Learning to Combine: Knowledge Aggregation for Multi-Source Domain Adaptation

Hang Wang[*], Minghao Xu[*], Bingbing Ni[**], and Wenjun Zhang

Shanghai Jiao Tong University, Shanghai 200240, China
`{wang--hang, xuminghao118, nibingbing, zhangwenjun}@sjtu.edu.cn`

Abstract. Transferring knowledges learned from multiple source domains to target domain is a more practical and challenging task than conventional single-source domain adaptation. Furthermore, the increase of modalities brings more difficulty in aligning feature distributions among multiple domains. To mitigate these problems, we propose a Learning to Combine for Multi-Source Domain Adaptation (LtC-MSDA) framework via exploring interactions among domains. In the nutshell, a knowledge graph is constructed on the prototypes of various domains to realize the information propagation among semantically adjacent representations. On such basis, a graph model is learned to predict query samples under the guidance of correlated prototypes. In addition, we design a Relation Alignment Loss (RAL) to facilitate the consistency of categories' relational interdependency and the compactness of features, which boosts features' intra-class invariance and inter-class separability. Comprehensive results on public benchmark datasets demonstrate that our approach outperforms existing methods with a remarkable margin. Our code is available at `https:github.com/ChrisAllenMing/LtC-MSDA`.

Keywords: Multi-Source Domain Adaptation, Learning to Combine, Knowledge Graph, Relation Alignment Loss

1 Introduction

Deep Neural Network (DNN) is expert at learning discriminative representations under the support of massive labeled data, and it has achieved incredible successes in many computer-vision-related tasks, *e.g.* object classification [17,11], object detection [34,23] and semantic segmentation [3,10]. However, when directly deploying the model trained on a specific dataset to the scenarios with distinct backgrounds, weather or illumination, undesirable performance decay commonly occurs, due to the existence of domain shift [49].

Unsupervised Domain Adaptation (UDA) is an extensively explored technique to address such problem, and it focuses on the transferability of knowledge learned from a labeled dataset (source domain) to another unlabeled one (target domain). The basic intuition behind these attempts is that knowledge transfer

---

[*] Equal contribution.
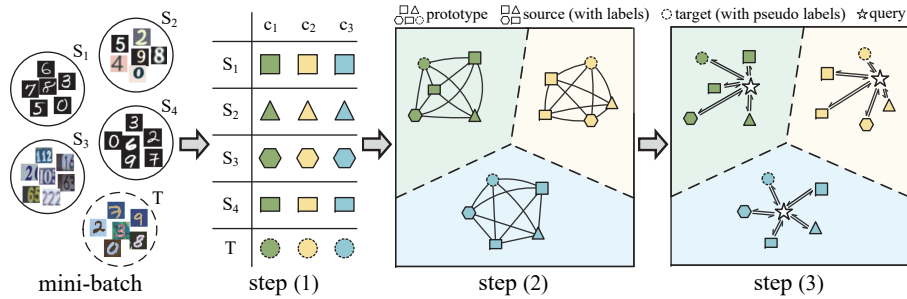[**] Corresponding author: Bingbing Ni.

Fig. 1: Given a randomly sampled mini-batch, in step (1), our model first updates each category's global prototype for all domains. In step (2), a knowledge graph is constructed on these prototypes. Finally, in step (3), a bunch of query samples are inserted into the graph and predicted via knowledge aggregation.

can be achieved by boosting domain-invariance of feature representations from different domains. In order to realize such goal, various strategies have been proposed, including minimizing explicitly defined domain discrepancy metrics [24,46,39], adversarial-training-based domain confusion [4,40,25] and GAN-based domain alignment [2,6,37].

However, in real-world applications, it is unreasonable to deem that the labeled images are drawn from a single domain. Actually, these samples can be collected under different deployment environments, *i.e.* from multiple domains, which reflect distinct modal information. Integrating such factor into domain alignment, a more practical problem is *Multi-Source Domain Adaptation* (MSDA), which dedicates to transfer the knowledges learned from multiple source domains to an unlabeled target domain.

Inspired by the theoretical analysis [29,12], recent works [45,33,52] predict target samples by combining the predictions of source classifiers. However, the interaction of feature representations learned from different domains has not been explored to tackle MSDA tasks. Compared to combining classifiers' predictions using hand-crafted or model-induced weights, knowledge propagation among multiple domains enables related feature representations to interact with each other before final prediction, which makes the operation of domain combination learnable. In addition, although category-level domain adaptation has been extensively studied in the literature, *e.g.* maximizing dual classifiers discrepancy [36,19] and prototype-based alignment [42,32], the relationships among categories are not constrained in these works. For instance, the source domain's knowledges that truck is more similar to car than person should also be applicable to target domain. Motivated by these limitations, we propose a novel framework and loss function for MSDA as follows.

Learning to Combine. We propose a new framework, *Learning to Combine for MSDA* (LtC-MSDA), which leverages the knowledges learned from multiple source domains to assist model's inference on target domain. In the training phase, three major steps are performed, which are graphically illustrated

in Figure 1. (1) *Global prototype[‡] maintenance*: Based on a randomly sampled mini-batch containing samples from source and target domains, we estimate the prototype representation of each category for all domains. In order to mitigate the randomness of these estimations, global prototypes are maintained through a moving average scheme. (2) *Knowledge graph construction*: In this step, a knowledge graph is constructed on the global prototypes of different domains, and the connection weight between two global prototypes is determined by their similarity. (3) *Knowledge-aggregation-based prediction*: Given a bunch of query samples from arbitrary domains, we first extend the knowledge graph with these samples. After that, a graph convolutional network (GCN) is employed to propagate feature representations throughout the extended graph and output the classification probability for each node. After training, the knowledge graph is saved, and only step (3) is conducted for model's inference.

Class-relation-aware Domain Alignment. During the process of domain adaptation, in order to exploit the relational interdependency among categories, we propose a *Relation Alignment Loss* (RAL), which is composed of a global and a local term. (1) *Global relation constraint*: In this term, based on the adjacency matrix of knowledge graph, we constrain the connection weight between two arbitrary classes to be consistent among all domains, which refines the relative position of different classes' features in the latent space. (2) *Local relation constraint*: This term facilitates the compactness of various categories' features. In specific, we restrain the feature representation of a sample to be as close as possible to its corresponding global prototype, which makes the features belonging to distinct categories easier to be separated.

