvalues greater than a certain threshold as the cluster number k ($k \ll d$) for each task. Afterwards, we calculate the task-specific semantic feature map $F \in \mathbb{R}^{(|X_S|+|Q_S|+|Q_T|)HW\times k}$ by measuring the *Cosine* similarity between the local features L and the centroids $C \in \mathbb{R}^{k\times d}$ of all semantic clusters, *i.e.,*

$$F = \frac{L}{||L||_2} \cdot \frac{C^\top}{||C||_2}. \tag{1}$$

Finally, we split F to 2×2 blocks based on height and weight dimension of the feature map, and then concatenate the four blocks together along the channel to generate semantic features $\hat{F} \in \mathbb{R}^{\frac{1}{4}(|X_S|+|Q_S|+|Q_T|)HW\times 4k}$. This is a simple yet effective way to maintain discriminative ability and spatial information of semantic features.

Moreover, to leverage the semantics from previous tasks to guide the semantic feature learning of the current task, we

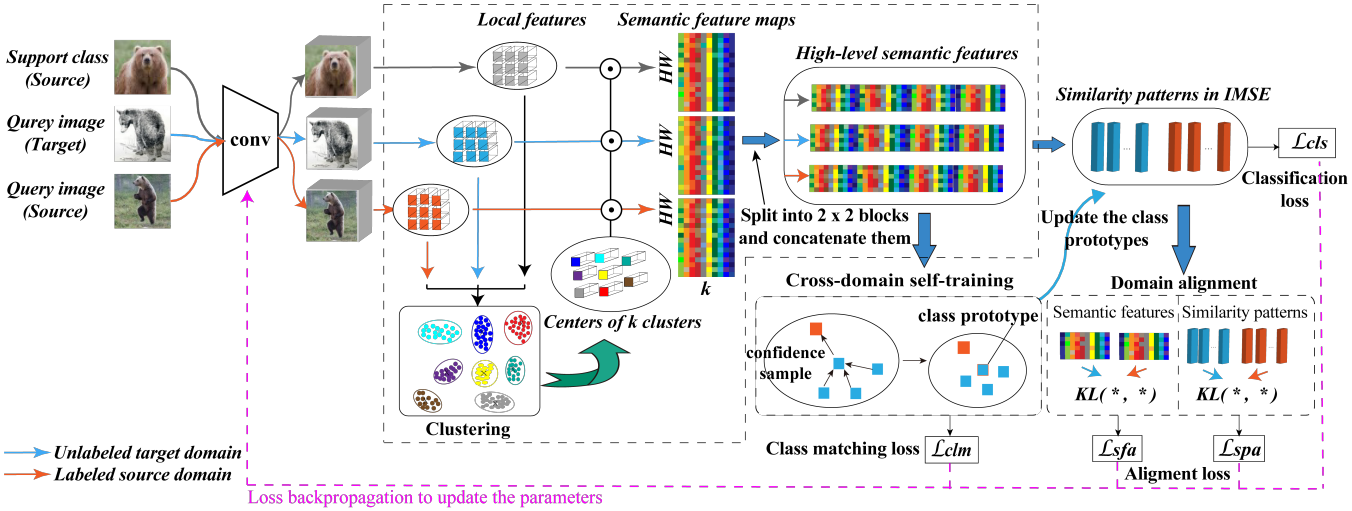

Figure 3: Illustration of our method training per episode for 1-shot FS-UDA tasks. First, support classes and query images from both domains are through a convolution network to extract their local features, followed by the task-specific semantic embedding module to learn high-level semantic features. Then, these semantic features are fed into the cross-domain self-training module to update the class prototypes for target domain classification and calculate the class matching loss $\mathcal{L}_{clm}$. Meanwhile, these semantic features are also used to generate similarity patterns in IMSE (Huang et al. 2021) for classification loss $\mathcal{L}_{cls}$. In addition, both semantic features and similarity patterns from both domains are aligned by the domain alignment module with the alignment losses $\mathcal{L}_{sfa}$ and $\mathcal{L}_{spa}$, respectively. Finally, all the losses are backpropagated to update our model.

utilize the centroids of previous clusters to update the initialization of clustering centroids by cross-attention (Chen, Fan, and Panda 2021). This makes K-means clustering converge rapidly.

