Graphical Modeling for Multi-Source Domain Adaptation

Minghao Xu*, Hang Wang*, Bingbing Ni†
*equal contribution   †corresponding author

Abstract—Multi-Source Domain Adaptation (MSDA) focuses on transferring the knowledge from multiple source domains to the target domain, which is a more practical and challenging problem compared to the conventional single-source domain adaptation. In this problem, it is essential to model multiple source domains and target domain jointly, and an effective domain combination scheme is also highly required. The graphical structure among different domains is useful to tackle these challenges, in which the interdependency among various instances/categories can be effectively modeled. In this work, we propose two types of graphical models, *i.e.* **C**onditional **R**andom **F**ield for MSDA (*CRF-MSDA*) and **M**arkov **R**andom **F**ield for MSDA (*MRF-MSDA*), for cross-domain joint modeling and learnable domain combination. In a nutshell, given an observation set composed of a query sample and the semantic prototypes (*i.e.* representative category embeddings) on various domains, the CRF-MSDA model seeks to learn the joint distribution of labels conditioned on the observations. We attain this goal by constructing a relational graph over all observations and conducting local message passing on it. By comparison, MRF-MSDA aims to model the joint distribution of observations over different Markov networks via an energy-based formulation, and it can naturally perform label prediction by summing the joint likelihoods over several specific networks. Compared to the CRF-MSDA counterpart, the MRF-MSDA model is more expressive and possesses lower computational cost. We evaluate these two models on four standard benchmark data sets of MSDA with distinct domain shift and data complexity, and both models achieve superior performance over existing methods on all benchmarks. In addition, the analytical studies illustrate the effect of different model components and provide insights about how the cross-domain joint modeling performs. Our code is available at https://github.com/Francis0625/Graphical-Modeling-for-Multi-Source-Domain-Adaptation.

Index Terms—Multi-Source Domain Adaptation, Graphical Model, Conditional Random Field, Markov Random Field

✦

1 INTRODUCTION

THE Unsupervised Domain Adaptation (UDA) methods [1], [2], [3], [4], [5], [6], [7] assume a single source domain with supervision and aim to transfer the knowledge acquired from the source domain to another unsupervised target domain. However, in real-world applications, it is unreasonable to assume that the labeled data are drawn from a single data distribution. Actually, these samples are always collected from different deployment environments, *i.e.* from multiple domains. For example, in an object classification task, one may have access to the annotated images captured in the morning, afternoon and evening, respectively, and the objective is to categorize the images captured at dawn. In addition, the diversity of weather, illumination and backgrounds can all lead to the existence of multiple domains in a specific data set. The problem under such a scenario is known as *Multi-Source Domain Adaptation* (MSDA) [8], in which one seeks to boost the model's performance on target domain by integrating the transferrable knowledge from various source domains. By employing MSDA algorithms' power of aligning multiple domains, we can better handle various real-world applications involving changing deployment environments, *e.g.* autonomous driving and intelligent surveillance.

Following the theoretical guarantee that the target distribution can be effectively approximated by the weighted combination of multiple source distributions [8], [9], recent works [10], [11], [12], [13] attempted to tackle the classification-based MSDA problem through aligning the feature distributions between source and target domains (or across different source domains) and combining the predictions of several domain-specific classification models. The core idea of these methods is to approach the conditional distribution of semantic label on target domain (*i.e.* $p_{\mathcal{T}}(y|x)$) with the mixture of the conditional distributions learned for multiple source domains. Specifically, given a sample from target domain, these methods first derive the probability of its corresponding label using the classifiers trained for each source domain and then combine all predictions via a weighted average. Although such scheme is effective on several benchmark data sets of MSDA, its expressivity is still limited for the lack of following two important model capabilities.

1) **Joint modeling across different domains.** The existing methods typically learn the conditional distribution of label on each domain in an independent way, which only model the dependency of label prediction on the statistics specific to a single domain. As a matter of fact, the interdependency between the statistics of various domains can also benefit the inference of a sample's semantic label. For instance, according to the similarity of a category-specific statistic, the correlated categories of different domains can be linked to each other, such that the cross-domain de-

- *All authors are with Shanghai Jiao Tong University, Shanghai 200240, China. H. Wang and B. Ni are also with Huawei Hisilicon.*
  E-mail: {xuminghao118, Wang--Hang, nibingbing}@sjtu.edu.cn
- * *Authors contributed equally to this work.*
- † *Corresponding author: Bingbing Ni.*

pendencies between these correlated categories can derive more precise predictions (*e.g.* if an image is to be classified as vehicle, it should possess sufficient similarities with the vehicles and other related categories on various domains). Therefore, it is desirable to devise a unified model which can effectively capture the joint dependencies between a query sample and all the source and target domains.

2) **Learnable domain combination.** In most existing works, the domain combination is commonly attained by the weighted average using hand-craft or model-induced weights. In these methods, after learning the classification model for each source domain, the inference on a sample from target domain is performed by combining the predictions of different models according to the similarity scores of various source-target domain pairs. Such combination scheme relies on the heuristics of domain relations and is not learnable along with the model. It is more favorable to learn the domain combination from the data, in which the combination component of the model is directly optimized according to the learning objective. In this way, the model can better represent the relations between different domains under the guidance of the data.

We would like to point out that the *graphical structure* among various domains is informative to address the problems above. Specifically, the scope of valid joint distributions can be explicitly specified by a graphical structure, and such structure also enables learnable message passing across different domains. Motivated by these facts, in this work, we explore two types of graphical models, *i.e.* **C**onditional **R**andom **F**ield for MSDA (*CRF-MSDA*) and **M**arkov **R**andom **F**ield for MSDA (*MRF-MSDA*)[1]. For joint modeling across various domains, both models introduce an additional set of random variables, named as prototypes [14], [15], [16], which serve as the representative embeddings of the semantic categories on all domains. On such basis, these two models learn two kinds of distributions over query sample and prototypes, in which the domain combination is intrinsically included and is thus learnable along with the whole model. These two graphical models are defined as follows.

CRF-MSDA seeks to model the conditional distribution of label for a query sample and all prototypes simultaneously. In specific, we first construct a graph over the query sample and the prototypes of different domains, in which the connection weight between two nodes is determined by the similarity of their features. We then employ a graph neural network (GNN) to propagate the local messages on the graph and use a linear classifier to predict the label of each node. During the learning phase, a global constraint is employed for the category-level alignment between different domains, and a local constraint is applied to promote the feature compactness surrounding the prototypes. In this model, the domain combination is achieved by the message passing between the prototypes of various domains, and

such combination can be learned along with the GNN.

MRF-MSDA aims to model the joint distribution of a query sample and all prototypes conditioned on a Markov network over them. For the MSDA problem, we consider a positive Markov network where all the prototypes belonging to the same category are connected, and the query sample is linked to the prototype associated to its corresponding domain and category. Also, some negative networks are derived by modifying the edges of the positive one. We optimize the joint distributions specified by various Markov networks through contrasting all the positive networks in a mini-batch with all negative ones. In this way, the embedding of query sample is encouraged to be similar with the prototypes of its corresponding category and dissimilar with the ones of other categories. On such basis, we derive the classification probability for a query sample by summing the joint likelihoods over several specific Markov networks which link the query sample to the prototypes within the same category but from different domains. Such scheme attains domain combination, and it can be learned with the supervision from ground-truth labels. Compared to CRF-MSDA, the learning of MRF-MSDA involves multiple Markov networks (*i.e.* positive and negative ones) for a single query sample, and thus more relational patterns between the query sample and prototypes can be learned. This property endows MRF-MSDA with stronger model expressivity.

Compared to the conference paper [17], this journal work makes the following additional contributions:

- We explicitly point out two important capabilities of an MSDA model, *i.e.* the joint modeling across different domains and the learnable domain combination.
- We re-organize the LtC-MSDA approach proposed in the conference paper under the framework of CRF, deriving the CRF-MSDA model.
- We novelly design a model that fully owns the two capabilities above. This model is designed based on the philosophy of MRF, called MRF-MSDA. Compared with CRF-MSDA that only models the dependency among labels, MRF-MSDA can jointly capture the dependency among observations and labels.
- We experimentally verify the superior performance of MRF-MSDA over CRF-MSDA, and MRF-MSDA establishes a new state-of-the-art on multiple MSDA benchmarks.

2 RELATED WORK

Unsupervised Domain Adaptation (UDA). UDA aims to generalize a model learned from a labeled source domain to another target domain without labels. Some previous methods attempts to narrow the domain shift between source and target domains via minimizing an explicit domain discrepancy metric, *e.g.* Maximum Mean Discrepancy (MMD) [18], [19], Weighted MMD [20], Multi-Kernel MMD [1], [21] and Wasserstein Distance [22], [23], [24]. Also, aligning the second-order statistics is explored in [3] to restrict the domain-invariance between two domains. Another group of methods perform adaptation by employing adversarial learning to align the source and target domains. Among these approaches, a domain discriminator is introduced to

---

1. Note that, the CRF-MSDA and MRF-MSDA models differ from the conventional CRFs and MRFs, which are parameterized by local potential functions on subgraphs. Our models are instead parameterized by highly expressive deep neural networks. However, they share the similar working mechanism with these conventional methods on both graphical and probabilistic modeling, and are thus named after CRF and MRF.

encourage domain-invariant features [2], [4], [5], [25], [26], [27], [28]. On par with the feature-level adaptation, generative models conduct distribution alignment on pixel level by image translation [29], [30], style transfer [31] or image generation [32], [33], [34], [35]. Cycle-consistency is also constrained to enforce the consistency of relevant semantics during distribution alignment [36], [37], [38]. Recently, a group of approaches performs category-level domain adaptation through utilizing dual classifier [6], [23], [39], domain prototype [15], [40], [41] or pseudo labels of target data [42], [43], [44], [45]. There are also other domain adaptation approaches that focus on designing model components for domain transfer [46] and exploring the transferability of label predictions [7], [47], [48].

To better exploit the structural dependency between the samples/categories of source and target domain, some existing methods [49], [50], [51] propose to construct relational graphs between two domains and perform domain alignment upon such inter-domain graphs. However, all these methods aim to tackle the UDA problem and cannot be trivially transferred to the setting with multiple source domains. By comparison, our work studies the graphical models upon multiple source domains and a target domain.