Our contributions can be summarized as follows:

1. We propose a Learning to Combine for MSDA (LtC-MSDA) framework, in which the knowledges learned from source domains interact with each other and assist model's prediction on target domain.
2. In order to better align the feature distributions of source and target domains, we design a Relation Alignment Loss (RAL) to constrain the global and local relations of feature representations.
3. We evaluate our model on three benchmark datasets with different domain shift and data complexity, and extensive results show that the proposed method outperforms existing approaches with a clear margin.

2   Related Work

Unsupervised Domain Adaptation (UDA). UDA seeks to generalize a model learned from a labeled source domain to a new target domain without labels. Many previous methods achieve such goal via minimizing an explicit domain discrepancy metric [41,46,24,19,39]. Adversarial learning is also employed to align two domains on feature level [4,40,25] or pixel level [2,6,37,44]. Recently,

---

[‡] Prototype is the mean embedding of all samples within the same class.

a group of approaches performs category-level domain adaptation through utilizing dual classifier [36,19], or domain prototype [42,32,43]. In this work, we further explore the consistency of category relations on all domains.

Multi-Source Domain Adaptation (MSDA). MSDA assumes data are collected from multiple source domains with different distributions, which is a more practical scenario compared to single-source domain adaptation. Early theoretical analysis [29,1] gave strong guarantees for representing target distribution as the weighted combination of source distributions. Based on these works, Hoffman *et al.* [12] derived normalized solutions for MSDA problems. Recently, Zhao *et al.* [51] aligned target domain to source domains globally using adversarial learning. Xu *et al.* [45] deployed multi-way adversarial learning and combined source-specific perplexity scores for target predictions. Peng *et al.* [33] proposed to transfer knowledges by matching the moments of feature representations. In [52], source distilling mechanism is introduced to fine-tune the separately pre-trained feature extractor and classifier.

Improvements over existing methods. In order to derive the predictions of target samples, former works [45,33,52] utilize the ensemble of source classifiers to output weighted classification probabilities, while such combination scheme prohibits the end-to-end learnable model. In this work, we design a *Learning to Combine* framework to predict query samples based on the interaction of knowledges learned from source and target domains, which makes the whole model end-to-end learnable.

Knowledge Graph. A knowledge graph describes entities and their interrelations, organized in a graph. Learning knowledge graphs and using attribute relationships has recently been of interest to the vision community. Several works [8,16] utilize knowledge graphs based on the defined semantic space for natural language understanding. For multi-label image classification [30,20], knowledge graphs are applied to exploit explicit semantic relations. In this paper, we construct a knowledge graph on global prototypes of different domains, which lays foundation for our method.

Graph Convolutional Network (GCN). GCN [15] is designed to compute directly on graph-structured data and model the inner structural relations. Such structures typically come from some prior knowledges about specific problems. Due to its effectiveness, GCNs have been widely used in various tasks, *e.g.* action recognition [47], person Re-ID [48,22] and point cloud learning [21]. For MSDA task, we employ GCN to propagate information on the knowledge graph.

3   Method

In Multi-Source Domain Adaptation (MSDA), there are M source domains S_1, S_2, $\cdots$, S_M. The domain $S_m = \{(x_i^{\mathcal{S}_m}, y_i^{\mathcal{S}_m})\}_{i=1}^{N_{\mathcal{S}_m}}$ is characterized by $N_{\mathcal{S}_m}$ i.i.d. labeled samples, where $x_i^{\mathcal{S}_m}$ follows one of the source distributions $\mathbb{P}_{\mathcal{S}_m}$ and $y_i^{\mathcal{S}_m} \in \{1, 2, \cdots, K\}$ (K is the number of classes) denotes its corresponding label. Similarly, target domain $\mathcal{T} = \{x_j^{\mathcal{T}}\}_{j=1}^{N_{\mathcal{T}}}$ is represented by $N_{\mathcal{T}}$ i.i.d. unlabeled samples, where $x_j^{\mathcal{T}}$ follows target distribution $\mathbb{P}_{\mathcal{T}}$. In the training phase, a

randomly sampled mini-batch $B = \{\widehat{\mathcal{S}}_1, \widehat{\mathcal{S}}_2, \cdots, \widehat{\mathcal{S}}_M, \widehat{\mathcal{T}}\}$ is used to characterize source and target domains, and $|B|$ denotes the batch size.

3.1 Motivation and Overview

For MSDA, the core research topic is how to achieve more precise predictions for target samples through fully utilizing the knowledges among different domains. In order to mitigate the error of single-source prediction, recent works [45,33,52] express the classification probabilities of target samples as the weighted average of source classifiers' predictions. However, such scheme requires prior knowledges about the relevance of different domains to obtain combination weights, which makes the whole model unable to be end-to-end learnable.

In addition, learning to generalize from multiple source domains to target domain has a "double-edged sword" effect on model's performance. From one perspective, samples from multiple domains provide more abundant modal information of different classes, and thus the decision boundaries are refined according to more support points. From the other perspective, the distribution discrepancy among distinct source domains increases the difficulty of learning domain-invariant features. Off-the-shelf UDA techniques might fail in the condition that multi-modal distributions are to be aligned, since the relevance among different modalities, *i.e.* categories of various domains, are not explicitly constrained in these methods. Such constraints [7,38] are proved to be necessary when large amounts of clusters are formed in the latent space.

To address above issues, we propose a *Learning to Combine for MSDA* (LtC-MSDA) framework. In specific, a knowledge graph is constructed on the prototypes of different domains to enable the interaction among semantically adjacent entities, and query samples are added into this graph to obtain their classification probabilities under the guidance of correlated prototypes. In this process, the combination of different domains' knowledges is achieved via information propagation, which can be learned by a graph model. On the basis of this framework, a *Relation Alignment Loss* (RAL) is proposed, which facilitates the consistency of categories' relational interdependency on all domains and boosts the compactness of feature embeddings within the same class.

3.2 Learning to Combine for MSDA

In the proposed LtC-MSDA framework, for each training iteration, a mini-batch containing samples from all domains is mapped to latent space, and the produced feature embeddings are utilized to update global prototypes and also served as queries. After that, global prototypes and query samples are structured as a knowledge graph. Finally, a GCN model is employed to perform information propagation and output classification probability for each node of knowledge graph. Figure 2 gives a graphical illustration of the whole framework, and its details are presented in the following parts.