After obtaining the semantic features $\hat{F}$, we use them for domain alignment and classification. Firstly, $\hat{F}$ is partitioned into $\hat{F}_{X_S}$, $\hat{F}_{Q_S}$, $\hat{F}_{Q_T}$ along with the first dimension. Then, we align $\hat{F}_{Q_S}$ and $\hat{F}_{Q_T}$ by minimizing the KL divergence of their distributions that will be introduced later. Meanwhile, we utilize $\hat{F}_{X_S}$, $\hat{F}_{Q_S}$ and $\hat{F}_{Q_T}$ to build 3-D similarity matrix M_q^c (Huang et al. 2021) between support and query sets. Finally, we calculate the similarity pattern p_q^c (measuring the similarity between query sample q and support class c) for classification (Huang et al. 2021). The classification loss using cross-entropy can be written by:

$$\mathcal{L}_{cls} = -\frac{1}{|Q_S|}\sum_{q\in Q_S}\log\left(\frac{\exp(\mathbf{1}\cdot p_q^c)}{\sum_{i=1}^{K}\exp(\mathbf{1}\cdot p_q^i)}\right). \quad (2)$$

Cross-domain Self-training

Since there is large domain shift between source and target domains, as well as label missing in target domain, adversarial domain adaptation on low-level local features cannot make samples of the same class between domains close, and thus could make the classes of two domains mismatched.

To alleviate the mismatching issue, we aim to find the most similar 'confidence' samples in Q_T with X_S to guide classification in target domain. We assume that it usually exists that the 'confidence' samples in Q_T could be classified correctly by X_S, when the distributions between domains are aligned. We iteratively select the 'confidence' samples in Q_T as the new prototypes to replace that in X_S for classification, as shown in Figure 2. We call the process as *cross-domain self-training*. The process can find more 'confidence' samples from Q_T than that in X_S for the same class, which could correct some misclassified samples in Q_T, thereby lightening the impact of domain gap.

Moreover, to improve the performance of the target domain classifier, we aim to make target domain samples q in Q_T closer to their most similar class and meanwhile far away from the other classes. Thus, we first calculate its similarity patterns p_q^{pos} (with the most similar class) and p_q^{neg} (with the second similar class), and then design the class matching loss with a margin m, which can be written by

$$\mathcal{L}_{clm} = \sum_{q\in Q_T}\max(\text{softmax}(p_q^{neg}) - \text{softmax}(p_q^{pos}) + m, 0), \quad (3)$$

where the similarity to the most similar class should be greater by m than the second similar class.

Two-level Domain Alignment

Conventional adversarial domain adaptation methods (Ganin et al. 2016)(Tzeng et al. 2017) iteratively train a discriminator to align the distribution of domains by adversarial training among tasks. However, they cannot be used to align the semantic features, because our semantic features are relevant to tasks, the semantics of the same channel could be varied for different tasks. Meanwhile, symmetrical alignment could bring the inference information of the target domain to the source domain (Li et al. 2020). Thus, we use asymmetrical KL divergence to align the distribution of domains on both semantic features and similarity patterns

within a task. Then, KL divergence can be calculated by:

$$KL(A,B) = \frac{1}{2}\big(tr(\Sigma_A^{-1}\Sigma_B) + \ln(\frac{\Sigma_A}{\Sigma_B}) + (\mu_A - \mu_B)\Sigma_A^{-1}(\mu_A - \mu_B)^\top - d\big), \tag{4}$$

where μ_A, μ_B, Σ_A and Σ_B are the mean vectors and the covariance matrices of sample matrix A and B, respectively. Thus, we minimize the KL divergence between semantic features $\hat{H}_{Q_S}$ and $\hat{H}_{Q_T}$ by