Multi-Source Domain Adaptation (MSDA). In comparison with the conventional single-source domain adaptation, MSDA assumes data are collected from multiple source domains with different distributions, which is a more practical and difficult scenario. Early theoretical analysis [8], [52] gave strong guarantees for representing target distribution as the weighted combination of source distributions to address the MSDA problem. Based on these works, Hoffman *et al.* [9] derived normalized solutions to determine the distribution-weighted combination. Recently, Zhao *et al.* [10] proposed to align target domain to multiple source domains globally by adversarial learning. Xu *et al.* [11] deployed multi-way adversarial learning and combined source-specific perplexity scores for target predictions. Peng *et al.* [12] introduced the idea of matching the high-order moments between domain-specific feature representations. In [13], source distilling mechanism is designed to fine-tune the separately pre-trained feature extractor and classifier. CMSS [53] designed a dynamic curriculum to iteratively select the best source samples for aligning to the target. Li *et al.* [54] enhanced model's domain adaptation performance by meta-learning.

Improvements over existing methods. Previous works [10], [11], [12], [13] typically model the conditional distribution of semantic label on each domain in an independent way, and the label predictions from these domain-specific models are further combined to approach the conditional distribution on target domain. In contrast, in this work, we explore the joint modeling across all source and target domains. The domain combination is intrinsically contained in the proposed CRF-MSDA/MRF-MSDA model and thus can be learned in a joint fashion.

Conditional Random Field (CRF) for Vision. CRFs are a class of probabilistic graphical modeling methods in which a set of observed variables X and another set of unobserved ones Y are considered, and these methods aim to model the conditional distribution $p(Y|X)$ utilizing the structure information among different variables. The concept of CRF was first proposed by Lafferty *et al.* [55] and applied to the field of segmenting and labeling text sequences, in which the label prediction on each observation well depends on the results of previous steps. Because of the strong capability of learning and inference on structured data, CRF-based approaches have been widely explored on various computer vision problems involved structured prediction, *e.g.* segmentation [56], [57], [58], image denoising [59], [60], stereo reconstruction [61], [62] and super-resolution [63], [64]. These approaches mainly utilize the interrelationships among adjacent pixels/super-pixels. By comparison, for the MSDA task, we focus on the interdependencies among the semantic categories on different domains, and a CRF-MSDA model is proposed to perform structured prediction. In addition, compared to previous works, the CRF model established in this work is defined on the latent space instead of upon input images.

Markov Random Field (MRF) for Vision. MRF is a probabilistic graphical model for joint distribution modeling over a set of random variables, which defines a family of joint distributions that can be factorized upon an undirected graph. MRFs were first introduced into the vision field by the work of Geman and Geman [65], and their proposed MRF framework can express a wide variety of spatially varying priors, which is proved to benefit the image restoration task. Due to the effectiveness on capturing the interdependencies existing in different components of the data, MRF-based methods have been successfully adapted to many computer vision problems such as image restoration [66], [67], segmentation [68], [69], [70], texture analysis [71] and optical flow prediction [72], [73], in which obvious performance gain has been observed. In these works, a fixed Markov network is commonly adopted to model the joint distribution of different variables. By comparison, in the proposed MRF-MSDA method, we consider multiple Markov networks for each set of observations, so that more relational patterns can be explored. Furthermore, MRF is generally employed as a generative model for approximating the data distribution, while our MRF-MSDA model can be naturally used for discriminative modeling by summing the joint likelihoods over several specific Markov networks.

3 CONDITIONAL RANDOM FIELD FOR MULTI-SOURCE DOMAIN ADAPTATION

3.1 Problem Definition

In Multi-Source Domain Adaptation (MSDA), there are M source domains $\mathcal{S}_1$, $\mathcal{S}_2$, $\cdots$, $\mathcal{S}_M$. The source domain $S_m = \{(x_i^{\mathcal{S}_m}, y_i^{\mathcal{S}_m})\}_{i=1}^{N_{\mathcal{S}_m}}$ contains $N_{\mathcal{S}_m}$ *i.i.d.* labeled samples, where $x_i^{\mathcal{S}_m}$ follows the source distribution $p_{\mathcal{S}_m}(x)$ and $y_i^{\mathcal{S}_m} \in \{1, 2, \cdots, K\}$ (K is the number of categories) denotes its corresponding label. Similarly, the 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 the target distribution $p_{\mathcal{T}}(x)$. In addition, on all the source and target domains, we define a prototype (*i.e.* a representative feature embedding) for each category, denoted as $\mathbb{C} = \{\{c_k^m\}_{k=1}^K\}_{m=1}^{M+1}$, where target domain is regarded as the $(M + 1)$-th domain in this notation.
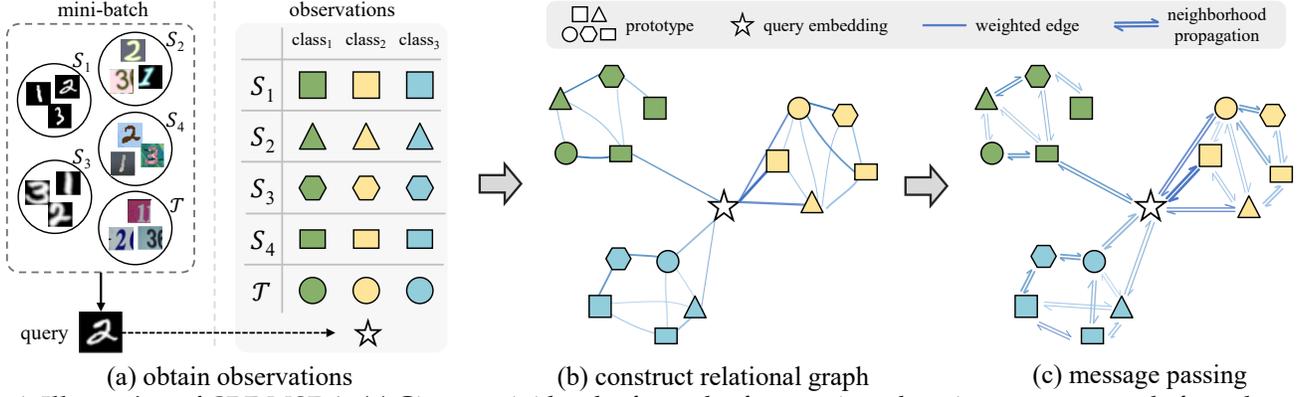
Fig. 1: **Illustration of CRF-MSDA.** (a) Given a mini-batch of samples from various domains, a query sample from the mini-batch together with all prototypes serve as the observations. (b) A relational graph is constructed over the observations. Note that, this graph should be fully-connected, while we omit some edges in it for better visualization. (c) The labels of all observations are predicted via the local message passing on the graph.

Given a query sample q from an arbitrary domain, the Conditional Random Field for MSDA (*CRF-MSDA*) considers an observed variable set X and an output variable set Y. The query sample's embedding z_q and all prototypes are deemed as observed variables, *i.e.* $X = \{z_q, c_1^1, \cdots, c_K^{M+1}\}$, and the semantic labels of these observations serve as the outputs, *i.e.* $Y = \{y_q, y_1^1, \cdots, y_K^{M+1}\}$. CRF-MSDA aims to model the conditional distribution $p(Y|X)$, in which a graph is constructed over the observed variables and their labels are predicted based on the local message passing on the graph. A graphical illustration of CRF-MSDA is shown in Fig. 1. Next, we introduce the detailed learning and inference scheme of the CRF-MSDA approach.

3.2 Model Learning

The CRF-MSDA model seeks to learn the conditional distribution of labels for the observed variables defined above. Specifically, for each learning step, a mini-batch of query samples from various domains are given, and these samples are mapped to the latent space by a feature extractor to update prototypes. After that, we structure each query sample and all prototypes as a graph, and a GNN is employed to perform local message propagation on this graph, which derives the feature representations combining the information from different domains for the observations. Upon these representations, a linear classifier predicts the categorical probability for each observed variable, and the ground-truth labels are used for supervision. In addition, we further introduce a global and a local constraint for domain alignment and feature compactness, respectively. The details are presented in the following parts.

3.2.1 Prototype Maintenance

During the learning phase, the prototypes are updated by the sampled mini-batches to better represent the data. Specifically, for each learning step, we sample a mini-batch B constituted by sets of query samples from all the source and target domains, *i.e.* $B = \{\widehat{\mathcal{S}}_1, \widehat{\mathcal{S}}_2, \cdots, \widehat{\mathcal{S}}_M, \widehat{\mathcal{T}}\}$, and the estimations of prototypes are derived on this mini-batch. For the source domain $\mathcal{S}_m$ ($1 \leqslant m \leqslant M$), the estimated prototype $\widehat{c}_k^m$ is defined as the mean embedding of all samples belonging to class k in the query sample set $\widehat{\mathcal{S}}_m$:

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

where $\widehat{\mathcal{S}}_m^k$ is the set of all samples belonging to class k in $\widehat{\mathcal{S}}_m$, and f stands for the feature extractor which maps an image to a low-dimensional embedding vector.

For the target domain $\mathcal{T}$, since the ground-truth label is unavailable, we first assign pseudo labels for the samples in $\widehat{\mathcal{T}}$ via the pseudo labeling strategy proposed by [42], and the estimated prototype $\widehat{c}_k^{M+1}$ for class k on target domain is defined as below:

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

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 the k-th category in $\widehat{\mathcal{T}}$.

Using these mini-batch-induced estimations, we update the prototypes on various domains through an exponential moving average scheme:

$$c_k^m \leftarrow \beta c_k^m + (1-\beta)\widehat{c}_k^m, \quad m = 1, 2, \cdots, M+1, \quad (3)$$

where β denotes the exponential decay rate, and it is fixed as 0.7 in all experiments. Such maintenance strategy can suppress the variance introduced by mini-batch sampling and derive smoother prototype estimations. In the literature [15], [74], [75], similar strategies have been explored to stabilize the learning process via smoother global variables.