Global prototype maintenance. This step updates global prototypes with mini-batch statistics. Based on a mini-batch B, we estimate the prototype of each
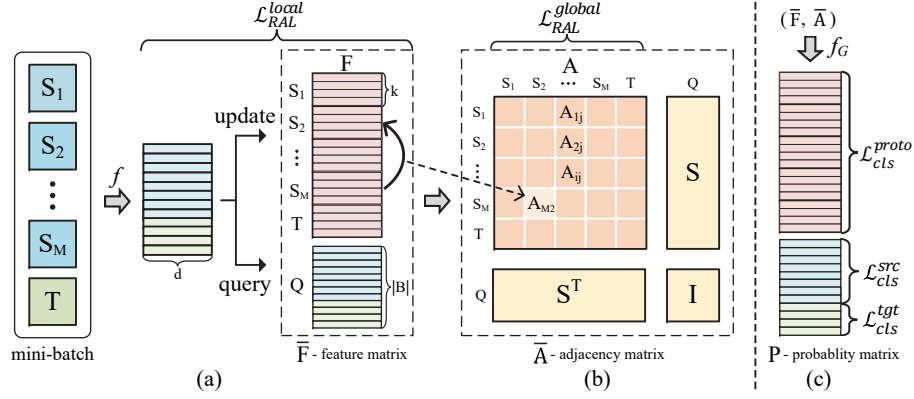
Fig. 2: **Framework overview.** (a) A randomly sampled mini-batch is utilized to update global prototypes and also serves as query samples, and the local relation loss $\mathcal{L}_{RAL}^{local}$ is constrained to promote feature compactness. (b) A knowledge graph is constructed on prototypes, whose adjacency matrix $\mathbf{A}$ embodies the relevance among different domains' categories. On the basis of block matrices in $\mathbf{A}$, global relation loss $\mathcal{L}_{RAL}^{global}$ is derived. (c) Extended by query samples, feature matrix $\bar{\mathbf{F}}$ and adjacency matrix $\bar{\mathbf{A}}$ are fed into a GCN model f_G to produce final predictions $\mathbf{P}$. On such basis, three kinds of classification losses are defined.

category for all domains. For source domain $\mathcal{S}_m$, the estimated prototype $\widehat{c}_k^{\mathcal{S}_m}$ is defined as the mean embedding of all samples belonging to class k in $\widehat{\mathcal{S}}_m$:

$$\widehat{c}_k^{\mathcal{S}_m} = \frac{1}{|\widehat{\mathcal{S}}_m^k|} \sum_{(x_i^{\mathcal{S}_m}, y_i^{\mathcal{S}_m}) \in \widehat{\mathcal{S}}_m^k} f(x_i^{\mathcal{S}_m}), \tag{1}$$

where $\widehat{\mathcal{S}}_m^k$ is the set of all samples with class label k in the sampling $\widehat{\mathcal{S}}_m$, and f represents the mapping from image to feature embedding.

For target domain $\mathcal{T}$, since ground truth information is unavailable, we first assign pseudo labels for the samples in $\widehat{\mathcal{T}}$ using the strategy proposed by [50], and the estimated prototype $\widehat{c}_k^{\mathcal{T}}$ of target domain is defined as follows:

$$\widehat{c}_k^{\mathcal{T}} = \frac{1}{|\widehat{\mathcal{T}}_k|} \sum_{(x_i^{\mathcal{T}}, \widehat{y}_i^{\mathcal{T}}) \in \widehat{\mathcal{T}}_k} f(x_i^{\mathcal{T}}), \tag{2}$$

where $\widehat{y}_i^{\mathcal{T}}$ is the pseudo label assigned to $x_i^{\mathcal{T}}$, and $\widehat{\mathcal{T}}_k$ denotes the set of all samples labeled as class k in $\widehat{\mathcal{T}}$. In order to correct estimation bias brought by the randomness of mini-batch samplings, we maintain the global prototypes for source and target domains with an exponential moving average scheme:

$$c_k^{\mathcal{S}_m} := \beta c_k^{\mathcal{S}_m} + (1 - \beta)\widehat{c}_k^{\mathcal{S}_m} \quad m = 1, 2, \cdots, M, \tag{3}$$

$$c_k^{\mathcal{T}} := \beta c_k^{\mathcal{T}} + (1 - \beta)\widehat{c}_k^{\mathcal{T}}, \tag{4}$$

where β is the exponential decay rate which is fixed as 0.7 in all experiments. Such moving average scheme is broadly used in the literature [14,42,9] to stabilize the training process through smoothing global variables.

Knowledge graph construction. In order to further refine category-level representations with knowledges learned from multiple domains, this step structures the global prototypes of various domains as a knowledge graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. In this graph, the vertex set $\mathcal{V}$ corresponds to $(M+1)K$ prototypes, and the feature matrix $\mathbf{F} \in \mathbb{R}^{|\mathcal{V}| \times d}$ (d: the dimension of feature embedding) is defined as the concatenation of global prototypes:

$$\mathbf{F} = \left[\underbrace{c_1^{\mathcal{S}_1} c_2^{\mathcal{S}_1} \cdots c_K^{\mathcal{S}_1}}_{\text{prototypes of } \mathcal{S}_1} \cdots \underbrace{c_1^{\mathcal{S}_M} c_2^{\mathcal{S}_M} \cdots c_K^{\mathcal{S}_M}}_{\text{prototypes of } \mathcal{S}_M} \underbrace{c_1^{\mathcal{T}} c_2^{\mathcal{T}} \cdots c_K^{\mathcal{T}}}_{\text{prototypes of } \mathcal{T}} \right]^{\mathrm{T}}. \tag{5}$$

The edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ describes the relations among vertices, and an adjacency matrix $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ is employed to model such relationships. In specific, we derive the adjacency matrix by applying a Gaussian kernel $\mathcal{K}_G$ over pairs of global prototypes:

$$\mathbf{A}_{i,j} = \mathcal{K}_G(\mathbf{F}_i^{\mathrm{T}}, \mathbf{F}_j^{\mathrm{T}}) = \exp\left(-\frac{||\mathbf{F}_i^{\mathrm{T}} - \mathbf{F}_j^{\mathrm{T}}||_2^2}{2\sigma^2} \right), \tag{6}$$

where $\mathbf{F}_i^{\mathrm{T}}$ and $\mathbf{F}_j^{\mathrm{T}}$ denote the i-th and j-th global prototype in feature matrix $\mathbf{F}$, and σ is the standard deviation parameter controlling the sparsity of $\mathbf{A}$.