$$\mathcal{L}_{sfa} = KL(\hat{F}_{Q_S}, \hat{F}_{Q_T}). \tag{5}$$

Meanwhile, we also minimize the KL divergence to align the similarity patterns $\{p^c_{q_S}\}$ of Q_S and $\{p^c_{q_T}\}$ of Q_T with class c, which can be written by

$$\mathcal{L}_{spa} = \sum_{c=1}^{N} KL(\{p^c_{q_S}\}, \{p^c_{q_T}\}). \tag{6}$$

In sum, we combine all the above losses, w.r.t. classification (Eq. (2)), class matching (Eq. (3)) and KL-based domain alignment (Eqs. (5) and (6)) to train our model on many episodes. The total objective function can be written by:

$$\min \mathcal{L}_{cls} + \lambda_{sfa}\mathcal{L}_{sfa} + \lambda_{spa}\mathcal{L}_{spa} + \lambda_{clm}\mathcal{L}_{clm}, \tag{7}$$

where the hyper-parameters λ_{sfa}, λ_{spa} and λ_{clm} are introduced to balance the effect of different loss terms.

Experiment

***DomainNet* dataset.** We conduct extensive experiments on a multi-domain benchmark dataset *DomainNet* to demonstrate the efficacy of our method. It was released in 2019 for the research of multi-source domain adaptation (Peng et al. 2019). It contains 345 categories and six domains per category, *i.e., quickdraw*, *clipart*, *real*, *sketch*, *painting* and *infograph* domains. In our experiments, we follow the setting of IMSE in (Huang et al. 2021) to remove data insufficient domain *infograph*. There are 20 combinations totally for evaluation, and the dataset is split into 217, 43 and 48 categories for episodic training, model validation and testing new tasks, respectively. Note that in each split every category contains the five-domain images.

Network architecture and setting. We employ ResNet-12 as the backbone of feature embedding network, which is widely used in few-shot learning (Huang et al. 2021) (Gidaris et al. 2020). We obtain semantic features by first clustering the local features from each class of support set and two query sets and then concatenating them. During this process, we adopt cross-attention that consists of three convolution parameters to generate (Q, K, V) for attention calculation. In cross-domain self-training module, we set the threshold 1.7 of similarity score to select the 'confidence' samples in target domain. The margin m in Eq. (3) is empirically set to 1.5. In addition, we follow the setting of IMSE (Huang et al. 2021) to obtain similarity patterns. The hyper-parameters λ_{sfa}, λ_{spa} and λ_{clm} are set to 0.1, 0.05 and 0.01, by grid search, respectively.

Model training, validation and testing. To improve the performance, before episodic training, the feature embedding network is pretrained by using source domain data in the auxiliary dataset, as in (Huang et al. 2021). Afterwards, we perform episodic training on 280 episodes, following the setting of (Huang et al. 2021). During episode training, the total loss in Eq. (7) is minimized to optimize the network parameters for each episode. Also, we employ Adam optimizer with an initial learning rate of 10^{-4}, and meanwhile reduce the learning rate by half every 280 episodes. For model validation, we compare the performance of different model parameters on 100 tasks, which is randomly sampled from the validate set containing 43 categories. Then, we select the model parameters with the best validation accuracy for testing. During the testing, we randomly select 3000 tasks to calculate the averaged top-1 accuracy on these tasks as the evaluation criterion.

Comparison Experiments for FS-UDA

We conduct extensive experiments on *DomainNet* to compare our method with five FSL methods (ProtoNet (Snell, Swersky, and Zemel 2017), DN4 (Li et al. 2019), ADM (Li et al. 2020), FEAT (Ye et al. 2020), DeepEMD (Zhang et al. 2020)), three UDA methods, (MCD (Saito et al. 2018), ADDA (Tzeng et al. 2017), DWT (Roy et al. 2019)), their combinations and the most related method IMSE (Huang et al. 2021). For fair comparison, the results of these above methods are all reported from (Huang et al. 2021) with the same setting. Moreover, we also modify IMSE by using our semantic features for classification and domain adversary, namely IMSE+TSE. For fair comparison, these compared methods also pretrain the embedding network before episodic training, and they are trained on 1000 episodes.