3.2.2 Graphical Modeling

In the CRF-MSDA model, we predict the labels of observed variables under the context determined by a graph, which models the conditional distribution $p(Y|X)$. In specific, for a query sample $q \in B$, we define the observed variable set with its embedding $z_q = f(q)$ and all prototypes, *i.e.* $X = \{z_q, c_1^1, \cdots, c_K^{M+1}\}$, and these observations are further structured as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. In this graph, the node set $\mathcal{V}$ is identical to X in which all nodes are represented by the embedding vectors with the same dimension, and the edge set $\mathcal{E} = \{(u, v, A_{uv})\}$ describes the relations among observations, where A_{uv} denotes the adjacency weight between

node u and v. In practice, we derive the adjacency weight A_{uv} by applying a radial basis function (RBF) kernel $\mathcal{K}$ upon the embeddings of two nodes:

$$A_{uv} = \mathcal{K}(X_u, X_v) = \exp\Big(-\frac{||X_u - X_v||_2^2}{2\sigma^2}\Big), \qquad (4)$$

where X_u and X_v stand for the embedding of node u and v, and σ is the bandwidth parameter. Note that, the adjacency weights between all node pairs form the adjacency matrix of the graph, *i.e.* $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$.

Based on such a graph, we seek to learn effective node representations aggregating the information from neighbors and perform label prediction in a factorized way:

$$p(Y|X) = \prod_{v \in \mathcal{V}} p(y_v|X). \qquad (5)$$

Following the above formulation, a Graph Neural Network (GNN) g is employed to produce node representations by propagating messages among different nodes, and, over these representations, a linear classifier c outputs the classification probability for each node. In specific, the label of node v is predicted as below:

$$\mathbf{H} = g(\mathcal{G}, \mathbf{A}), \quad \hat{y}_v = p(y_v|X) = c(h_v), \qquad (6)$$

where $\mathbf{H} \in \mathbb{R}^{|\mathcal{V}| \times d}$ are the representations of all nodes produced by GNN (d indicates the dimensionality), h_v is the representation of node v, and $\hat{y}_v$ denotes the label prediction for that node.

3.2.3 Learning Objectives

For model learning, we aim to promote the discriminability and domain-invariance of feature representations, and these two objectives are pursued by constraining two kinds of objective functions for classification and alignment, respectively. The details are stated as follows.

Classification constraints. We define several classification constraints over the label predictions to enhance features' discriminability. In the observation set X, the prototypes are labeled by their corresponding category (*e.g.* the prototype c_k^m belongs to class k), which defines the following cross-entropy objective function:

$$\mathcal{L}_{cls}^{proto} = -\frac{1}{(M+1)K} \sum_{m=1}^{M+1} \sum_{k=1}^{K} \log\big(\hat{y}_k^m[k]\big), \qquad (7)$$

where $\hat{y}_k^m[k]$ denotes the classification probability of prototype c_k^m for the k-th category, and this class prediction is performed upon the post-GNN representation of prototype c_k^m, which better represents its corresponding semantic category via message passing. For the query sample q, when it is from source domains, the ground-truth label y_q is available. Using all the source domain samples in mini-batch B as query, we derive the supervised objective function for source domain as below:

$$\mathcal{L}_{cls}^{src} = -\frac{1}{M} \sum_{m=1}^{M} \Big(\mathbb{E}_{(q,y_q) \in \widehat{\mathcal{S}}_m} \log\big(\hat{y}_q[y_q]\big)\Big), \qquad (8)$$

where $\hat{y}_q[y_q]$ stands for the query sample's classification probability for the category specified by its label. In another case, when the query sample is drawn from the target domain, we cannot access the ground-truth annotation.

Therefore, we resort to an entropy-induced constraint which is able to facilitate more deterministic predictions on the samples from target domain:

$$\mathcal{L}_{cls}^{tgt} = -\mathbb{E}_{q \in \widehat{\mathcal{T}}} \sum_{k=1}^{K} \hat{y}_q[k] \log\big(\hat{y}_q[k]\big). \qquad (9)$$

For correctly classifying the nodes of various graphs established with different query samples from the mini-batch, the overall classification objective function is composed of three terms for prototypes, source domain queries and target domain queries, respectively:

$$\mathcal{L}_{cls} = \mathcal{L}_{cls}^{proto} + \mathcal{L}_{cls}^{src} + \mathcal{L}_{cls}^{tgt}. \qquad (10)$$

Alignment constraints. Besides pursuing feature discriminability, we also expect the feature distributions of various domains to be invariant, and, especially, such domain-invariance is better to be attained on category-level. Formally, the marginal distribution of the samples from the source and target domain can be expressed as the summation over the conditional distributions associated to different categories:

$$p_{\mathcal{S}_m}(x) = \sum_{y \in \mathbb{Y}} p_{\mathcal{S}_m}(y) \, p_{\mathcal{S}_m}(x|y), \qquad (11)$$

$$p_{\mathcal{T}}(x) = \sum_{y \in \mathbb{Y}} p_{\mathcal{T}}(y) \, p_{\mathcal{T}}(x|y), \qquad (12)$$

where $\mathbb{Y}$ denotes the set of all categories. Under the assumption that the marginal distribution of category $p(y)$ is identical across different domains (*i.e.* the proportions of the samples from various categories are domain-invariant), the goal is to align each conditional distribution $p(x|y)$ ($y \in \mathbb{Y}$) over all domains. To realize such a goal, we pursue the category-level domain alignment on the global level of the latent space, and the feature compactness surrounding various prototypes is constrained from a local point of view.

For the global objective, we expect the relevance between two arbitrary categories to be consistent on all domains. Specifically, we extract the first $(M+1)K$ rows and columns of the adjacency matrix, denoted as $\widetilde{\mathbf{A}} = \mathbf{A}_{1:(M+1)K}^{1:(M+1)K}$, where the block matrix $\widetilde{\mathbf{A}}_{i,j} = \widetilde{\mathbf{A}}_{(j-1)K+1:jK}^{(i-1)K+1:iK}$ ($1 \leqslant i,j \leqslant M+1$) measures all categories' relevance between the i-th and j-th domain. When various domains are well aligned on category level, these block matrices should be similar to each other, which leads to the following objective function for domain alignment:

$$\mathcal{L}_{global} = \frac{1}{(M+1)^4} \sum_{i,j,m,n=1}^{M+1} ||\widetilde{\mathbf{A}}_{i,j} - \widetilde{\mathbf{A}}_{m,n}||_F, \qquad (13)$$

where $||\cdot||_F$ is the Frobenius norm. In this function, the intra-class invariance is boosted by the constraints on block matrices' main diagonal elements, and the consistency of inter-class relationships is promoted by the constraints on other elements of block matrices.

For the local objective, we expect the query samples to be compactly embedded around their corresponding prototypes, which eases the category-level alignment by deriving more separated features among distinct categories. In specific, we constrain the embeddings of the samples

in mini-batch B with the following objective function for feature compactness:

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

(14)

where $\widehat{\mathcal{S}}_m^k$ ($1 \leqslant m \leqslant M$) and $\widehat{\mathcal{T}}_k$ represent the samples belonging to class k in the sample set $\widehat{\mathcal{S}}_m$ and $\widehat{\mathcal{T}}$, respectively.

Overall learning objective. Combining the classification and alignment constraints, the overall learning objective with respect to feature extractor f, GNN g and classifier c is defined as below:

$$\min_{f,g,c} \mathcal{L}_{cls} + \lambda_1 \mathcal{L}_{global} + \lambda_2 \mathcal{L}_{local},$$

(15)

where λ_1 and λ_2 are the trade-off parameters balancing among different learning objectives.

3.3 Model Inference

After the learning phase, we store the feature extractor f, GNN model g, linear classifier c and all prototypes $\mathbb{C} = \{\{c_k^m\}_{k=1}^K\}_{m=1}^{M+1}$. In the inference phase, given a query sample q, we first extract its embedding z_q with the extractor f and combine the embedding with all prototypes to form the observation set $X = \{z_q, c_1^1, \cdots, c_K^{M+1}\}$. After that, following the scheme in Sec. 3.2.2, a graph $\mathcal{G}$ is constructed over the observations. Upon this graph, the GNN g and linear classifier c are consecutively applied to derive the label predictions for all nodes. Finally, we take the prediction for the node corresponding to the query sample as the output.

4 MARKOV RANDOM FIELD FOR MULTI-SOURCE DOMAIN ADAPTATION

4.1 Problem Definition

In this model, the definitions of source and target domains and prototypes follow those in Sec. 3.1. Unlike the CRF-MSDA model, given a query sample q from an arbitrary domain, the Markov Random Field for MSDA (*MRF-MSDA*) seeks to model the joint distribution of all observed variables (*i.e.* the query sample's embedding and all prototypes) conditioned on a Markov network $\mathcal{G}$, denoted as $p(X|\mathcal{G}) = p(z_q, c_1^1, \cdots, c_K^{M+1}|\mathcal{G})$. Over all observations, a positive Markov network is formed to depict the desired interdependency among them. Specifically, all the prototypes belonging to the same category are connected, and the query sample is linked to the prototype associated to its corresponding domain and category. In addition, through modifying some edges in the positive network, the negative Markov networks are derived for comparison. Learning over these different networks guides the model to connect the query sample with correlated prototypes and thus enables label prediction. A graphical illustration of MRF-MSDA is presented in Fig. 2. The detailed learning and inference schemes are stated in the following sections.

4.2 Model Learning

Over a set of observations, the MRF-MSDA model is expected to be able to discriminate the positive Markov network from the negative ones through joint distribution modeling, and it can be further utilized for label prediction by summing the joint likelihoods over several specific Markov networks that link the query sample to the prototypes within a category. Specifically, we represent the joint distribution of observations on a specific Markov network with an energy-based formulation, and the joint distributions for various Markov networks are learned via Noise Contrastive Estimation (NCE) [76], [77]. Furthermore, the ground-truth labels of query samples are employed to supervise the joint-likelihood-induced label predictions. Instead of being updated via moving average as in CRF-MSDA, the prototypes in MRF-MSDA serve as model parameters and are learned along with the whole model. Next, we elucidate the details of model learning.