Knowledge-aggregation-based prediction. In this step, we aim to obtain more accurate predictions for query samples under the guidance of multiple domains' knowledges. We regard the mini-batch B as a bunch of query samples and utilize them to establish an extended knowledge graph $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{\mathcal{E}})$. In this graph, the vertex set $\bar{\mathcal{V}}$ is composed of the original vertices in $\mathcal{V}$, *i.e.* global prototypes, and query samples' feature embeddings, which yields an extended feature matrix $\bar{\mathbf{F}} \in \mathbb{R}^{|\bar{\mathcal{V}}| \times d}$ as follows:

$$\bar{\mathbf{F}} = \left[\mathbf{F}^{\mathrm{T}} \; f(q_1) \; f(q_2) \; \cdots \; f(q_{|B|}) \right]^{\mathrm{T}}, \tag{7}$$

where q_i ($i = 1, 2, \cdots, |B|$) denotes the i-th query sample.

The edge set $\bar{\mathcal{E}}$ is expanded with the edges of new vertices. Concretely, an extended adjacency matrix $\bar{\mathbf{A}}$ is derived by adding the connections between global prototypes and query samples:

$$\mathbf{S}_{i,j} = \mathcal{K}_G(\mathbf{F}_i^{\mathrm{T}}, f(q_j)) = \exp\left(-\frac{||\mathbf{F}_i^{\mathrm{T}} - f(q_j)||_2^2}{2\sigma^2} \right), \tag{8}$$

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{S} \\ \mathbf{S}^{\mathrm{T}} & \mathbf{I} \end{bmatrix}, \tag{9}$$

where $\mathbf{S} \in \mathbb{R}^{|\mathcal{V}| \times |B|}$ is the similarity matrix measuring the relevance between original and new vertices. Considering that the semantic information from a single sample is not precise enough, we ignore the interaction among query samples and use an identity matrix $\mathbf{I}$ to depict their relations.

After these preparations, a Graph Convolutional Network (GCN) is employed to propagate feature representations throughout the extended knowledge graph, such that the representations within the same category are encouraged to be consistent across all domains and query samples. In specific, inputted with the feature matrix $\bar{\mathbf{F}}$ and adjacency matrix $\bar{\mathbf{A}}$, the GCN model f_G outputs the classification probability matrix $\mathbf{P} \in \mathbb{R}^{|\bar{\mathcal{V}}| \times K}$ as follows:

$$\mathbf{P} = f_G\left(\bar{\mathbf{F}}, \ \bar{\mathbf{A}}\right). \tag{10}$$

Model inference. After training, we store the feature extractor f, GCN model f_G, feature matrix $\mathbf{F}$ and adjacency matrix $\mathbf{A}$. For inference, only the *knowledge-aggregation-based prediction* step is conducted. Concretely, based on the feature embeddings extracted by f, the extended feature matrix $\bar{\mathbf{F}}$ and adjacency matrix $\bar{\mathbf{A}}$ are derived by Eq. 7 and Eq. 9 respectively. Using these two matrices, the GCN model f_G produces the classification probabilities for test samples.

3.3   Class-relation-aware Domain Alignment

In the training phase, our model is optimized by two kinds of losses which facilitate the domain-invariance and distinguishability of feature representations. The details are stated below.

Relation Alignment Loss (RAL). This loss aims to conduct domain alignment on category level. During the domain adaptation process, except for promoting the invariance of same categories' features, it is necessary to constrain the relative position of different categories' feature embeddings in the latent space, especially when numerous modalities exist in the task, *e.g.* MSDA. Based on this idea, we propose the RAL which consists of a global and a local constraint:

$$\mathcal{L}_{RAL} = \lambda_1 \mathcal{L}_{RAL}^{global} + \lambda_2 \mathcal{L}_{RAL}^{local}, \tag{11}$$

where λ_1 and λ_2 are trade-off parameters.

For the global term, we facilitate the relevance between two arbitrary classes to be consistent on all domains, which is implemented through measuring the similarity of block matrices in $\mathbf{A}$:

$$\mathcal{L}_{RAL}^{global} = \frac{1}{(M+1)^4} \sum_{i,j,m,n=1}^{M+1} ||\mathbf{A}_{i,j} - \mathbf{A}_{m,n}||_F, \tag{12}$$

where the block matrix $\mathbf{A}_{i,j}$ $(1 \leqslant i, j \leqslant M+1)$ evaluates all categories' relevance between the i-th and j-th domain, which is shown in Figure 2(b), and $||\cdot||_F$ denotes Frobenius norm. In this loss, features' intra-class invariance is boosted by the constraints on block matrices' main diagonal elements, and the consistency of different classes' relational interdependency is promoted by the constraints on other elements of block matrices.

For the local term, we enhance the feature compactness of each category via impelling the feature embeddings of samples in mini-batch B to approach their

corresponding global prototypes, which derives the following loss function:

$$\mathcal{L}_{RAL}^{local} = \frac{1}{|B|} \sum_{k=1}^{K} \Bigg(\sum_{m=1}^{M} \sum_{(x_i^{\mathcal{S}_m}, y_i^{\mathcal{S}_m}) \in \widehat{\mathcal{S}}_m^k} ||f(x_i^{\mathcal{S}_m}) - c_k^{\mathcal{S}_m}||_2^2 \\ + \sum_{(x_i^{\mathcal{T}}, \widehat{y}_i^{\mathcal{T}}) \in \widehat{\mathcal{T}}_k} ||f(x_i^{\mathcal{T}}) - c_k^{\mathcal{T}}||_2^2 \Bigg). \tag{13}$$

Classification losses. This group of losses aims to enhance features' distinguishability. Based on the predictions of all vertices in extended knowledge graph $\bar{\mathcal{G}}$, the classification loss is defined as the composition of three terms for global prototypes, source samples and target samples respectively:

$$\mathcal{L}_{cls} = \mathcal{L}_{cls}^{proto} + \mathcal{L}_{cls}^{src} + \mathcal{L}_{cls}^{tgt}. \tag{14}$$

For the global prototypes and source samples, since their labels are available, two cross-entropy losses are employed for evaluation:

$$\mathcal{L}_{cls}^{proto} = \frac{1}{(M+1)K} \Bigg(\sum_{m=1}^{M} \sum_{k=1}^{K} \mathcal{L}_{ce}\big(p(c_k^{\mathcal{S}_m}), k\big) + \sum_{k=1}^{K} \mathcal{L}_{ce}\big(p(c_k^{\mathcal{T}}), k\big) \Bigg), \tag{15}$$

$$\mathcal{L}_{cls}^{src} = \frac{1}{M} \sum_{m=1}^{M} \Big(\mathbb{E}_{(x_i^{\mathcal{S}_m}, y_i^{\mathcal{S}_m}) \in \widehat{\mathcal{S}}_m} \mathcal{L}_{ce}\big(p(x_i^{\mathcal{S}_m}), y_i^{\mathcal{S}_m}\big) \Big), \tag{16}$$

where $\mathcal{L}_{ce}$ denotes the cross-entropy loss function, and $p(x)$ represents the classification probability of x.

