Information Maximizing Adaptation Network With Label Distribution Priors for Unsupervised Domain Adaptation

Pei Wang, Yun Yang, Yuelong Xia, Kun Wang, Xingyi Zhang, *Senior Member, IEEE*, and Song Wang, *Senior Member, IEEE*

***Abstract*—Unsupervised domain adaptation, which transfers knowledge from the source domain to the target domain, has still been a challenging problem. However, previous domain adaptation methods typically minimize the domain discrepancy by using the pseudo target labels. Since the pseudo labels can be noisy, which may cause misalignment and unsatisfying adaptation performance. To address the above challenges, we propose an information maximization adaptation network with label distribution priors. We revisit feature alignment in unsupervised domain adaptation from the perspective of distribution alignment, and find that learning discriminant feature representation requires to minimizing distribution discrepancy and maximizing source mutual information between the outputs of the classifier and feature representations. Due to domain shift, maximizing target mutual information may align features to incorrect class directly. We propose a weighted target mutual information by re-weighting the estimated mutual information via the mean prediction confidence in mini-batch, which can eliminate the negative impact of inaccurate estimation. In addition, we introduce a regularization term of label priors distribution to encourage the similarity to the real label distribution. Extensive experimental results on three benchmark datasets show that our proposed method can achieve remarkable results compared with previous methods.**

***Index Terms*—Information theory, label distribution priors, mutual information, unsupervised domain adaptation.**

Manuscript received 16 February 2022; revised 1 July 2022 and 21 August 2022; accepted 25 August 2022. Date of publication 1 September 2022; date of current version 1 November 2023. This work was supported in part by the Chinese Natural Science Foundation under Grants 61876166 and 61663046, and in part by Yunnan provincial major science and technology special plan Projects: Digitization Research and Application Demonstration of Yunnan characteristic industry, under Grant 202002AD080001. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Xinxiao Wu. *(Corresponding author: Yun Yang.)*

Pei Wang, Yuelong Xia, and Kun Wang are with the School of Information Science & Engineering, Yunnan University, Kunming 650504, China (e-mail: peiwang@mail.ynu.edu.cn; xyl@mail.ynu.edu.cn; kunwang@mail.ynu.edu.cn).

Yun Yang is with the National Pilot School of Software, Yunnan University, Kunming 650091, China (e-mail: yangyan19@hotmail.com).

Xingyi Zhang is with the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, Anhui University, Hefei 230039, China (e-mail: xyzhanghust@gmail.com).

Song Wang is with the College of Engineering and Computing, University of South Carolina, Columbia, SC 29208 USA (e-mail: songwang@cec.sc.edu).

Digital Object Identifier 10.1109/TMM.2022.3203574

I. Introduction

DEEP learning has achieved great success in diverse applications, particularly computer vision [1], [2], natural language processing [3] and medicine [4], [5], [6]. Despite the success has been made, deep learning depends on a large amount of labeled data, where the labeling process is both time-consuming and expensive. A natural solution is to learn a model through many labeled source domain (train) data and directly generalize it to the target (test) domain. However, due to the distribution discrepancy across domains, the performance of the learned model decreases sharply in the target domain. This phenomenon is known as distribution shift [7], [8], which has become an important research issue in machine learning.

Domain adaptation aims to reduce domain distribution discrepancy by learning domain invariant representations [7]. This paper focus on unsupervised domain adaptation (UDA), since it appears in many real-world applications. In UDA setting, the training data composes of labeled data from the source domain and unlabeled data from the target domain [8]. UDA methods can be divided into two categories: statistic moment matching-based approaches [9], [10], [11], [12] and adversarial-based methods [13], [14], [15], [16], [17], [18], [19], [20], [21]. The former reduces the discrepancy by aligning the higher-order statistical moments of the domain (e.g. local maximum mean discrepancy); The latter learns the domain invariant feature representations by fooling the domain discriminator. Some recent studies have shown that over aligning the distribution between domains can damage the transfer performance [22], [23]. The main reason is as follows. The above methods leverage the knowledge from a labeled source domain to the target domain by aligning the domain-level features without considering the category information. Due to the domain discrepancy, the large target data density near the decision boundary (i.e. high uncertainty), which may lead to misalignment and unsatisfying adaptation performance.