4.2.1 Graphical Modeling

Joint distribution modeling. In the MRF-MSDA model, the joint distributions of observations are modeled over various Markov networks. Specifically, for a query sample q, its embedding $z_q = f(q)$ together with all prototypes serve as the observed variables, *i.e.* $X = \{z_q, c_1^1, \cdots, c_K^{M+1}\}$. Note that, MRF-MSDA model uses a CNN encoder f to map the query sample q to a lower-dimensional embedding z_q, while the prototypes in this model are represented by learnable embedding vectors $\{c_1^1, \cdots, c_K^{M+1}\}$ following conventional graph embedding methods [78], [79], [80]. Over these observations, it is expected that the prototypes within a same category are interrelated, and the query sample is most relevant to the prototype associated to its corresponding domain and category, which defines a positive Markov network $\mathcal{G}^+ = (\mathcal{V}, \mathcal{E}^+)$. In this network, the node set $\mathcal{V}$ is identical to the observation set X where the embeddings of all nodes are with the same dimension, and the edge set $\mathcal{E}^+ = \{(u, v)\}$ reflects the desired relationships among observations as stated above. We graphically illustrate the structure of $\mathcal{G}^+$ for an arbitrary query in Fig. 2(b). Based on the positive network $\mathcal{G}^+$, we randomly modify some edges in it to further construct N_{neg} negative Markov networks $\{\mathcal{G}_n^- = (\mathcal{V}, \mathcal{E}_n^-)\}_{n=1}^{N_{neg}}$ (the details about the edge modification scheme are stated in Sec. 5.1). Upon a specific Markov network $\mathcal{G}$, we use an energy-based formulation to define the joint likelihood of the observations as follows:

$$p(X|\mathcal{G}) = \frac{1}{Z} \exp\left(-f_E(X, \mathcal{G})\right),$$

(16)

$$f_E(X, \mathcal{G}) = \frac{1}{\tau} \sum_{(u,v) \in \mathcal{E}} ||X_u - X_v||_2^2,$$

(17)

where Z stands for the partition function, τ denotes the temperature parameter, and X_u and X_v represent the embeddings of node u and v (these two nodes are connected in network $\mathcal{G}$). The energy function f_E sums up the energies on all edges of the network. Using such joint likelihood definition, we perform model learning based on maximum likelihood estimation (MLE), and the concrete learning objective is introduced in Sec. 4.2.2.
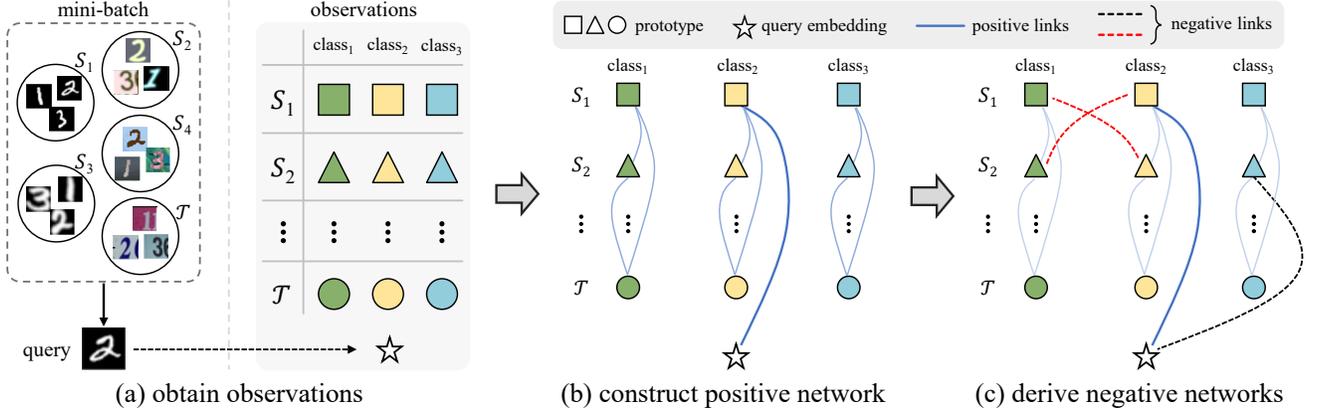
Fig. 2: **Illustration of MRF-MSDA.** (a) A query sample from the mini-batch and all prototypes serve as the observations. (b) A positive Markov network is constructed to connect the prototypes within the same category and the corresponding query-prototype pair. (c) Negative Markov networks are derived by randomly modifying some edges in the positive network. The model learns the correct connection through contrasting the positive network with negative ones, which enables better label prediction.

Joint-likelihood-induced label prediction. Considering the semantics underlying the observed variables, we propose to derive the classification probability of query sample using the joint likelihoods defined over several specific Markov networks. For example, we consider the case that only the prototypes within the same category are interrelated, and the query sample q belongs to class k and is from the m-th domain ($1 \leqslant m \leqslant M+1$), where the target domain is regarded as the $(M+1)$-th domain. The Markov network corresponding to such case is denoted as $\mathcal{G}_k^m$, in which K cliques are formed among the prototypes of K categories (*i.e.* all the prototypes within a category are connected to each other), and the query sample is linked to prototype c_k^m. Using the joint likelihood of observations over $\mathcal{G}_k^m$, we define the probability that query sample q is from the k-th category of the m-th domain as follows:

$$p(y_d = m, y = k|q) = \frac{1}{\mathcal{N}} \, p(X|\mathcal{G}_k^m), \quad (18)$$

$$\mathcal{N} = \sum_{m=1}^{M+1} \sum_{k=1}^{K} p(X|\mathcal{G}_k^m), \quad (19)$$

where the random variable y_d represents the domain label, and $\mathcal{N}$ is the normalizing constant. Through summing the probability $p(y_d = m, y = k|q)$ over all domains, we derive the classification probability of query q on class k as below:

$$\hat{y}_q[k] = p(y = k|q) = \sum_{m=1}^{M+1} p(y_d = m, y = k|q). \quad (20)$$

4.2.2 Learning Objectives

For learning the MRF-MSDA model, we aim at boosting the model's discriminative capability for label prediction and also maximizing the likelihoods on positive Markov networks while minimizing those on negative ones. These two learning objectives are pursued by classification constraints and maximum likelihood estimation (MLE), respectively. Detailed approaches are introduced as follows.

Classification constraints. We utilize two classification constraints to enhance model's discriminability on both source and target domains. In specific, for each learning

step, we draw a mini-batch of query samples from source and target domains, denoted as $B = \{\widehat{\mathcal{S}}_1, \widehat{\mathcal{S}}_2, \cdots, \widehat{\mathcal{S}}_M, \widehat{\mathcal{T}}\}$. Considering the unavailability of the ground-truth labels on target domain, we follow the formulations in Eqs. 8 and 9 to obtain a supervised constraint $\mathcal{L}_{cls}^{src}$ for source domain and a label-free constraint $\mathcal{L}_{cls}^{tgt}$ for target domain. For the discriminative modeling on these two kinds of domains, the overall classification objective function combines two constraints as below:

$$\mathcal{L}_{cls} = \mathcal{L}_{cls}^{src} + \mathcal{L}_{cls}^{tgt}. \quad (21)$$

Maximum likelihood estimation (MLE). Except for the discriminative modeling, we also expect that the model is able to identify the correct interrelationships among observations. We pursue such goal through enhancing the likelihoods for positive Markov networks and diminishing those for negative networks. This scheme guides the model to assign higher likelihoods to the networks that connect the query sample with correlated prototypes, which can benefit label prediction.

However, it is hard to directly optimize with the joint likelihood defined in Eq. 16 due to the intractability of evaluating the partition function exactly. As a substitute, inspired by the idea of Noise Contrastive Estimation (NCE) [76], [77], we propose to optimize upon the unnormalized joint likelihood, *i.e.* $\tilde{p}(X|\mathcal{G}) = \exp\big(-f_E(X, \mathcal{G})\big)$, by contrasting the positive Markov network with the negative ones. In practice, for constructing positive networks for the query samples from target domain, we again adopt the pseudo labeling scheme proposed by [42] to assign pseudo labels to the samples in $\widehat{\mathcal{T}}$. Formally, we define the following MLE-based objective function:

$$\mathcal{L}_{MLE} = -\frac{1}{|B|} \sum_{q \in B} \Big(\tilde{p}(X|\mathcal{G}^+) - \frac{1}{N_{neg}} \sum_{n=1}^{N_{neg}} \tilde{p}(X|\mathcal{G}_n^-) \Big), \quad (22)$$

where $|B|$ denotes the batch size, and $\mathcal{G}^+$ and $\mathcal{G}_i^-$ ($1 \leqslant n \leqslant N_{neg}$) are the positive and negative Markov networks for the query q, respectively. The partition function naturally vanishes in this expression after contrasting the joint likelihood defined over the positive network with the ones associated

to negative networks.

By optimizing with such objective function, two desired properties can be attained: (1) the embedding of the query sample is encouraged to approach the prototypes of its corresponding domain and category; (2) the prototypes from different domains but within the same category are aligned in the latent space, *i.e.* achieving domain invariance.

Overall objective. In the MRF-MSDA model, the prototypes $\mathbb{C} = \{\{c_k^m\}_{k=1}^K\}_{m=1}^{M+1}$ are optimized along with the feature extractor f to minimize the classification and MLE-based objective functions as below:

$$\min_{f,\mathbb{C}} \mathcal{L}_{cls} + \alpha\mathcal{L}_{MLE}, \tag{23}$$

where α is the trade-off weight for the MLE objective.

4.3 Model Inference

When model learning is finished, we save the feature extractor f and all prototypes $\mathbb{C}$. During inference, given a query sample q, its embedding z_q is extracted by the feature extractor, and the observation set $X = \{z_q, c_1^1, \cdots, c_K^{M+1}\}$ is formed by z_q and all prototypes. After that, following the label prediction scheme proposed in Sec. 4.2.1, we derive the classification probability for the query sample by summing the joint likelihoods of the observations over several specific Markov networks.

4.4 Comparison between CRF-MSDA and MRF-MSDA

In this section, we compare the proposed CRF-MSDA and MRF-MSDA model from two aspects, *i.e.* the model expressivity and the computational complexity, to shed the light on the effectiveness of these two types of graphical models.

4.4.1 Model Expressivity

Given a set of observations $X = \{z_q, c_1^1, \cdots, c_K^{M+1}\}$ composed of a query sample embedding and the prototypes on all domains, CRF-MSDA seeks to model the joint distribution of their corresponding labels $Y = \{y_q, y_1^1, \cdots, y_K^{M+1}\}$ conditioned on the observations, *i.e.* $p(Y|X)$. By comparison, MRF-MSDA aims to model the joint distribution of both observations and labels, *i.e.* $p(X, Y)$. Therefore, compared to CRF-MSDA, MRF-MSDA can not only capture the dependency among labels but also capture the dependency among different observations. As a matter of fact, the dependency among observations is useful to predict more accurate labels. For example, it can constrain the label predictions of correlated/uncorrelated observations to be similar/dissimilar. Such an advantage endows MRF-MSDA with stronger model expressivity.