Comparison analysis. Table 1 shows the results of all the compared methods for 20 cross-domain combinations, which records the averaged classification accuracy of target domain samples over 3000 5-way 1-shot/5-shot FS-UDA tasks. As observed, our TSECS achieves the best performance for all combinations and their average. Specifically, the UDA and FSL baselines in the first two parts perform the worst. In the third part, the combination methods with ADDA (Tzeng et al. 2017) perform domain adversarial training each episode, thus generally better than the above two parts, but still inferior to IMSE (Huang et al. 2021) and our TSECS. This is because the combination methods only perform domain alignment based on original feature maps, not considering the alignment of similarity patterns (related to classification predictions). Also, IMSE is worse than IMSE+TSE, which indicates high-level semantic features are more effective for FS-UDA than local features. However, they are still much worse than our method, showing the efficacy of high-level semantic features and cross-domain self-training for FS-UDA.

On the other hand, we can see that the 20 cross-domain combinations have considerably different performances. This is because several domains (*e.g., quickdraw*) are significantly different from other domains, while several other domains (e.g. *real, clipart*) are with the similar styles and features. Thus, for most compared methods, the performance becomes relatively low when the domain gap is large. For example, from *quickdraw* to *painting*, it performs the worst in all the other combinations because of larger domain

Methods	5-way, 1-shot						5-way, 5-shot					
	skt	*rel*	*qdr*	*pnt*	*cli*	**avg**	*skt*	*rel*	*qdr*	*pnt*	*cli*	**avg**
MCD	44.43	37.50	36.86	33.13	39.16	38.21	61.27	47.07	38.22	49.34	52.39	49.65
ADDA	45.11	42.02	38.15	37.25	42.06	40.91	61.00	55.14	36.08	51.56	57.71	51.76
DWT	46.78	41.68	39.45	33.45	45.68	41.38	60.40	53.56	36.99	50.37	57.40	51.74
ProtoNet	47.87	43.67	39.72	29.29	45.55	41.21	59.57	54.81	34.25	50.94	59.46	51.80
DN4	48.66	46.86	44.08	30.61	47.87	43.61	56.89	52.09	33.47	50.00	54.65	49.41
ADM	45.94	41.74	39.98	27.88	45.04	40.11	62.13	55.21	31.10	54.99	60.50	52.78
FEAT	48.20	46.36	30.85	41.02	47.73	43.14	62.14	57.22	34.05	54.67	60.83	53.78
DeepEMD	49.03	47.03	30.82	34.01	47.94	43.46	62.79	56.47	34.75	54.87	60.80	53.93
ADDA+ProtoNet	38.36	45.72	40.45	41.04	46.80	42.45	61.18	57.89	38.62	55.72	59.20	54.52
ADDA+ADM	49.52	44.94	32.02	42.95	49.37	44.65	57.55	51.79	36.27	50.50	55.58	50.33
ADDA+DN4	49.26	47.20	33.05	44.75	49.00	43.76	61.06	53.79	31.11	54.56	60.82	52.27
ADDA+FEAT	51.07	46.11	34.22	44.45	50.18	45.20	63.88	55.10	39.41	57.47	61.96	55.56
ADDA+DeepEMD	50.21	48.02	32.21	44.19	49.17	44.75	63.58	58.06	36.53	57.39	61.93	55.49
IMSE	54.18	51.45	39.28	46.88	52.51	48.86	57.62	61.54	43.46	59.99	65.42	59.60
IMSE+TSE	58.15	54.17	48.70	48.77	55.84	52.79	69.37	62.48	52.85	62.16	66.93	62.76
TSECS (ours)	**63.30**	**60.41**	**49.20**	**54.36**	**63.57**	**58.20**	**76.66**	**70.35**	**54.67**	**71.12**	**73.48**	**69.25**

Table 1: Comparison of our method with the related methods for 5-way 1-shot or 5-shot FS-UDA tasks. The first three blocks and IMSE are reported from (Huang et al. 2021), while the last two are the variant of IMSE we designed and ours, respectively. Each cell represents the average accuracy (%) of a compared method adapting from the current domain (source) to four other domains (target), as well as the average of all 20 combinations. The best results are in bold. Due to space limitation, the complete results for all 20 combinations could be seen in our arxiv version (Yu et al. 2023).