With regard to the problems of domain-level alignment, more and more researchers pay attention to the class-aware methods of UDA (also called conditional distribution alignment or sub-domain adaptation) [15], [24], [25]. These methods align the distribution of the same category across domains to eliminate the intra-class distribution discrepancy and improve the

generalization ability and discriminability of the learned model. Unfortunately, calculating the intra-class distribution discrepancy requires labels from both domains. However, the target domain labels are not available in UDA. A straightforward solution is to use the classifier output as the target label. For example, Weighed maximum mean discrepancy (WMMD) [26] and class-specific MMD (CMMD) [27] measure the discrepancy by assigning the pseudo label of the target domain. Still, the estimated label might be wrong, thus deep subdomain adaptation network (DSAN) [24] weighting the target sample by using the probability prediction to eliminate the side-effect of the wrong labeling. In addition, some methods update pseudo labels through clustering [28], [29]. Experiments show that the fuzzy samples (samples far from the cluster centers) will gradually disappear during training. For example, Transferrable Prototypical Networks (TPN) [28] investigate domain adaptation with the general-purposes and specific-tasks based on pseudo class-prototypes. Contrastive Adaptation network (CAN) [29] introduces a contrastive domain discrepancy, which minimizing intra-class discrepancy and maximizing inter-class discrepancy simultaneously. Although class-aware based methods have achieved remarkable success, the generated pseudo labels may be inaccurate and noisy for the target domain, which will lead to the degradation of generalization performance.

To deal with such challenges, existing methods focuses on how to explore the unlabeled data of the target domain, such as entropy minimization (EntMin) [30], batch nuclear-norm maximization (BNM) [31], Minimum Class Confusion (MCC) [32]. However, these methods may affect the generalization ability of the learned model across domains without making full use of the information of the source domain. To further analyze how the source domain data affects the generalization performance of the model, we reinvestigate feature alignment from the perspective of distribution alignment. We theoretically prove that the two key issues to realize unsupervised adaptation are aligning feature distribution and maximizing the mutual information [33] between representation and the output of the learned model. The former transfers knowledge by reducing the distribution discrepancy across domains, while the latter encourages discrimination by minimizing the cross entropy and encourages prediction diversity by maximizing the label entropy.

In addition, some existing methods have proved to improve the quality of pseudo labels by maximizing the target mutual information [34], [35]. Although these methods have made some advanced, it may align the feature representations to the wrong class. To deal with this issue, we propose a weighted target domain mutual information based on the prediction confidence score. Considering the results predicted by the classifier may be inaccurate and noisy, we propose to use the classification accuracy of the current batch as the confidence score of mutual information on the target domain, which can effectively eliminate the influence of noisy pseudo labels and further improve the discriminability of feature representation.

Although information maximization can learn the task-relevant feature representations across domains, a mini-batch may contain only samples from a subset of categories during training. Thus, the diversity term in information maximization might be misclassify the samples as classes that do not exist in the mini-batch [31], which may damage the adaptation performance, this phenomenon called as diversity collapse. To address this problem, we introduce a label distribution priors regularization that encourages the estimated label distribution to be close to the real label distribution, which can rectify the prediction output diversity on mini-batch source domain. A label distribution priors can improve the transfer learning performance.

The main contributions of this paper are as follows:

1) From the perspective of feature alignment, we theoretically validate that it not only needs to align feature distribution but also maximize the mutual information between feature representation and outputs of the classifier on source domain. To our best knowledge, it is the first time to improve adaptation performance in UDA by maximizing the mutual information on source domain.

2) Considering mutual information maximization of the target domain directly may misalign the features, we propose a weighted target mutual information based on the prediction confidence score. It can effectively eliminate the influence of data distribution discrepancy and improve the discriminability of the learned representation.

3) We take consideration a label distribution priors information into our final objective, which can prevent the predicted empirical distribution far away from the real label distribution, and avoid the problem of diversity collapse.

4) We propose an information maximization adaptation network with label distribution priors, which can eliminate the influence of noisy pseudo labels in and end-to-end training manner without adding any additional modules and parameters. Extensive experiments on three benchmark datasets validate that the effective of the proposed method.