4.4.2 Computational Complexity

We compare the computational complexity of two models on processing a single query sample step by step. For feature extraction and label prediction steps, the time complexity of learning and inference are identical for both models. However, for the graph construction step, the time complexity of two phases are different, and we thus discuss the complexity of learning and inference separately.

Feature extraction. Given a query sample, both models employ a feature extractor to obtain the query's embedding, which possesses identical computational cost.

Graph construction. The computational complexity of this step differs between learning and inference. In the learning phase, since the prototypes are online updated, the graph construction involves the computation of all prototypes and the query sample. The relational graph $\mathcal{G}$ constructed in the CRF-MSDA model requires to compute the pair-wise adjacency weight between $(M + 1)K$ prototypes and a query sample, and thus the time complexity equals to $\mathcal{O}(M^2K^2)$. For the MRF-MSDA model, a set of Markov networks $\mathbb{G} = \{\{\mathcal{G}_k^m\}_{k=1}^K\}_{m=1}^{M+1}$ are established. These networks have the same connections among prototypes (*i.e.* K cliques for K categories) and a different edge linking the query sample to $(M + 1)K$ distinct prototypes. In order to derive the joint likelihoods of the observations over these networks, $\frac{M(M+1)}{2}K$ pair-wise energies among prototypes and $(M + 1)K$ energies between the query and each prototype are computed, which owns a time complexity of $\mathcal{O}(M^2K)$. Therefore, MRF-MSDA is less computationally expensive than CRF-MSDA in this step during learning.

In the inference phase, the prototypes are fixed, and thus the adjacency weights (for CRF-MSDA) and the energies (for MRF-MSDA) among prototypes can be pre-computed. Therefore, given a query sample, the rest of computation is only between the query sample and prototypes, which has a time complexity of $\mathcal{O}(MK)$ for both models. In this way, CRF-MSDA and MRF-MSDA own an identical computational cost for graph construction during inference.

Label prediction. The CRF-MSDA model performs message passing on the constructed graph via a GNN model and predicts query's label by a linear classifier. By comparison, the label prediction of MRF-MSDA only requires the basic arithmetic calculations upon the joint likelihoods, which is model-free and more efficient.

In summary, for both learning and inference, the MRF-MSDA model is more computationally efficient than the CRF-MSDA counterpart in terms of processing a single query sample. In Sec. 6.3, we further conduct an empirical time complexity analysis to verify the points above.

5 EXPERIMENTS

In this section, we first describe the experimental settings and then compare the proposed models with existing methods on various benchmark data sets of MSDA to demonstrate their effectiveness.

5.1 Experimental Setup

Model details. For the CRF-MSDA model, we adopt a two-layer GCN [85] model to propagate messages among the observations, and, for each node of the relational graph, a linear classifier maps its d-dimensional feature representation to the K-dimensional classification probability. For the MRF-MSDA model, we consider two ways of deriving negative Markov networks based on a positive network: (1) The link between the query sample and its corresponding prototype is deleted, and we connect the query sample with any one of the rest $K-1$ prototypes within the same domain but belonging to different categories, which defines $K - 1$ negative networks; (2) We additionally select two random prototypes associated with distinct categories and connect

TABLE 1: The training setups on four different data sets.

data set	# domains	# classes	image size	backbone	batch size[*]	learning rate	# training epochs	feature dimension
Digits-five	5	10	32×32	3 conv-2 fc	128	2×10^{-4}	100	2048
Office-31 [81]	3	31	252×252	AlexNet	16	5×10^{-5}	100	4096
PACS [82]	4	7	224×224	ResNet-18	16	5×10^{-5}	100	512
DomainNet [12]	6	345	224×224	ResNet-101	16	5×10^{-5}	20	2048

[*] Batch size here denotes the number of examples sampled from one domain in each iteration.

TABLE 2: Classification accuracy (mean $\pm$ std %) of various methods on five MSDA tasks of *Digits-five*.

Standards	Methods	$\rightarrow$ **mm**	$\rightarrow$ **mt**	$\rightarrow$ **up**	$\rightarrow$ **sv**	$\rightarrow$ **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 [1]	63.8±0.7	96.3±0.5	94.2±0.9	62.5±0.7	85.4±0.8	80.4
	CORAL [3]	62.5±0.7	97.2±0.8	93.5±0.8	64.4±0.7	82.8±0.7	80.1
	DANN [83]	71.3±0.6	97.6±0.8	92.3±0.9	63.5±0.8	85.4±0.8	82.0
	ADDA [4]	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 [1]	67.9±0.8	97.5±0.6	93.5±0.8	67.8±0.6	86.9±0.5	82.7
	DANN [83]	70.8±0.8	97.9±0.7	93.5±0.8	68.5±0.5	87.4±0.9	83.6
	JAN [84]	65.9±0.7	97.2±0.7	95.4±0.8	75.3±0.7	86.6±0.6	84.1
	ADDA [4]	72.3±0.7	97.9±0.6	93.1±0.8	75.0±0.8	86.7±0.6	85.0
	MCD [6]	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 [10]	69.5±0.3	98.0±0.9	92.4±0.7	69.2±0.6	87.4±0.5	83.3
	DCTN [11]	70.5±1.2	96.2±0.8	92.8±0.3	77.6±0.4	86.8±0.8	84.8
	M³SDA [12]	72.8±1.1	98.4±0.7	96.1±0.8	81.3±0.9	89.6±0.6	87.7
	MDDA [13]	78.6±0.6	98.8±0.4	93.9±0.5	79.3±0.8	89.7±0.7	88.1
	CMSS [53]	75.3±0.6	99.0±0.1	97.7±0.1	**88.4**±0.5	93.7±0.2	90.8
	CRF-MSDA	85.6±0.8	99.0±0.4	98.3±0.4	83.2±0.6	93.0±0.5	91.8
	MRF-MSDA	**90.7**±0.7	**99.2**±0.2	**98.5**±0.4	85.8±0.7	**94.7**±0.5	**93.7**

them, which defines other N_2 negative networks. In total, for each query sample, we employ $N_{neg} = N_2 + K - 1$ negative networks to contrast with the positive one.

Training details. We list the basic training settings on four different data sets in Tab. 1. The setup differences on these data sets are mainly due to the distinction of data complexity, which follows the common experimental setups in the literature [10], [11], [12], [17]. In all experiments, we adopt an Adam [74] optimizer (weight decay: 5×10^{-4}) to train the model. For all the comparisons in this section, we use the following parameter settings for two proposed models: (1) For CRF-MSDA, the trade-off parameters λ_1 and λ_2 are set as 20 and 0.001 respectively, and the bandwidth parameter σ is set as 0.005; (2) For MRF-MSDA, the trade-off weight α is set as 1.0, the temperature parameter τ is set as 0.1, and the negative sampling size N_{neg} is set as $K + 5$ (*i.e.* $N_2 = 6$ negative networks per query are sampled by the second sampling way stated above). All these parameter setups are determined by the grid search on the source domains' validation sets of the $\rightarrow$ **mm** task (an MSDA task on Digits-five data set). For simplicity, we use "$\rightarrow D$" to denote the task of transferring from other domains to domain D. Our approach is implemented with PyTorch [86], and the source code will be released for reproducibility.

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 stan-

dards. (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 knowledge learned from multiple source domains are transferred to target domain. For the first two settings, previous single-source UDA methods, *e.g.* DAN [1], JAN [84], DANN [83], ADDA [4], MCD [6], are introduced for comparison. For the *Multi-Source* setting, we compare our approach with several existing MSDA algorithms, *e.g.* MDAN [10], DCTN [11], M³SDA [12], MDDA [13], and CMSS [53]. We report the performance of these methods on Digits-five and DomainNet from Peng *et al.* [12], on Office-31 from Zhao *et al.* [13] and on PACS from Yang *et al.* [53].

5.2 Experiments on Digits-five

Data set. The Digits-five data set is composed of five digital image domains, including MNIST (**mt**) [88], MNIST-M (**mm**) [83], SVHN (**sv**) [89], USPS (**up**) [90] and Synthetic Digits (**syn**) [83]. Each domain contains ten categories corresponding to the digits ranging from 0 to 9. Following the setting in DCTN [11], we sample 25000 images for training, 6000 images for validation and 9000 images for test on MNIST, MINST-M, SVHN and Synthetic Digits, and the entire USPS data set serves as a domain. The reported

TABLE 3: Classification accuracy (%) of various methods on three MSDA tasks of *Office-31*.

Standards	Methods	→ D	→ W	→ A	Avg
Single Best	Source-only	99.0	95.3	50.2	81.5
	RevGrad [2]	99.2	96.4	53.4	83.0
	DAN [1]	99.0	96.0	54.0	83.0
	RTN [87]	99.6	96.8	51.0	82.5
	ADDA [4]	99.4	95.3	54.6	83.1
Source Combine	Source-only	97.1	92.0	51.6	80.2
	DAN [1]	98.8	96.2	54.9	83.3
	JAN [84]	99.4	95.9	54.6	83.3
	DANN [83]	99.2	95.8	55.2	83.4
	ADDA [4]	99.2	96.0	55.9	83.7
	MCD [6]	99.5	96.2	54.4	83.4
Multi-Source	MDAN [10]	99.2	95.4	55.2	83.3
	DCTN [11]	99.6	96.9	54.9	83.8
	M³SDA [12]	99.4	96.2	55.4	83.7
	MDDA [13]	99.2	97.1	56.2	84.2
	CRF-MSDA	99.6	97.2	**56.9**	84.6
	MRF-MSDA	**99.7**	**97.4**	**56.9**	**84.7**

TABLE 4: Classification accuracy (mean ± std %) of various methods on four MSDA tasks of *PACS*.

Methods	→ A	→ C	→ P	→ S	Avg
Source-only	76.0±0.9	73.3±0.8	91.7±0.6	64.2±1.8	76.3
MDAN [10]	79.1±0.4	76.0±0.7	91.4±0.9	72.0±0.8	79.6
DCTN [11]	84.7±0.7	86.7±0.6	95.6±0.8	71.8±1.0	84.7
M³SDA [12]	89.3±0.4	89.9±1.0	97.3±0.3	76.7±2.9	88.3
MDDA [13]	86.7±0.6	86.2±0.7	93.9±0.7	77.6±0.9	86.1
Meta-MCD [54]	87.4±0.7	86.2±0.9	97.1±0.5	78.3±0.8	87.2
CMSS [53]	88.6±0.4	90.4±0.8	96.9±0.3	82.0±0.6	89.5
CRF-MSDA	90.2±0.5	90.5±0.6	97.2±0.5	81.5±0.7	89.9
MRF-MSDA	**92.2±0.4**	**93.3±0.6**	**98.0±0.3**	**86.7±0.8**	**92.6**

similar, which restricts the benefit brought by cross-domain joint modeling in our framework, especially in "→ A" task.