Components			Target Domains			
TSE	*catt*	*CS*	*cli*	*rel*	*qdr*	*pnt*
✓			61.98	60.00	52.21	51.62
		✓	57.07	53.31	41.93	46.66
✓	✓		62.74	60.54	53.64	54.23
✓		✓	68.25	61.15	58.31	53.34
✓	✓	✓	**69.45**	**65.00**	**62.25**	**56.51**

Table 2: Ablation study (%) of the modules designed in TSECS, where the FS-UDA tasks are evaluated from a domain (*sketch*) to the other four domains in *DomainNet*.

Components			Target Domains			
$\mathcal{L}_{sfa}$	$\mathcal{L}_{spa}$	$\mathcal{L}_{clm}$	*cli*	*rel*	*qdr*	*pnt*
✓			66.67	58.84	56.91	43.28
	✓		64.28	57.32	52.11	42.46
		✓	66.83	58.29	56.51	44.25
✓	✓		66.64	62.64	57.41	53.40
✓		✓	68.04	63.98	59.13	55.39
	✓	✓	67.61	62.47	53.07	54.14
✓	✓	✓	**69.45**	**65.00**	**62.25**	**56.51**

Table 3: Ablation study (%) of the three losses designed in TSECS, where the FS-UDA tasks are evaluated from a domain (*sketch*) to the other four domains in *DomainNet*.

gap, but our TSECS outperforms IMSE and the other compared methods by 8% and 12%, respectively. We found that our method has the larger performance improvement over IMSE, for these combinations containing *quickdraw*, which shows the efficacy of our method for large domain gap. Also, like TSECS, IMSE+TSE performs much better than IMSE for large domain gap, which indicates the high-level semantic features could conduct domain adaptation better than local features. In sum, these results reflect the advantages of our TSECS to deal with domain shift and task generalization in FS-UDA, no matter how large the domain gap is.

Ablation study of our method. We conduct various experiments on *DomainNet* to evaluate the effect of our modules: task-specific semantic embedding (*TSE*), cross-domain self-training (*CS*) and cross-attention in *TSE* (*catt*). The accuracies on the four target domains are reported in Table 2. As seen, our method achieve the best performance when three modules are all used. The performance of the single *CS* is the worst that shows that local features cannot align the distributions of the two domains, thus affecting cross-domain self-training. The module *TSE* is introduced into four combinations, all improving the performance, which validates the efficacy of our task-specific semantic features for FS-UDA again. Also, the addition of cross-attention into *TSE* will further improve the performance, which can help discover more semantics from previous tasks.

Ablation study of different losses. We conduct various experiments on *DomainNet* to further evaluate the effect of different losses in Eq. (7). Besides the classification loss ($\mathcal{L}_{cls}$), we combine the remaining three loss terms: 1) semantic features alignment loss ($\mathcal{L}_{sfa}$), 2) similarity pattern alignment loss ($\mathcal{L}_{spa}$), and 3) class matching loss ($\mathcal{L}_{clm}$). We evaluate 5-way 1-shot FS-UDA tasks from *sketch* to the other four domains, respectively, and their accuracies are reported in Table 3. As observed, the more the number of loss terms involved, the higher the accuracy. The combination of