II. Related Work

UDA mainly learns domain invariant representations by reducing the distribution discrepancy across domains. Compared with domain adaptation, UDA considers more practical applications, in which the target domain label data is not available. There are two kinds of distribution discrepancy measures: statistical moment matching-based approachs [9], [10] and adversarial-based approachs [14], [15], [16], [17], [18]. The former aligns the distribution of data by matching multi-order statistical moments, while the latter confuses a domain discriminator to extract domain invariant representations.

The statistic moment matching-based methods are further divided into maximum mean discrepancy (MMD) [10], [26], central moment discrepancy (CMD) [11] and second-order statistical matching [36]. MMD is widely used in adaptation domain. In addition, some extensions of MMD, such as conditional MMD [37] and joint MMD [9] are used to measure the distribution discrepancy of different domains in Hilbert-Schmidt norm space. In order to solve the impact of label shift, weighted MMD [26] and generalized label shift (GLS) [23] are proposed. These methods further improve the performance of the model by estimating the label weight ratio and the re-weighting of samples. The adversarial-based approaches aim to train a domain

discriminator to distinguish whether the input comes from the source domain or the target domain. The maximum classifier discrepancy (MCD) [38]learns the domain invariant representations by minimizing the prediction disagreement of the two different classifiers. These methods align domain-level features without considering category information. Recently, some studies have shown that by aligning the relevant subdomain distributions of domain-specific layer activations across different domains based on a local maximum mean discrepancy (LMMD) [15], [24], [25] has achieved better generalization performance. For instance, some methods, such as CMMD [27] and LMMD [24], align the domain distribution by capturing the fine-grained information with regard to each class. Besides, CAN [29] proposes a contrastive distribution discrepancy that models the intra-class discrepancy and the inter-class discrepancy explicitly. But, CAN relies on clustering and category aware sampling strategies.

In UDA, some methods further explore unlabeled data to improve the generalization ability of the learned model [9], [31], [32], [39], [40]. For example, EntMin [30] is used to obtain the deterministic prediction of target domain samples [9]. Furthermore, Chen et al. [39] propose the maximize square loss of prediction output to reduce the influence of easy-to-transfer samples in EntMin. MCC [32] tackle a variety of domain adaptation scenarios by minimizing the confusion loss of target classification prediction without any modification. Transferrable Prototypical Networks (TPN) [28] is domain adaptation method based on pseudo class prototype. Self-Ensembling (SE) [40] relies on ensemble learning and data augmentation. Some recent works explore the transferability, discriminability, and diversity of feature matrix from the perspective of matrix analysis, such as BNM [31], AFN [41] and batch spectral penalization (BSP) [22]. Specifically, AFN [41] enhances the feature transferability by increasing the feature norm, while BSP [22] balances transferability and discriminability by penalizes the largest eigenvalues of the feature matrix. Compared with AFN and BSP, BNM [31] simultaneously enhances the discriminability and diversity of the prediction by using the batch nuclear norm of feature matrix to avoid falling into trivial solutions, i.e. a large number of samples are predicted into a few classes. Dynamic weighted learning (DWL) [21] dynamically adjusts the degree of transferability and discriminability on the target domain to avoid the problem of discriminability vanishing and excessive alignment. To eliminates the influence of irrelevant semantic features, Semantic Concentration for Domain Adaptation (SCDA) [19] encourage to find the area with the most principal features by minimax the prediction distribution of the same class of samples in adversarial manner. Selective Entropy Optimization vis Committee Consistency (SENTRY) [42] selectively optimizes the entropy of target samples through the consistency of multiple random image transformations. However, these methods ignore a priors information contained in the batch source domain data.

Some self-training methods train model by generating high-quality pseudo labels [43], [44]. With the increase of the number of categories, the quality of pseudo labels is seriously decreased. Such self-training strategy suffers from error accumulation, which will reduce the generalization performance. Different from the above methods, we learn the task-relevant representation via information maximization, which avoids the problem of error accumulation and can flexibly handle different task scenarios. Additional, although we also used pseudo labels in LMMD loss, it is worth to note that we weighting the target samples by prediction probability. This weighting strategy can increase the tolerance of noisy labels, since the samples with high confidence will make a greater contribution to distribution alignment.