5.4 Experiments on PACS

Data set. The PACS [82] data set includes 4 domains, *i.e.* Photo (P), Art paintings (A), Cartoon (C) and Sketch (S). Each domain contains 7 categories, and significant domain shift (*i.e.* distinct painting styles) exists between different domains. Following two previous works [53], [54], only the approaches under the *Multi-Source* setting are employed for comparison. The mean and standard deviation of model performance over five independent runs are presented.

Results. In Tab. 4, we report the performance of various methods on four tasks. It can be observed that the proposed CRF-MSDA model performs comparably with the CMSS [53] approach. MRF-MSDA achieves the highest accuracy on all four tasks, and, especially, a 4.7% performance gain is obtained on the "→ S" task. The superior performance of MRF-MSDA can be mainly ascribed to its explorations of diverse intra- and inter-domain relations, which enables more precise label prediction when the distributional gap between different domains is large.

5.5 Experiments on DomainNet

Data set. DomainNet [12] is by far the largest and most difficult data set for MSDA. It consists of around 0.6 million images and 6 domains, *i.e.* clipart (clp), infograph (inf), painting (pnt), quickdraw (qdr), real (rel) and sketch (skt). Each domain includes the same 345 categories of common objects. The reported model performance is averaged over five independent runs using the same setting.

Results. The results of various approaches on Domain-Net are presented in Tab. 5. CRF-MSDA and MRF-MSDA perform comparably on this data set, and the latter achieves the best performance on five of six tasks. In particular, a 1.4% performance increase on average accuracy is gained by MRF-MSDA. The major challenge of this data set is the great complexity of data distribution, which is caused by two factors: (1) Large distributional gaps exist among different domains, *e.g.* from real images to sketches; (2) The numerous semantic categories within each domain lead to more complex single-domain data distribution. The CRF-MSDA model mitigates such dilemma by conducting category-level domain alignment and promoting feature compactness, while the MRF-MSDA model approaches such complex data

results are averaged over five independent runs under the same configuration.

Results. In Tab. 2, we compare the proposed CRF-MSDA and MRF-MSDA models with other works. Source-only stands for the model trained with only source domain data, which serves as the baseline. Compared to the state-of-the-art CMSS [53] approach, CRF-MSDA achieves notable performance gain on the "→ **mm**" task and surpasses it in terms of average accuracy over all tasks. The MRF-MSDA model performs best on four of five tasks and obtains a 12.1% performance increase relative to previous methods. These promising results illustrate the effectiveness of cross-domain joint modeling and learnable domain combination which are first explored in our approaches. MRF-MSDA outperforms CRF-MSDA on all five tasks, which mainly owes to its exploration of more diverse relational patterns over the observations by using positive and negative Markov networks.

5.3 Experiments on Office-31

Data set. Office-31 [81] is a classical domain adaptation benchmark with 31 categories and 4652 images. It contains three domains, *i.e.* Amazon (A), Webcam (W) and DSLR (D), and the data are collected from office environments. The data of Amazon are collected from amazon.com, while the data of Webcam and DSLR are captured by web camera and digital single-lens reflex camera under different conditions. There are 2,817, 795 and 498 images in A, W and D, respectively. Our methods are evaluated by five independent runs, and, following MDDA [13], we report the mean accuracy.

Results. Tab. 3 compares our methods with existing algorithms on three tasks. The MRF-MSDA model outperforms the state-of-the-art method, MDDA [13], with 0.5% in terms of average accuracy, and the CRF-MSDA model performs comparably with MRF-MSDA. On this data set, our approaches do not have obvious superiority, which probably ascribes to two reasons. (1) First, performance saturation occurs on "→ D" and "→ W" tasks, in which the Source-only model achieves performance higher than 95%. (2) Second, the Webcam and DSLR domains are highly

TABLE 5: Classification accuracy (mean $\pm$ std %) of various methods on six MSDA tasks of *DomainNet*.

Standards	Methods	$\to$ clp	$\to$ inf	$\to$ pnt	$\to$ qdr	$\to$ rel	$\to$ 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 [1]	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 [84]	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 [83]	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 [4]	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 [6]	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 [1]	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 [84]	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 [83]	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 [4]	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 [6]	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 [10]	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 [11]	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
	M³SDA [12]	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 [13]	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
	Meta-MCD [54]	62.8±0.2	21.4±0.1	50.5±0.1	15.5±0.2	64.6±0.2	50.4±0.1	44.2
	CMSS [53]	**64.2**±0.2	28.0±0.2	53.6±0.4	16.0±0.1	63.4±0.1	53.8±0.4	46.5
	CRF-MSDA	63.1±0.4	**28.7**±0.5	56.1±0.5	16.3±0.5	66.1±0.6	53.8±0.6	47.4
	MRF-MSDA	63.9±0.3	**28.7**±0.4	**56.3**±0.4	**16.8**±0.4	**67.1**±0.6	**54.3**±0.5	**47.9**

TABLE 6: The performance of CRF-MSDA under four model configurations on *Digits-five*.

$\mathcal{L}_{global}$	$\mathcal{L}_{local}$	$\to$ mm	$\to$ mt	$\to$ up	$\to$ sv	$\to$ 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 7: The performance of MRF-MSDA under three model configurations on *Digits-five*.

$\mathcal{L}_{cls}$	$\mathcal{L}_{MLE}$	$\to$ mm	$\to$ mt	$\to$ up	$\to$ sv	$\to$ syn	Avg
✓		86.54	99.07	98.36	82.90	92.80	91.93
	✓	82.30	99.05	98.21	83.66	92.34	91.11
✓	✓	**90.69**	**99.24**	**98.50**	**85.61**	**94.66**	**93.74**

distribution by modeling the joint distributions over various observations, which is a more direct scheme and performs better in practice.

6 ANALYSIS

In this section, we provide more in-depth analysis of the proposed methods to verify the effectiveness of major model components, in which both quantitative and qualitative studies are conducted for validation.

6.1 Ablation Study

6.1.1 Ablation Study for CRF-MSDA

In this part, we analyze the effect of the global and local alignment objective functions on the CRF-MSDA model. In Tab. 6, we evaluate model's performance under four configurations on the Digits-five data set. In the baseline setting (*1st* row), only the classification constraint (Eq. 10) is utilized to optimize the model. On the basis of the baseline setting, the global alignment constraint $\mathcal{L}_{global}$ (Eq. 13) can greatly enhance model's performance by performing category-level domain alignment (*2nd* row). For the local alignment constraint $\mathcal{L}_{local}$ (Eq. 14), after adding it to the baseline configuration, a 2.12% performance gain is achieved in terms of average accuracy (*3rd* row), which demonstrates the effectiveness of $\mathcal{L}_{local}$ on promoting the separability of feature

representations. In addition, when $\mathcal{L}_{global}$ and $\mathcal{L}_{local}$ are simultaneously applied, the highest classification accuracy is obtained (*4th* row), which shows the complementarity of global and local alignment constraints.

6.1.2 Ablation Study for MRF-MSDA

This set of experiments study the effect of classification and MLE-based objective functions on the MRF-MSDA model. Tab. 7 reports the performance of MRF-MSDA under three configurations on the Digits-five data set. When the classification constraint $\mathcal{L}_{cls}$ (Eq. 21) or MLE-based constraint $\mathcal{L}_{MLE}$ (Eq. 22) is individually applied (*1st/2nd* row), the classification accuracy is obviously lower than the full model configuration (*3rd* row), *i.e.* using both objective functions. These results illustrate the benefits of both joint distribution modeling and discriminative modeling on the observations. Through combining these two objectives, MRF-MSDA can derive more precise label prediction for the query sample.

6.2 Sensitivity Analysis

6.2.1 Sensitivity Analysis for CRF-MSDA

Sensitivity of bandwidth parameter σ. In this experiment, we discuss the selection of bandwidth parameter σ which controls the sparsity of the adjacency matrix **A** defined
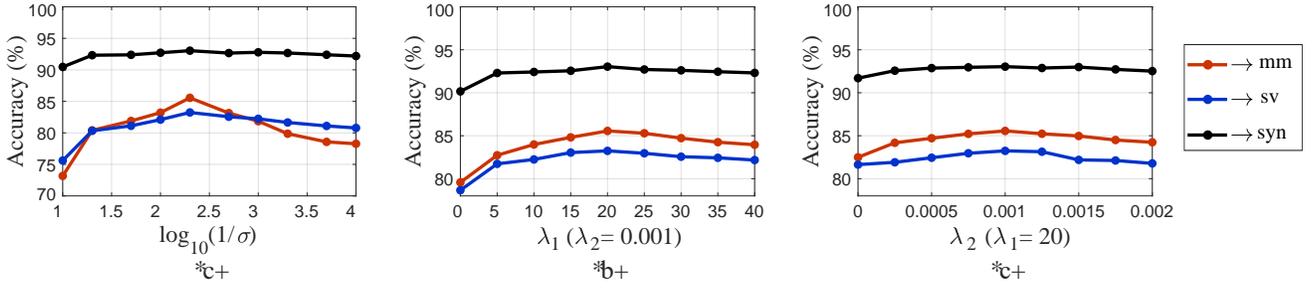
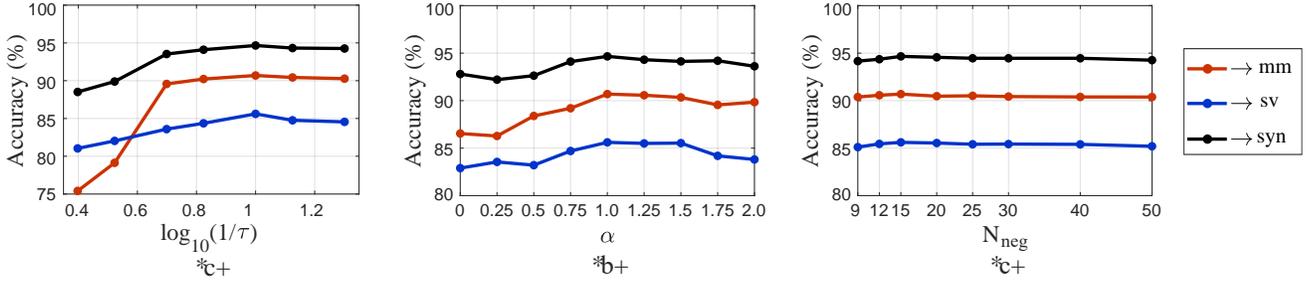Fig. 3: Sensitivity analysis for three parameters of CRF-MSDA on the *Digits-five* data set.