For the target samples, it is desirable to make their predictions more deterministic, and thus an entropy loss is utilized for measurement:

$$\mathcal{L}_{cls}^{tgt} = -\mathbb{E}_{(x_i^{\mathcal{T}}, \widehat{y}_i^{\mathcal{T}}) \in \widehat{\mathcal{T}}} \sum_{k=1}^{K} p(\widehat{y}_i^{\mathcal{T}} = k | x_i^{\mathcal{T}}) \log p(\widehat{y}_i^{\mathcal{T}} = k | x_i^{\mathcal{T}}), \tag{17}$$

where $p(y = k|x)$ is the probability that x belongs to class k.

Overall objectives. Combining the classification and domain adaptation losses defined above, the overall objectives for feature extractor f and GCN model f_G are as follows:

$$\min_{f} \mathcal{L}_{cls} + \mathcal{L}_{RAL}, \qquad \min_{f_G} \mathcal{L}_{cls}. \tag{18}$$

4 Experiments

In this section, we first describe the experimental settings and then compare our model with existing methods on three Multi-Source Domain Adaptation datasets to demonstrate its effectiveness.

Table 1: Classification accuracy (mean ± std %) on *Digits-five* dataset.

Standards	Methods	→ mm	→ mt	→ up	→ sv	→ syn	Avg
Single Best	Source-only	59.2±0.6	97.2±0.6	84.7±0.8	77.7±0.8	85.2±0.6	80.8
	DAN [24]	63.8±0.7	96.3±0.5	94.2±0.9	62.5±0.7	85.4±0.8	80.4
	CORAL [39]	62.5±0.7	97.2±0.8	93.5±0.8	64.4±0.7	82.8±0.7	80.1
	DANN [5]	71.3±0.6	97.6±0.8	92.3±0.9	63.5±0.8	85.4±0.8	82.0
	ADDA [40]	71.6±0.5	97.9±0.8	92.8±0.7	75.5±0.5	86.5±0.6	84.8
Source Combine	Source-only	63.4±0.7	90.5±0.8	88.7±0.9	63.5±0.9	82.4±0.6	77.7
	DAN [24]	67.9±0.8	97.5±0.6	93.5±0.8	67.8±0.6	86.9±0.5	82.7
	DANN [5]	70.8±0.8	97.9±0.7	93.5±0.8	68.5±0.5	87.4±0.9	83.6
	JAN [27]	65.9±0.7	97.2±0.7	95.4±0.8	75.3±0.7	86.6±0.6	84.1
	ADDA [40]	72.3±0.7	97.9±0.6	93.1±0.8	75.0±0.8	86.7±0.6	85.0
	MCD [36]	72.5±0.7	96.2±0.8	95.3±0.7	78.9±0.8	87.5±0.7	86.1
Multi-Source	MDAN [51]	69.5±0.3	98.0±0.9	92.4±0.7	69.2±0.6	87.4±0.5	83.3
	DCTN [45]	70.5±1.2	96.2±0.8	92.8±0.3	77.6±0.4	86.8±0.8	84.8
	M^{3}SDA [33]	72.8±1.1	98.4±0.7	96.1±0.8	81.3±0.9	89.6±0.6	87.7
	MDDA [52]	78.6±0.6	98.8±0.4	93.9±0.5	79.3±0.8	89.7±0.7	88.1
	LtC-MSDA	**85.6**±0.8	**99.0**±0.4	**98.3**±0.4	**83.2**±0.6	**93.0**±0.5	**91.8**

4.1   Experimental Setup

Training details. For all experiments, a GCN model with two graph convolutional layers is employed, in which the dimension of feature representation is $d \rightarrow d \rightarrow K$ (d: the dimension of feature embedding; K: the number of classes). Unless otherwise specified, the trade-off parameters λ_1, λ_2 are set as 20, 0.001 respectively, and the standard deviation σ is set as 0.005. In addition, "$\rightarrow D$" denotes the task of transferring from other domains to domain D.

Performance comparison. We compare our approach with state-of-the-art methods to verify its effectiveness. For the sake of fair comparison, we introduce three standards. (1) *Single Best*: We report the best performance of single-source domain adaptation algorithm among all the sources. (2) *Source Combine*: All the source domain data are combined into a single source, and domain adaptation is performed in a traditional single-source manner. (3) *Multi-Source*: The knowledges learned from multiple source domains are transferred to target domain. For the first two settings, previous single-source UDA methods, *e.g.* DAN [24], JAN [27], DANN [5], ADDA [40], MCD [36], are introduced for comparison. For the *Multi-Source* setting, we compare our approach with four existing MSDA algorithms, MDAN [51], DCTN [45], M^{3}SDA [33] and MDDA [52].

4.2   Experiments on Digits-five

Dataset. Digits-five dataset contains five digit image domains, including MNIST (**mt**) [18], MNIST-M (**mm**) [5], SVHN (**sv**) [31], USPS (**up**) [13], and Synthetic Digits (**syn**) [5]. Each domain contains ten classes corresponding to digits ranging from 0 to 9. We follow the setting in DCTN [45] to sample the data.

Results. Table 1 reports the performance of our method compared with other works. Source-only denotes the model trained with only source domain data, which serves as the baseline. From the table, it can be observed that the proposed LtC-MSDA surpasses existing methods on all five tasks. In particular, a performance gain of 7.0% is achieved on the "$\rightarrow$ **mm**" task. The results

Table 2: Classification accuracy (%) on *Office-31* dataset.