This paper is related to the work that explicitly model the diversity and the mutual information, such as source hypothesis transfer (SHOT) [45], data free multi-source unsupervised domain adaptation (DECISION) [46], Domain Preservations Nets(DPN) [34], Contrastive Learning and Mutual Information Maximization [47], and minimal-entropy diversity maximization (MEDM) [48]. Different form these methods, our work has the following differences. Firstly, we maximize the mutual information on source domain data, which can better improve the discriminability of the learned feature representation. Secondly, we re-weighting the mutual information on the target domain by the mean confidence threshold to eliminate the influence of distribution shift.

III. Method

In this section, we give our main contributions, i.e., label distribution priors and information maximization adaptation networks. In Section 3.1, we review the overview of unsupervised domain adaptation network. In Section 3.2, we analyze the information maximization from the perspective of information theory, and give the corresponding alternative: maximization the mutual information between the feature representations and the classifier outputs. Besides, we describe our proposed label distribution priors in Section 3.3. Then, in Section 3.4, we introduce the information maximization adaptation network with label distribution priors. Finally, we theoretically analyze the effectiveness of the proposed method in UDA.

A. Overview of UDA

In UDA scenario, we are given a training dataset with n_s labeled examples sampling from the source domain $\mathcal{D}_s = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^{n_s}$, and a test dataset with n_t unlabeled samples following the target domain $\mathcal{D}_t = \{(\mathbf{x}_i^t)\}_{i=1}^{n_t}$. Denote by $P_s(\mathbf{x}, \mathbf{y})$ and $P_t(\mathbf{x}, \mathbf{y})$ the data distribution of the source domain $\mathcal{D}_s$ and the target domain $\mathcal{D}_t$, respectively. In UDA, we focus on the covariate shift, where the marginal distribution across domains are different, i.e., $P_s(\mathbf{x}) \neq P_t(\mathbf{x})$, whereas the conditional distribution are the same, i.e., $P_s(\mathbf{y}|\mathbf{x}) = P_t(\mathbf{y}|\mathbf{x})$. The goal of this paper is to learn deep neural network $\mathbf{y} = f(\mathbf{x})$ by eliminating the discrepancy of joint distribution, such that the model learned from the source domain can generalize well to the target domain [7].

Formally, we aim to find the function $f(\mathbf{x})$ to minimize the expected error on the target domain. Therefore,

the expected loss of the objective function can be defined as: $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim P_t(\mathbf{x},\mathbf{y})}[\ell(f(\mathbf{x}),\mathbf{y})] = \int P_t(\mathbf{x},\mathbf{y}) \cdot \ell(f(\mathbf{x}),\mathbf{y})\mathrm{d}x\mathrm{d}y$, where $\ell(f(\mathbf{x}),\mathbf{y})$ is the loss function, i.e., the cross entropy loss function for the classification task. In UDA, the function learned $f(\mathbf{x})$ from the source domain can be directly applied to the target domain due to the conditional distribution stay same. The objective function of UDA can be define as:

$$\min_f \frac{1}{n_s}\sum_{i=1}^{n_s} \ell(f(\mathbf{x}_i^s),\mathbf{y}_i^s) + \beta M_k(P_s,P_t) \tag{1}$$

where the first term $\ell(f(\mathbf{x}_i^s),\mathbf{y}_i^s)$ is cross-entropy loss of source samples, and the second term $M_k(P_s,P_t)$ denote as the distribution discrepancy across domains. λ is a parameter for trade off the importance of the cross-entropy loss and the distribution discrepancy. In training phase, we use the mini-batch stochastic descent algorithm to fine-tune the pre-trained model on the ImageNet. Note that in the iterative process, the pseudo labels of target samples often become more accurate.

To align the data distribution, most existing methods measure the distribution discrepancy by using the maximum mean discrepancy, which is a nonparametric measure. Since it does not need to estimate the probability density, it has been widely used in transfer learning. The core idea is that if all statistical moments are consistent, the distribution is also the same. In other words, the MMD [10] is a domain adaptation loss by calculating the higher-order statistical moments of the data. It is defined as follows:

$$M_k(P_s,P_t) = \|\mathbb{E}_{P_s}[\phi(\mathbf{X}_s)] - \mathbb{E}_{P_t}[\phi(\mathbf{X}_t)]\|_{\mathcal{H}}^2 \tag{2}$$

Denote by $\mathbf{X}_s$ and $\mathbf{X}_t$ as the source domain and target domain respectively, let ϕ is the nonlinear feature mapping function, $\mathcal{H}$ denoted as reproducing kernel Hillbert space. According to the definition of MMD, if $M_k(P,Q)=0$, then $P=Q$. As mentioned above, since the existence of covariate shift, aligning the marginal distribution of the same category between domains is a suitable metric function, and is beneficial for UDA.