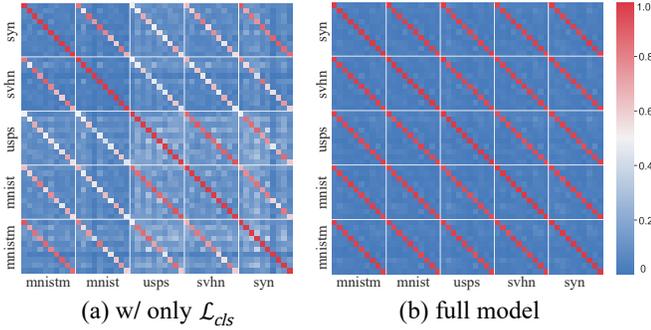Fig. 4: Sensitivity analysis for three parameters of MRF-MSDA on the *Digits-five* data set.

Fig. 5: The adjacency matrix in CRF-MSDA. (Results are evaluated on the "→ **mm**" task of *Digits-five*.)

TABLE 8: Running time over 100 iterations of different methods on *Digits-five* in both training and inference phase.

Methods	training (s)	inference (s)
M^3SDA [12]	19.92 ± 0.07	$\mathbf{4.96} \pm 0.08$
CRF-MSDA	17.41 ± 0.09	5.62 ± 0.12
MRF-MSDA	$\mathbf{16.54} \pm 0.10$	5.40 ± 0.08

in Eq. 4. In Fig. 3(a), we plot the performance of models trained with different σ values. We can observe that, on all three tasks, the highest accuracy is achieved when the value of σ is around 0.005. Under such condition, the adjacency matrix can capture the relations among observations most appropriately. Also, it is worth noticing that 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 part, we evaluate the CRF-MSDA model's sensitivity to λ_1 and λ_2 which balance between different learning objectives. Fig. 3(b) and Fig. 3(c) show model's performance under various λ_1 (λ_2) values when the other trade-off parameter λ_2 (λ_1) is fixed. It can be observed that CRF-MSDA model's performance is not sensitive to λ_1 and λ_2 when they are around 20 and 0.001, respectively. When these two parameters approach 0, obvious performance decrease occurs, which again verifies that both global and local alignment constraints are indispensable.

6.2.2 Sensitivity Analysis for MRF-MSDA

Sensitivity of temperature parameter τ. This experiments studies the selection of temperature parameter τ which

scales the energy function defined in Eq. 17. According to Fig. 4(a), when τ is around 0.1, the corresponding scaling can benefit the MRF-MSDA model to the greatest extent. With the increase of the temperature parameter, the model's performance drops apparently, *e.g.* a nearly 15% decrease on the "→ **mm**" task when $\tau = 0.4$. This phenomenon illustrates that the joint distribution modeling of MRF-MSDA relies on a proper scale of energies to define the joint likelihoods.

Sensitivity of trade-off parameter α. In this part, we analyze the sensitivity of the trade-off parameter α which balances between the objectives for classification and maximum likelihood estimation. From the line chart in Fig. 4(b), we can observe that the MRF-MSDA model performs stably better when the value of α is around 1.0 compared to other settings. Such value of α is able to attain an appropriate balance between two learning objectives.

Sensitivity of negative sampling size N_{neg}. The optimization of MRF-MSDA depends on sampling negative Markov networks to contrast with, which derives a parameter of negative sampling size N_{neg}. Based on the results shown in Fig. 4(c), we can conclude that the performance of MRF-MSDA is not sensitive to the value of N_{neg}, which, we think, owes to the strong negative samples (*i.e.* negative networks with only minor difference relative to the positive one) used in our method.
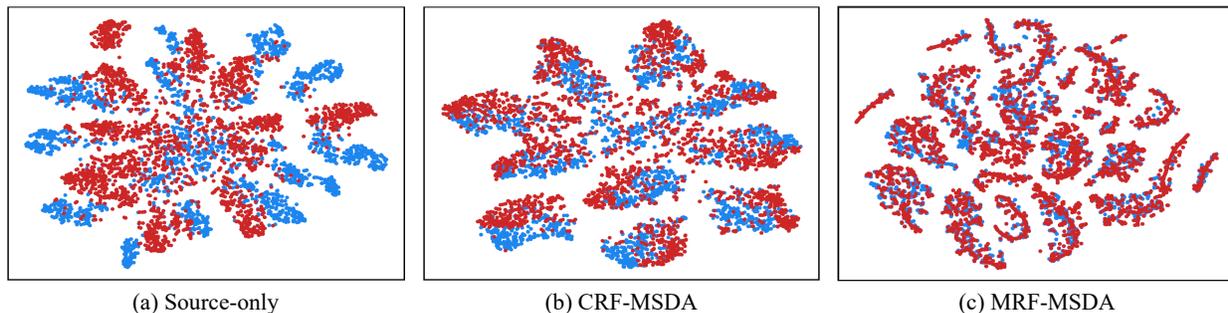
(a) Source-only   (b) CRF-MSDA   (c) MRF-MSDA

Fig. 6: Visualization of feature embeddings. (All results are evaluated on the "→ **mm**" task.)

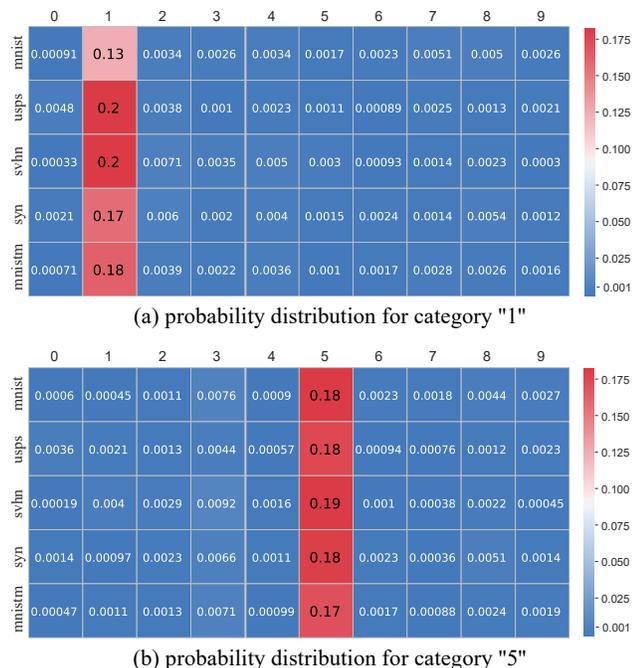(a) probability distribution for category "1"

(b) probability distribution for category "5"

Fig. 7: The probability distribution in MRF-MSDA. (Results are evaluated on the "→ **mm**" task of *Digits-five*.)

6.3 Times Complexity Analysis

Table 8 reports the running time of 100 iterations in both training and inference phase of different methods on Digits-five data set. The hardware conditions for the experiments are Intel(R) Xeon(R) CPU E5-2620 v4@2.40 GHz with 8 processors and one NVIDIA TITAN Xp GPU. All the reported results are averaged over 10 independent runs under the same configuration. From the table we can observe that in the training phase, under 100 iterations, the MRF model is about 0.9 seconds faster than the CRF model. And in the inference phase, MRF-MSDA model also runs slightly faster than CRF-MSDA model. These experimental results verify that MRF-MSDA is indeed more computationally efficient than CRF-MSDA for both learning and inference.

6.4 Visualization

6.4.1 Visualization for CRF-MSDA

In the CRF-MSDA model, the adjacency matrix $\mathbf{A}$ (Eq. 4) quantifies the category-level relevance between various domains. In Fig. 5, we visualize $\mathbf{A}$ under two model configurations, in which each pixel denotes the adjacency between

two categories from arbitrary domains. Compared to the configuration with only classification constraint, the full model applying both classification and alignment objective functions achieves better cross-domain consistency on the relevance among various categories, which illustrates the effectiveness of global-level alignment.

6.4.2 Visualization for MRF-MSDA

In this part, we visualize the category-specific probability distribution derived by MRF-MSDA. For the query sample q, we combine the probabilities $p(y_d = m, y = k|q)$ $(1 \leqslant m \leqslant M + 1, 1 \leqslant k \leqslant K)$ defined in Eq. 18 as a probability matrix $\mathbf{P}_q \in \mathbb{R}^{(M+1) \times K}$. Through averaging $\mathbf{P}_q$ over all the query samples within a specific category, we can obtain the probability matrix for that category. In Fig. 7, we visualize the probability matrix for category "1" and "5" on the target domain's test set of "→ **mm**" task. We can observe that high probability values evenly distribute on the corresponding categories ("1" or "5") of various domains, which demonstrates that MRF-MSDA effectively aligns the samples within the same category and separates the ones from distinct categories in the latent space.

6.4.3 Visualization of feature embeddings

In Figure 6, we utilize t-SNE [91] to visualize the feature distributions of one of source domains (SVHN) and target domain (MNIST-M). Compared with the Source-only baseline, the proposed CRF-MSDA and MRF-MSDA model make the features of target domain more discriminative and better aligned with those of source domain. Compared to CRF-MSDA, the feature representations derived by MRF-MSDA are better aligned across two domains, which is in line with the better empirical performance of MRF-MSDA on the "→ **mm**" task.

7 Conclusions and Future Work

In this work, we aim to address the Multi-Source Domain Adaptation (MSDA) problem. Specifically, we propose two graphical models, *i.e.* Conditional Random Field for MSDA (CRF-MSDA) and Markov Random Field for MSDA (MRF-MSDA), to realize cross-domain joint modeling and learnable domain combination. Extensive experiments on various benchmark data sets of MSDA illustrate the superior performance of our methods over existing works. In the future work, we will explore other graphical models for MSDA, *e.g.* Bayesian networks and chain graphs.

REFERENCES

[1] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *International Conference on Machine Learning*, 2015.