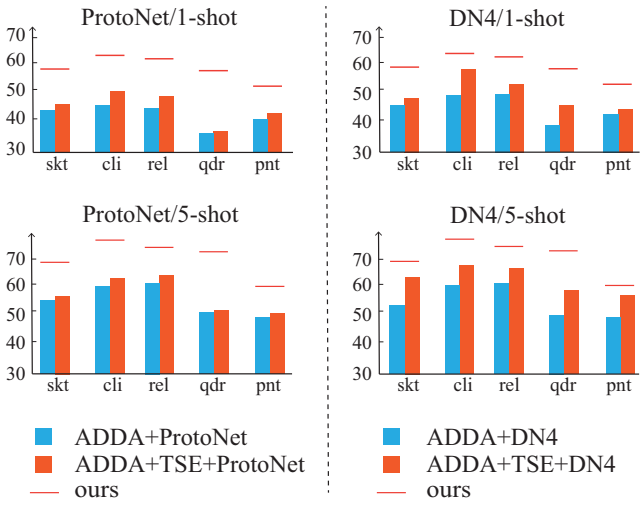

Figure 4: Comparison of introducing our TSE module or not into two FSL methods ProtoNet (Snell, Swersky, and Zemel 2017) and DN4 (Li et al. 2019) with ADDA (Tzeng et al. 2017) combined, *i.e.,* ADDA+ProtoNet and ADDA+DN4.

all the three losses is the best. For the single loss, both $\mathcal{L}_{sfa}$ and $\mathcal{L}_{clm}$ perform better than $\mathcal{L}_{spa}$, and their combination is also considerably better than the other paired combinations, showing the efficacy of semantic feature domain alignment and class matching in target domain. Based on the above, adding $\mathcal{L}_{spa}$ further improves the performance, indicating positive effect of aligning the similarity patterns.

Evaluation on the effect of our task-specific semantic embedding module on two FSL methods with ADDA (Tzeng et al. 2017) combined. Compared with ADDA+DN4 and ADDA+ProtoNet, we add our semantic embedding module (*TSE*) with the loss $\mathcal{L}_{sfa}$ into their feature embedding models, and test them on 3000 new 5-way 1/5-shot FS-UDA tasks. For simplification and clarification, we calculate the averaged accuracies from every domain to the other four domains and show them in Figure 4. As seen, the methods using *TSE* generally perform better than that without it, which validates that the semantic embedding in *TSE* could generate more discriminative semantic features for classification than original local features. In addition, the performances of these methods are still far from our method because using ADDA is insufficient to align the domains and could result in class mismatching, but our method can effectively solve it by cross-domain self-training.

Evaluation of dataset generalization. We evaluate the generalization of our model trained on *DomainNet* to adapt to a substantially different dataset *miniImageNet*. We modify *miniImageNet* by transferring a half of real images (rel) into sketch images (skt) by MUNIT (Huang et al. 2018) to produce two domains for FS-UDA. We compare our method with ADDA+DN4, ADDA+DeepEMD and IMSE for 5-way 1-shot FS-UDA tasks for $rel \leftrightarrow skt$. The results are shown as Table 4. As observed, our method outperforms other methods, specially for *ske* $\rightarrow$ *rel*. For *rel* $\rightarrow$ *skt*, our method is slightly better than IMSE, because the style of *sketch* images in *miniImageNet* is relatively different from that in *DomainNet*, which could effect the learned semantic features.

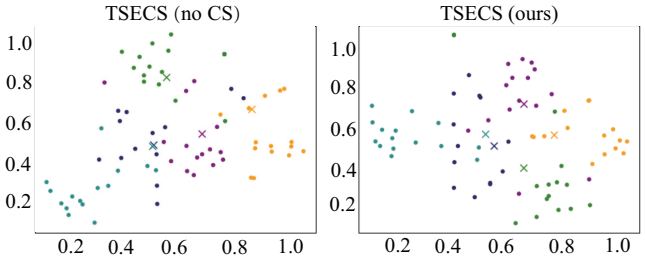

Figure 5: The *tSNE* visualization of our TSECS using cross-domain self-training or not for a 5-way 5-shot FS-UDA task from *sketch* to *clipart*. The samples with different colors belong to different classes, and the stars in the left and right figures represent the class centroids of support set and selected target domain query samples, respectively.