Standards	Methods	$\rightarrow$ D	$\rightarrow$ W	$\rightarrow$ A	Avg
Single Best	Source-only	99.0	95.3	50.2	81.5
	RevGrad [4]	99.2	96.4	53.4	83.0
	DAN [24]	99.0	96.0	54.0	83.0
	RTN [26]	**99.6**	96.8	51.0	82.5
	ADDA [40]	99.4	95.3	54.6	83.1
Source Combine	Source-only	97.1	92.0	51.6	80.2
	DAN [24]	98.8	96.2	54.9	83.3
	RTN [26]	99.2	95.8	53.4	82.8
	JAN [27]	99.4	95.9	54.6	83.3
	ADDA [40]	99.2	96.0	55.9	83.7
	MCD [36]	99.5	96.2	54.4	83.4
Multi-Source	MDAN [51]	99.2	95.4	55.2	83.3
	DCTN [45]	**99.6**	96.9	54.9	83.8
	M^{3}SDA [33]	99.4	96.2	55.4	83.7
	MDDA [52]	99.2	97.1	56.2	84.2
	LtC-MSDA	**99.6**	**97.2**	**56.9**	**84.6**

demonstrate the effectiveness of our approach on boosting model's performance through integrating multiple domains' knowledges.

4.3 Experiments on Office-31

Dataset. Office-31 [35] is a classical domain adaptation benchmark with 31 categories and 4652 images. It contains three domains: Amazon (A), Webcam (W) and DSLR (D), and the data are collected from office environment.

    Results. In Table 2, we report the performance of our approach and existing methods on three tasks. The LtC-MSDA model outperforms the state-of-the-art method, MDDA [52], with 0.4% in the term of average classification accuracy, and a 0.7% performance improvement is obtained on the hard-to-transfer task, "$\rightarrow$ A". On this dataset, our approach doesn't have obvious superiority, which probably ascribes to two reasons. (1) First, domain adaptation models exhibit saturation when evaluated on "$\rightarrow$ D" and "$\rightarrow$ W" tasks, in which Source-only models achieve performance higher than 95%. (2) Second, the Webcam and DSLR domains are highly similar, which restricts the benefit brought by multiple domains' interaction in our framework, especially in "$\rightarrow$ A" task.

4.4 Experiments on DomainNet

Dataset. DomainNet [33] is by far the largest and most difficult domain adaptation dataset. It consists of around 0.6 million images and 6 domains: clipart (clp), infograph (inf), painting (pnt), quickdraw (qdr), real (rel) and sketch (skt). Each domain contains the same 345 categories of common objects.

    Results. The results of various methods on DomainNet are presented in Table 3. Our model exceeds existing works with a notable margin on all six tasks. In particular, a 4.2% performance gain is achieved on mean accuracy. The major challenges of this dataset are two-fold. (1) Large domain shift exists among different domains, *e.g.* from real images to sketches. (2) Numerous categories increase

Table 3: Classification accuracy (mean ± std %) on *DomainNet* dataset.

Standards	Methods	→ clp	→ inf	→ pnt	→ qdr	→ rel	→ skt	Avg
Single Best	Source-only	39.6±0.6	8.2±0.8	33.9±0.6	11.8±0.7	41.6±0.8	23.1±0.7	26.4
	DAN [24]	39.1±0.5	11.4±0.8	33.3±0.6	16.2±0.4	42.1±0.7	29.7±0.9	28.6
	JAN [27]	35.3±0.7	9.1±0.6	32.5±0.7	14.3±0.6	43.1±0.8	25.7±0.6	26.7
	DANN [5]	37.9±0.7	11.4±0.9	33.9±0.6	13.7±0.6	41.5±0.7	28.6±0.6	27.8
	ADDA [40]	39.5±0.8	14.5±0.7	29.1±0.8	14.9±0.5	41.9±0.8	30.7±0.7	28.4
	MCD [36]	42.6±0.3	19.6±0.8	42.6±1.0	3.8±0.6	50.5±0.4	33.8±0.9	32.2
Source Combine	Source-only	47.6±0.5	13.0±0.4	38.1±0.5	13.3±0.4	51.9±0.9	33.7±0.5	32.9
	DAN [24]	45.4±0.5	12.8±0.9	36.2±0.6	15.3±0.4	48.6±0.7	34.0±0.5	32.1
	JAN [27]	40.9±0.4	11.1±0.6	35.4±0.5	12.1±0.7	45.8±0.6	32.3±0.6	29.6
	DANN [5]	45.5±0.6	13.1±0.7	37.0±0.7	13.2±0.8	48.9±0.7	31.8±0.6	32.6
	ADDA [40]	47.5±0.8	11.4±0.7	36.7±0.5	14.7±0.5	49.1±0.8	33.5±0.5	32.2
	MCD [36]	54.3±0.6	22.1±0.7	45.7±0.6	7.6±0.5	58.4±0.7	43.5±0.6	38.5
Multi-Source	MDAN [51]	52.4±0.6	21.3±0.8	46.9±0.4	8.6±0.6	54.9±0.6	46.5±0.7	38.4
	DCTN [45]	48.6±0.7	23.5±0.6	48.8±0.6	7.2±0.5	53.5±0.6	47.3±0.5	38.2
	M^{3}SDA [33]	58.6±0.5	26.0±0.9	52.3±0.6	6.3±0.6	62.7±0.5	49.5±0.8	42.6
	MDDA [52]	59.4±0.6	23.8±0.8	53.2±0.6	12.5±0.6	61.8±0.5	48.6±0.8	43.2
	LtC-MSDA	**63.1**±0.5	**28.7**±0.7	**56.1**±0.5	**16.3**±0.5	**66.1**±0.6	**53.8**±0.6	**47.4**

the difficulty of learning discriminative features. Our approach tackles these two problems as follows. For the first issue, the global term of *Relation Alignment Loss* constrains the similarity between two arbitrary categories to be consistent on all domains, which encourages better feature alignment in the latent space. For the second issue, the local term of *Relation Alignment Loss* promotes the compactness of the same categories' features, which eases the burden of feature separation among different classes.

5   Analysis

In this section, we provide more in-depth analysis of our method to validate the effectiveness of major components, and both quantitative and qualitative experiments are conducted for verification.

5.1   Ablation Study

Effect of domain adaptation losses. In Table 4, we analyze the effect of global and local *Relation Alignment Loss* on Digits-five dataset.