[2] Y. Ganin and V. S. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International Conference on Machine Learning*, 2015.

[3] B. Sun and K. Saenko, "Deep CORAL: correlation alignment for deep domain adaptation," in *ECCV Workshop*, 2016.

[4] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[5] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Advances in Neural Information Processing Systems*, 2018.

[6] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[7] R. Xu, G. Li, J. Yang, and L. Lin, "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation," in *International Conference on Computer Vision*, 2019.

[8] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation with multiple sources," in *Advances in Neural Information Processing Systems*, 2008.

[9] J. Hoffman, M. Mohri, and N. Zhang, "Algorithms and theory for multiple-source adaptation," in *Advances in Neural Information Processing Systems*, 2018.

[10] H. Zhao, S. Zhang, G. Wu, J. M. F. Moura, J. P. Costeira, and G. J. Gordon, "Adversarial multiple source domain adaptation," in *Advances in Neural Information Processing Systems*, 2018.

[11] R. Xu, Z. Chen, W. Zuo, J. Yan, and L. Lin, "Deep cocktail network: Multi-source unsupervised domain adaptation with category shift," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[12] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *IEEE International Conference on Computer Vision*, 2019.

[13] S. Zhao, G. Wang, S. Zhang, Y. Gu, Y. Li, Z.-C. Song, P. Xu, R. Hu, H. Chai, and K. Keutzer, "Multi-source distilling domain adaptation," in *AAAI Conference on Artificial Intelligence*, 2020.

[14] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems*, 2017.

[15] S. Xie, Z. Zheng, L. Chen, and C. Chen, "Learning semantic representations for unsupervised domain adaptation," in *International Conference on Machine Learning*, 2018.

[16] M. Xu, H. Wang, B. Ni, H. Guo, and J. Tang, "Self-supervised graph-level representation learning with local and global structure," in *International Conference on Machine Learning*, 2021.

[17] H. Wang, M. Xu, B. Ni, and W. Zhang, "Learning to combine: Knowledge aggregation for multi-source domain adaptation," in *European Conference on Computer Vision*, 2020.

[18] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, pp. 723–773, 2012.

[19] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *CoRR*, vol. abs/1412.3474, 2014.

[20] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo, "Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[21] M. Long, Y. Cao, Z. Cao, J. Wang, and M. I. Jordan, "Transferable representation learning with deep adaptation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 3071–3085, 2018.

[22] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *AAAI Conference on Artificial Intelligence*, 2018.

[23] C. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced wasserstein discrepancy for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[24] Y. Balaji, R. Chellappa, and S. Feizi, "Normalized wasserstein for mixture distributions with applications in adversarial learning and domain adaptation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6500–6508.

[25] M. Chen, S. Zhao, H. Liu, and D. Cai, "Adversarial-learned loss for domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[26] S. Cui, S. Wang, J. Zhuo, C. Su, Q. Huang, and Q. Tian, "Gradually vanishing bridge for adversarial domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 455–12 464.

[27] H. Tang and K. Jia, "Discriminative adversarial domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[28] M. Xu, H. Wang, B. Ni, R. Zhu, Z. Sun, and C. Wang, "Cross-category video highlight detection via set-based learning," in *IEEE International Conference on Computer Vision*, October 2021, pp. 7970–7979.

[29] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[30] M. Liu and O. Tuzel, "Coupled generative adversarial networks," in *Advances in Neural Information Processing Systems*, 2016.

[31] A. Dundar, M. Liu, T. Wang, J. Zedlewski, and J. Kautz, "Domain stylization: A strong, simple baseline for synthetic to real image domain adaptation," *CoRR*, vol. abs/1807.09384, 2018.

[32] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li, "Deep reconstruction-classification networks for unsupervised domain adaptation," in *European Conference on Computer Vision*, 2016.

[33] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa, "Generate to adapt: Aligning domains using generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[34] M. Xu, J. Zhang, B. Ni, T. Li, C. Wang, Q. Tian, and W. Zhang, "Adversarial domain adaptation with domain mixup," in *AAAI Conference on Artificial Intelligence*, 2020.

[35] G. Yang, H. Xia, M. Ding, and Z. Ding, "Bi-directional generation for unsupervised domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[36] J. Hoffman, E. Tzeng, T. Park, J. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," in *International Conference on Machine Learning*, 2018.

[37] L. Tran, K. Sohn, X. Yu, X. Liu, and M. Chandraker, "Gotta adapt 'em all: Joint pixel and feature-level domain adaptation for recognition in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[38] R. Takahashi, A. Hashimoto, M. Sonogashira, and M. Iiyama, "Partially-shared variational auto-encoders for unsupervised domain adaptation with target shift," in *European Conference on Computer Vision*, 2020.

[39] H. Wang, T. Shen, W. Zhang, L.-Y. Duan, and T. Mei, "Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation," in *European Conference on Computer Vision*. Springer, 2020, pp. 642–659.

[40] Y. Pan, T. Yao, Y. Li, Y. Wang, C. Ngo, and T. Mei, "Transferrable prototypical networks for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[41] M. Xu, H. Wang, B. Ni, Q. Tian, and W. Zhang, "Cross-domain detection via graph-induced prototype alignment," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[42] W. Zhang, W. Ouyang, W. Li, and D. Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[43] C. Chen, W. Xie, W. Huang, Y. Rong, X. Ding, Y. Huang, T. Xu, and J. Huang, "Progressive feature alignment for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[44] K. Saito, D. Kim, S. Sclaroff, T. Darrell, and K. Saenko, "Semi-supervised domain adaptation via minimax entropy," in *IEEE International Conference on Computer Vision*, 2019.

[45] X. Jiang, Q. Lao, S. Matwin, and M. Havaei, "Implicit class-conditioned domain alignment for unsupervised domain adap-

tation," in *International Conference on Machine Learning*. PMLR, 2020, pp. 4816–4827.

[46] X. Wang, Y. Jin, M. Long, J. Wang, and M. I. Jordan, "Transferable normalization: Towards improving transferability of deep neural networks," in *Advances in Neural Information Processing Systems*, 2019.

[47] S. Cui, S. Wang, J. Zhuo, L. Li, Q. Huang, and Q. Tian, "Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3941–3950.

[48] Y. Jin, X. Wang, M. Long, and J. Wang, "Minimum class confusion for versatile domain adaptation," in *European Conference on Computer Vision*, 2020.

[49] X. Ma, T. Zhang, and C. Xu, "GCAN: graph convolutional adversarial network for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8266–8276.

[50] Y. Luo, Z. Wang, Z. Huang, and M. Baktashmotlagh, "Progressive graph learning for open-set domain adaptation," in *International Conference on Machine Learning*, 2020, pp. 6468–6478.

[51] X. Yang, C. Deng, T. Liu, and D. Tao, "Heterogeneous graph attention network for unsupervised multiple-target domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, 2020.

[52] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning bounds for domain adaptation," in *Advances in Neural Information Processing Systems*, 2007.

[53] L. Yang, Y. Balaji, S. Lim, and A. Shrivastava, "Curriculum manager for source selection in multi-source domain adaptation," in *European Conference on Computer Vision*, 2020.

[54] D. Li and T. M. Hospedales, "Online meta-learning for multi-source and semi-supervised domain adaptation," in *European Conference on Computer Vision*, 2020.

[55] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[56] R. Vemulapalli, O. Tuzel, M.-Y. Liu, and R. Chellapa, "Gaussian conditional random field network for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[57] Z.-H. Yuan, T. Lu, Y. Wu *et al.*, "Deep-dense conditional random fields for object co-segmentation," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2017.

[58] Q.-H. Pham, T. Nguyen, B.-S. Hua, G. Roig, and S.-K. Yeung, "Jsis3d: Joint semantic-instance segmentation of 3d point clouds with multi-task pointwise networks and multi-value conditional random fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[59] S. Roth and M. J. Black, "Fields of experts: A framework for learning image priors," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[60] R. Vemulapalli, O. Tuzel, and M.-Y. Liu, "Deep gaussian conditional random field network: A model-based deep network for discriminative denoising," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[61] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[62] Y. Xue, J. Chen, W. Wan, Y. Huang, C. Yu, T. Li, and J. Bao, "Mvscrf: Learning multi-view stereo with conditional random fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[63] M. F. Tappen, B. C. Russell, and W. T. Freeman, "Efficient graphical models for processing images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[64] X. Wang, K. Yu, C. Dong, and C. C. Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[65] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 721–741, 1984.

[66] J. Sun and M. F. Tappen, "Learning non-local range markov random field for image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[67] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3929–3938.

[68] K. Held, E. R. Kops, B. J. Krause, W. M. Wells, R. Kikinis, and H.-W. Muller-Gartner, "Markov random field segmentation of brain mr images," *IEEE transactions on medical imaging*, vol. 16, no. 6, pp. 878–886, 1997.

[69] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang, "Deep learning markov random field for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 8, pp. 1814–1828, 2017.

[70] L. Bao, B. Wu, and W. Liu, "Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5977–5986.

[71] G. R. Cross and A. K. Jain, "Markov random field texture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 1, pp. 25–39, 1983.

[72] F. Heitz and P. Bouthemy, "Multimodal estimation of discontinuous optical flow using markov random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 12, pp. 1217–1232, 1993.

[73] V. Lempitsky, C. Rother, S. Roth, and A. Blake, "Fusion moves for markov random field optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1392–1405, 2009.

[74] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

[75] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[76] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *International Conference on Artificial Intelligence and Statistics*, 2010.

[77] ——, "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics," *Journal of Machine Learning Research*, vol. 13, pp. 307–361, 2012.

[78] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: online learning of social representations," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014.

[79] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: large-scale information network embedding," in *International Conference on World Wide Web*, 2015.

[80] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[81] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *European Conference on Computer Vision*, 2010.

[82] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Deeper, broader and artier domain generalization," in *IEEE International Conference on Computer Vision*, 2017, pp. 5542–5550.

[83] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[84] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *International Conference on Machine Learning*, 2017.

[85] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations*, 2017.

[86] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NeurIPS Workshop*, 2017.

[87] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Advances in Neural Information Processing Systems*, 2016.

[88] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[89] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *NIPS Workshops*, 2011.

[90] J. J. Hull, "A database for handwritten text recognition research," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

[91] L. V. D. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 2605, pp. 2579–2605, 2008.