Methods	$skt \rightarrow rel$	$rel \rightarrow skt$
ADDA+DN4	44.01 ± 0.87	40.61 ± 0.90
ADDA+DeepEMD	46.14 ± 0.82	45.91 ± 0.77
IMSE	48.78 ± 0.78	48.52 ± 0.81
TSECS (ours)	**53.33** ± 1.08	**49.83** ± 0.96

Table 4: Evaluation (%) of dataset generalization for 5-way 1-shot FS-UDA tasks between domains *real* and *sketch*, performing episodic training on *DomainNet* and testing on expanded dataset *miniImageNet*.

Visualization of our method using cross-domain self-training or not. We illustrate the *tSNE* results of a 5-way 5-shot FS-UDA task from *sketch* to *clipart* in Figure 5. Note that the class prototypes in the left subfigure belong to the support set in source domain, while those in the right subfigure are generated by 'confidence' samples in target domain. It is obvious that two class prototypes in the left subfigure are fully overlapped so that many samples could not be correctly classified. In contrast, the right subfigure has the better class prototypes, and samples from different classes are more distinguishable. This shows the efficacy of our cross-domain self-training that finds 'confidence' samples to train the target domain classifier and uses class matching loss $\mathcal{L}_{clm}$ to shorten the distance of samples of the same class.

Conclusion

In this paper, we propose a novel method TSECS for FS-UDA. We extract high-level semantic features than local features to measure the similarity of query images in target domain to support classes in source domain. Moreover, we design cross-domain self-training to train a target domain classifier. In addition, asymmetrical KL-divergence is used to align the semantic features between domains. Extensive experiments on *DomainNet* show the efficacy of our TSECS, significantly improving the performance for FS-UDA.

Acknowledgments

Wanqi Yang and Ming Yang are supported by National Natural Science Foundation of China (Grant Nos. 62076135, 62276138, 61876087). Lei Wang is supported by an Australian Research Council Discovery Project (No. DP200101289) funded by the Australian Government.

References

Bertinetto, L.; Henriques, J. F.; Torr, P.; and Vedaldi, A. 2019. Meta-learning with differentiable closed-form solvers. In *International Conference on Learning Representations*, 1–8.

Chen, C.-F. R.; Fan, Q.; and Panda, R. 2021. CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 347–356.

Chen, W.-Y.; Liu, Y.-C.; Kira, Z.; Wang, Y.-C. F.; and Huang, J.-B. 2019. A Closer Look at Few-shot Classification. In *International Conference on Learning Representations*, 1–16.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In Precup, D.; and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, 1126–1135.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; March, M.; and Lempitsky, V. 2016. Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, 17(59): 1–35.

Gidaris, S.; Bursuc, A.; Komodakis, N.; Perez, P.; and Cord, M. 2020. Learning Representations by Predicting Bags of Visual Words. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6926–6936.

Huang, S.; Yang, W.; Wang, L.; Zhou, L.; and Yang, M. 2021. Few-Shot Unsupervised Domain Adaptation with Image-to-Class Sparse Similarity Encoding. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, 677–685. New York, NY, USA: Association for Computing Machinery. ISBN 9781450386517.

Huang, X.; Liu, M.-Y.; Belongie, S.; and Kautz, J. 2018. Multimodal Unsupervised Image-to-Image Translation. In Ferrari, V.; Hebert, M.; Sminchisescu, C.; and Weiss, Y., eds., *Computer Vision – ECCV 2018*, 179–196. Cham: Springer International Publishing. ISBN 978-3-030-01219-9.

Kim, D.; Saito, K.; Oh, T.-H.; Plummer, B. A.; Sclaroff, S.; and Saenko, K. 2021. CDS: Cross-Domain Self-supervised Pre-training. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 9103–9112.

Li, W.; Wang, L.; Huo, J.; Shi, Y.; Gao, Y.; and Luo, J. 2020. Asymmetric Distribution Measure for Few-shot Learning. In Bessiere, C., ed., *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 2957–2963. International Joint Conferences on Artificial Intelligence Organization. Main track.