On the basis of baseline setting (1st row), the global consistency loss (2nd rows) can greatly promote model's performance by promoting category-level domain alignment. For the local term, after adding it to the baseline configuration (3rd row), a 2.12% performance gain is achieved, which demonstrates the effectiveness of $\mathcal{L}_{RAL}^{local}$ on enhancing the separability of feature representations. Furthermore, the combination of $\mathcal{L}_{RAL}^{global}$ and $\mathcal{L}_{RAL}^{local}$ (4th row) obtains the best performance, which shows the complementarity of global and local constraints.

Effect of classification losses. Table 5 presents the effect of different classification losses on Digits-five dataset. The configuration of using only source samples' classification loss $\mathcal{L}_{cls}^{src}$ (1st row) serves as the baseline. After adding the entropy constraint for target samples (3rd row), the accuracy increases by

Table 4: Ablation study for domain adaptation losses on global and local levels.

$\mathcal{L}_{RAL}^{global}$	$\mathcal{L}_{RAL}^{local}$	$\rightarrow$ **mm**	$\rightarrow$ **mt**	$\rightarrow$ **up**	$\rightarrow$ **sv**	$\rightarrow$ **syn**	Avg
		74.85	98.60	97.95	74.56	88.54	86.90
✓		82.49	98.97	98.06	81.64	91.70	90.57
	✓	79.57	98.64	98.06	78.66	90.16	89.02
✓	✓	85.56	98.98	98.32	83.24	93.04	91.83

Table 5: Ablation study for three kinds of classification losses.

$\mathcal{L}_{cls}^{src}$	$\mathcal{L}_{cls}^{proto}$	$\mathcal{L}_{cls}^{tgt}$	$\rightarrow$ **mm**	$\rightarrow$ **mt**	$\rightarrow$ **up**	$\rightarrow$ **sv**	$\rightarrow$ **syn**	Avg
✓			73.65	98.47	96.61	78.20	88.93	87.17
✓	✓		78.44	98.64	96.77	79.24	89.05	88.43
✓		✓	81.36	98.76	97.93	81.26	91.70	90.20
✓	✓	✓	85.56	98.98	98.32	83.24	93.04	91.83

3.03%, which illustrates the effectiveness of $\mathcal{L}_{cls}^{tgt}$ on making target samples' features more discriminative. Prototypes' classification loss $\mathcal{L}_{cls}^{proto}$ is able to further boost the performance by constraining prototypes' distinguishability (4th row).

5.2   Sensitivity Analysis

Sensitivity of standard deviation σ. In this part, we discuss the selection of parameter σ which controls the sparsity of adjacency matrix. In Figure 3(a), we plot the performance of models trained with different σ values. The highest accuracy on target domain is achieved when the value of σ is around 0.005. Also, it is worth noticing that obvious performance decay occurs when the adjacency matrix is too dense or sparse, *i.e.* $\sigma > 0.05$ or $\sigma < 0.0005$.

   Sensitivity of trade-off parameters λ_1, λ_2. In this experiment, we evaluate our approach's sensitivity to λ_1 and λ_2 which trade off between domain adaptation and classification losses. Figure 3(b) and Figure 3(c) show model's performance under different λ_1 (λ_2) values when the other parameter λ_2 (λ_1) is fixed. From the line charts, we can observe that model's performance is not sensitive to λ_1 and λ_2 when they are around 20 and 0.001, respectively. In addition, performance decay occurs when these two parameters approach 0, which demonstrates that both global and local constraints are indispensable.

5.3   Visualization

Visualization of adjacency matrix. Figure 4(a) shows the adjacency matrix **A** before and after applying the *Relation Alignment Loss* (RAL), in which each pixel denotes the relevance between two categories from arbitrary domains. It can be observed that, after adding RAL, the relevance among various categories is apparently more consistent across different domains, which is compatible with the relational structure constrained by the global term of RAL.
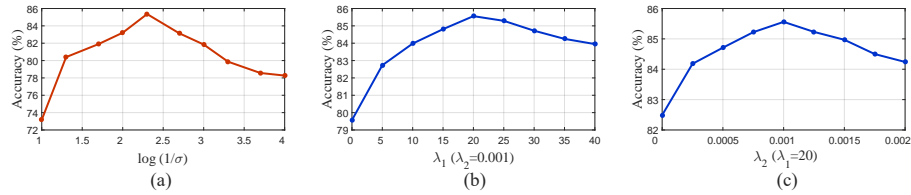
Fig. 3: Sensitivity analysis of standard deviation σ (left) and trade-off parameters λ_1, λ_2 (middle, right). (All results are reported on the "$\rightarrow$ **mm**" task.)

(a) Adjacency matrix **A** with and without RAL constraint.
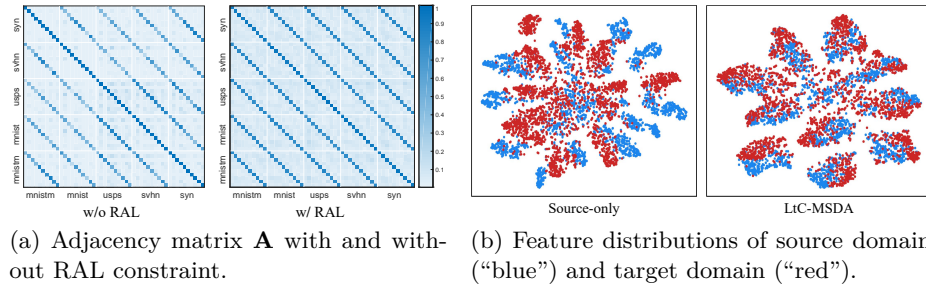
(b) Feature distributions of source domain ("blue") and target domain ("red").

Fig. 4: Visualization of adjacency matrix and feature embeddings. (All results are evaluated on the "$\rightarrow$ **mm**" task.)

Visualization of feature embeddings. In Figure 4(b), we utilize t-SNE [28] to visualize the feature distributions of one of source domains (SVHN) and target domain (MNIST-M). Compared with the Source-only baseline, the proposed LtC-MSDA model makes the features of target domain more discriminative and better aligned with those of source domain.