However, the MMD-based approaches reduce the distributions discrepancy by aligning global distributions, but ignores the relationships between the same category across domains. Accordingly, we can not only learn the transferable representations, but also ensure that the learned representations are more discriminative by considering the correlation between the same category within different domains. Thus, a LMMD [15], [24] is proposed by minimizing the class conditional discrepancy, which can be define as:

$$M_l(P_s,P_t) = \mathbb{E}_c \left\|\mathbb{E}_{P_s^c}[\phi(\mathbf{X}_s)] - \mathbb{E}_{P_t^c}[\phi(\mathbf{X}_t)]\right\|_{\mathcal{H}}^2 \tag{3}$$

where P_s^c, P_t^c represent the distribution with category label c in source domain and target domain respectively. Since LMMD capture fine-grained information across different domains, the model can learn a more transferable representation of features.

Unfortunately, the labels of the target domain is unavailable for UDA. To measure the distribution discrepancy based on LMMD [15], [24], [25], some existing methods use the estimated pseudo label $\hat{y}$ replaces ground-truth labels. It is important to note that, the accuracy of label estimated affects the performance of LMMD. When the model predicts incorrectly, aligning the class conditional distribution will reduce the model's transferability. Fortunately, the output of the network can be regarded as the probability distribution of the label, whose value represents the probability that the sample belongs to the category. Thus, DSAN [24] uses the probability value to weight the all samples in the target domain, which may eliminate the influence of noisy pseudo-label. Motivated by DSAN, we compute the distribution discrepancy $M_k(\mathcal{D}_s,\mathcal{D}_t)$ by using hard pseudo labels for domain adaptation, which can further remove the target domain samples with low confidence. Note that when calculating the LMMD, we use the target domain samples within the same pseudo category instead of the whole target domain data in DSAN. Given $\mathcal{D}_s$ and $\mathcal{D}_t$ represent the source domain and the target domain drawn from distribution P_s and P_t respectively. Base on the analysis mentioned above, the empirical estimation of LMMD in our paper is redefined as follows.

$$\begin{aligned}\hat{M}_l(\mathcal{D}_s,\mathcal{D}_t) &= \frac{1}{C}\sum_{c=1}^{C}\left\|\frac{1}{n_s^c}\sum_{y_i^s=c}\phi(\mathbf{x}_i^s) - w_{ic}^t\sum_{\hat{y}_j^t=c}\phi(\mathbf{x}_j^t)\right\|_{\mathcal{H}}^2 \\ &= \frac{1}{C}\sum_{c=1}^{C}\left[\frac{1}{(n_s^c)^2}\sum_{i=1}^{n_s^c}\sum_{j=1}^{n_s^c}k(\mathbf{z}_i^s,\mathbf{z}_j^s)\right. \\ &\quad + (w_{ic}^t)^2\sum_{i=1}^{n_t^c}\sum_{j=1}^{n_t^c}k(\mathbf{z}_i^t,\mathbf{z}_j^t) \\ &\quad \left. - \frac{2w_{ic}^t}{n_s^c}\sum_{i=1}^{n_t^c}\sum_{j=1}^{n_t^c}k(\mathbf{z}_i^s,\mathbf{z}_j^t)\right]\end{aligned} \tag{4}$$

where n_s^c and n_t^c denote as the number of the c-th class in source and target domains, respectively. z denotes the feature representations. Let $w_{ic}^t = \frac{\hat{y}_{ic}^t}{\sum_{j=1}^{n_t}\hat{y}_{ic}^t}$ denotes as the weight that the target sample i belongs to class c. The above formula uses the uncertainty of samples to eliminate the side effect of noisy pseudo labels.