Li, W.; Wang, L.; Xu, J.; Huo, J.; Gao, Y.; and Luo, J. 2019. Revisiting Local Descriptor Based Image-To-Class Measure for Few-Shot Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7260–7268.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. 2015. Learning Transferable Features with Deep Adaptation Networks. In Bach, F.; and Blei, D., eds., *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, 97–105. Lille, France: PMLR.

Luo, Y.; Liu, P.; Guan, T.; Yu, J.; and Yang, Y. 2020. Adversarial Style Mining for One-Shot Unsupervised Domain Adaptation. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 20612–20623. Curran Associates, Inc.

Peng, X.; Bai, Q.; Xia, X.; Huang, Z.; Saenko, K.; and Wang, B. 2019. Moment Matching for Multi-Source Domain Adaptation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 1406–1415.

Ravi, S.; and Larochelle, H. 2017. Optimization as a Model for Few-Shot Learning. In *International Conference on Learning Representations*.

Roy, S.; Siarohin, A.; Sangineto, E.; Bulo, S. R.; Sebe, N.; and Ricci, E. 2019. Unsupervised Domain Adaptation Using Feature-Whitening and Consensus Loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9471–9480.

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)*, 3723–3732.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical Networks for Few-shot Learning. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30, 4077–4087. Curran Associates, Inc.

Tang, H.; Chen, K.; and Jia, K. 2020. Unsupervised Domain Adaptation via Structurally Regularized Deep Clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tseng, H.-Y.; Lee, H.-Y.; Huang, J.-B.; and Yang, M.-H. 2020. Cross-Domain Few-Shot Classification via Learned Feature-Wise Transformation. In *International Conference on Learning Representations*.

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial Discriminative Domain Adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2962–2971.

Tzeng, E.; Hoffman, J.; Zhang, N.; Saenko, K.; and Darrell, T. 2014. Deep Domain Confusion: Maximizing for Domain Invariance. *CoRR*, abs/1412.3474: 1–9.

Vinyals, O.; Blundell, C.; Lillicrap, T.; kavukcuoglu, k.; and Wierstra, D. 2016. Matching Networks for One Shot Learning. In Lee, D.; Sugiyama, M.; Luxburg, U.; Guyon, I.; and

Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 29, 3630–3638. Curran Associates, Inc.

Yang, W.; Yang, C.; Huang, S.; Wang, L.; and Yang, M. 2022. Few-shot Unsupervised Domain Adaptation via Meta Learning. In *IEEE International Conference on Multimedia and Expo (ICME)*.

Ye, H.-J.; Hu, H.; Zhan, D.-C.; and Sha, F. 2020. Few-Shot Learning via Embedding Adaptation With Set-to-Set Functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8805–8814.

Yu, L.; Yang, W.; Huang, S.; Wang, L.; and Yang, M. 2023. High-level semantic feature matters few-shot unsupervised domain adaptation. *arXiv preprint arXiv:2301.01956*.

Yue, X.; Zheng, Z.; Zhang, S.; Gao, Y.; Darrell, T.; Keutzer, K.; and Vincentelli, A. S. 2021. Prototypical Cross-Domain Self-Supervised Learning for Few-Shot Unsupervised Domain Adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13834–13844.

Zhang, C.; Cai, Y.; Lin, G.; and Shen, C. 2020. DeepEMD: Few-Shot Image Classification With Differentiable Earth Mover's Distance and Structured Classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12200–12210.

Zou, Y.; Yu, Z.; Liu, X.; Kumar, B. V. K. V.; and Wang, J. 2019. Confidence Regularized Self-Training. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 5981–5990.

Zou, Y.; Yu, Z.; Vijaya Kumar, B. V. K.; and Wang, J. 2018. Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-training. In Ferrari, V.; Hebert, M.; Sminchisescu, C.; and Weiss, Y., eds., *Computer Vision – ECCV 2018*, 297–313. Cham: Springer International Publishing. ISBN 978-3-030-01219-9.