6   Conclusion

In this paper, we propose a Learning to Combine for Multi-Source Domain Adaptation (LtC-MSDA) framework. In this framework, the knowledges learned from multiple domains are aggregated to assist the prediction for query samples. Furthermore, we conduct class-relation-aware domain alignment via constraining global category relationships and local feature compactness. Extensive experiments and analytical studies demonstrate the prominent performance of our approach under various domain shift settings.

Acknowledgement

References

1. Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Wortman, J.: Learning bounds for domain adaptation. In: Advances in Neural Information Processing Systems (2007)
2. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
3. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. In: International Conference on Learning Representations (2015)
4. Ganin, Y., Lempitsky, V.S.: Unsupervised domain adaptation by backpropagation. In: International Conference on Machine Learning (2015)
5. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.S.: Domain-adversarial training of neural networks. Journal of Machine Learning Research **17**(1), 2096–2030 (2016)
6. Ghifary, M., Kleijn, W.B., Zhang, M., Balduzzi, D., Li, W.: Deep reconstruction-classification networks for unsupervised domain adaptation. In: European Conference on Computer Vision (2016)
7. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: IEEE Conference on Computer Vision and Pattern Recognition (2006)
8. Hakkani-Tür, D., Heck, L.P., Tür, G.: Using a knowledge graph and query click logs for unsupervised learning of relation detection. In: IEEE International Conference on Acoustics, Speech and Signal Processing (2013)
9. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.B.: Momentum contrast for unsupervised visual representation learning. In: IEEE Conference on Computer Vision and Pattern Recognition (2020)
10. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. In: IEEE International Conference on Computer Vision (2017)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
12. Hoffman, J., Mohri, M., Zhang, N.: Algorithms and theory for multiple-source adaptation. In: Advances in Neural Information Processing Systems (2018)
13. anJonathan J. Hull: A database for handwritten text recognition research. IEEE Transactions on pattern analysis and machine intelligence **16**(5), 550–554 (1994)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (2015)
15. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (2017)
16. Krishnamurthy, J., Mitchell, T.: Weakly supervised training of semantic parsers. In: Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. pp. 754–765 (2012)
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (2012)
18. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
19. Lee, C., Batra, T., Baig, M.H., Ulbricht, D.: Sliced wasserstein discrepancy for unsupervised domain adaptation. In: IEEE Conference on Computer Vision and Pattern Recognition (2019)

20. Lee, C., Fang, W., Yeh, C., Wang, Y.F.: Multi-label zero-shot learning with structured knowledge graphs. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
21. Liu, J., Ni, B., Li, C., Yang, J., Tian, Q.: Dynamic points agglomeration for hierarchical point sets learning. In: IEEE International Conference on Computer Vision (2019)
22. Liu, J., Ni, B., Yan, Y., Zhou, P., Cheng, S., Hu, J.: Pose transferrable person re-identification. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
23. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., Berg, A.C.: SSD: single shot multibox detector. In: European Conference on Computer Vision (2016)
24. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks. In: International Conference on Machine Learning (2015)
25. Long, M., Cao, Z., Wang, J., Jordan, M.I.: Conditional adversarial domain adaptation. In: Advances in Neural Information Processing Systems (2018)
26. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Unsupervised domain adaptation with residual transfer networks. In: Advances in Neural Information Processing Systems (2016)
27. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Deep transfer learning with joint adaptation networks. In: International Conference on Machine Learning (2017)
28. Maaten, L.V.D., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(2605), 2579–2605 (2008)
29. Mansour, Y., Mohri, M., Rostamizadeh, A.: Domain adaptation with multiple sources. In: Advances in Neural Information Processing Systems (2008)
30. Marino, K., Salakhutdinov, R., Gupta, A.: The more you know: Using knowledge graphs for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
31. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning. In: NIPS Workshops (2011)
32. Pan, Y., Yao, T., Li, Y., Wang, Y., Ngo, C., Mei, T.: Transferrable prototypical networks for unsupervised domain adaptation. In: IEEE Conference on Computer Vision and Pattern Recognition (2019)
33. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: IEEE International Conference on Computer Vision (2019)
34. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems (2015)
35. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: European Conference on Computer Vision (2010)
36. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
37. Sankaranarayanan, S., Balaji, Y., Castillo, C.D., Chellappa, R.: Generate to adapt: Aligning domains using generative adversarial networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
38. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: IEEE Conference on Computer Vision and Pattern Recognition (2015)

39. Sun, B., Saenko, K.: Deep CORAL: correlation alignment for deep domain adaptation. In: ECCV Workshop (2016)
40. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
41. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T.: Deep domain confusion: Maximizing for domain invariance. CoRR **abs/1412.3474** (2014)
42. Xie, S., Zheng, Z., Chen, L., Chen, C.: Learning semantic representations for unsupervised domain adaptation. In: International Conference on Machine Learning (2018)
43. Xu, M., Wang, H., Ni, B., Tian, Q., Zhang, W.: Cross-domain detection via graph-induced prototype alignment. In: IEEE Conference on Computer Vision and Pattern Recognition (2020)
44. Xu, M., Zhang, J., Ni, B., Li, T., Wang, C., Tian, Q., Zhang, W.: Adversarial domain adaptation with domain mixup. In: AAAI Conference on Artificial Intelligence (2020)
45. Xu, R., Chen, Z., Zuo, W., Yan, J., Lin, L.: Deep cocktail network: Multi-source unsupervised domain adaptation with category shift. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
46. Yan, H., Ding, Y., Li, P., Wang, Q., Xu, Y., Zuo, W.: Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
47. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: AAAI Conference on Artificial Intelligence (2018)
48. Yan, Y., Zhang, Q., Ni, B., Zhang, W., Xu, M., Yang, X.: Learning context graph for person search. In: IEEE Conference on Computer Vision and Pattern Recognition (2019)
49. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: Advances in Neural Information Processing Systems (2014)
50. Zhang, W., Ouyang, W., Li, W., Xu, D.: Collaborative and adversarial network for unsupervised domain adaptation. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
51. Zhao, H., Zhang, S., Wu, G., Moura, J.M.F., Costeira, J.P., Gordon, G.J.: Adversarial multiple source domain adaptation. In: Advances in Neural Information Processing Systems (2018)
52. Zhao, S., Wang, G., Zhang, S., Gu, Y., Li, Y., Song, Z.C., Xu, P., Hu, R., Chai, H., Keutzer, K.: Multi-source distilling domain adaptation. In: AAAI Conference on Artificial Intelligence (2020)