B. Information Maximization

Unsupervised domain adaptation aims to learn the domain-invariant features across domains, such that the classifier trained in the source domain can correctly classify the representation from target domains. To learn such representation, we revisit domain-invariant feature learning from the perspective of distribution alignment. Specially, we assume that the distribution of source feature representations is $p(\mathbf{z_s}|\mathbf{y_s})$. Since the source domain labels are accessible, the feature representations assumed to be tractable. In the contrast, because there are no labels of related tasks in the target domain, the distribution of target feature representations is intractable. The target feature distribution is defined as $q(\mathbf{z_t})$. To learn the discriminative feature distribution, the feature alignment of UDA can be formalized to minimize the

KL distance between the above two distributions.

$$\begin{aligned} &KL(q(\mathbf{z_t})||p(\mathbf{z_t}|\mathbf{y_s})) \\ &= \sum q(\mathbf{z_t}) \log \frac{q(\mathbf{z_s})}{p(\mathbf{z_s}|\mathbf{y_s})} \\ &= \sum q(\mathbf{z_t}) \log \frac{q(\mathbf{z_t})p(\mathbf{y_s})}{p(\mathbf{z_s},\mathbf{y_s})} \\ &= \sum q(\mathbf{z_t}) \log \frac{q(\mathbf{z_t})p(\mathbf{y_s})}{p(\mathbf{z_s})p(\mathbf{y_s}|\mathbf{z_s})} \\ &= \sum q(\mathbf{z_t}) \log \frac{q(\mathbf{z_t})}{p(\mathbf{z_s})} + \sum q(\mathbf{z_t}) \log \frac{p(\mathbf{y_s})}{p(\mathbf{y_s}|\mathbf{z_s})} \\ &= KL(q(\mathbf{z_t})||p(\mathbf{z_s})) - H(\mathbf{y_s}) + H(\mathbf{y_s}|\mathbf{z_s}) \end{aligned} \tag{5}$$

where $KL(q||p)$ is the distance between p and q. This objective can be divided into two items: minimizing the discrepancy of feature distribution and maximizing the mutual information between the outputs of the classifier and task-relevant representations on source domain. The former encourages to knowledge transfer across domain, while the latter can learn discriminative task-relevant feature representations. Due to the effectiveness of distribution alignment between classes, LMMD used as the distribution discrepancy measure in our experiment. According the information theory, the mutual information $I(\mathbf{z};\mathbf{y})$ [33] between the learned features $\mathbf{z} = p(\mathbf{z}|\mathbf{x})$ and the label $\mathbf{y}$ can be formalized as follows.

$$\begin{aligned} I(\mathbf{z};\mathbf{y}) &= H(\mathbf{y}) - H(\mathbf{y}|\mathbf{z}) \\ &= \underbrace{\mathbb{E}_{\mathbf{y}}\left[p(\mathbf{y})\right]}_{\mathcal{L}_{div}} - \underbrace{\mathbb{E}_{(\mathbf{z},\mathbf{y})}\left[-\log p(\mathbf{y}|\mathbf{z})\right]}_{\mathcal{L}_{ent}} \end{aligned} \tag{6}$$

where $H(\mathbf{p}) = -\sum_i^n p_i \log p_i$ represents the Shannon entropy of variable $\mathbf{p} = \{p_i\}_{i=1}^n$. According to information theory, it represents the uncertainty of the system. The smaller the entropy, the more stable the system. The mutual information in (5) can be decomposed into two separate loss term $\mathcal{L}_{div} = H(\mathbf{y})$ [40], [48] and $\mathcal{L}_{ent} = H(\mathbf{y}|\mathbf{z})$ [30], which indicates that the mutual information depends on both the discriminability and the diversity of prediction output. For information maximization, an intuitive explanation is that maximizing mutual information can be seen as encouraging the model to produce unambiguous clustering assignment, i.e. discriminability or close to one-hot encodings, while encouraging the uniform cluster size, i.e. diversity or class balance.

Source Mutual Information: Based on the above analysis, MI employs entropy minimization to encourage the prediction output close to one-hot encodings. Since the availability of labels for the source domain, we model entropy minimization by minimizing the cross entropy between the prediction output $\hat{\mathbf{y}}$ and the true label $\mathbf{y}$ in practice. Assuming that the true distribution of the source domain data is p_s and the empirical distribution is $\hat{p}_s$, the mutual information between the learned representation $\mathbf{z}_s$ and labels $\mathbf{y}_s$ is recorded as $I(\mathbf{z}_s;\mathbf{y}_s)$, which can be written as:

$$\begin{aligned} I(\mathbf{z}_s;\mathbf{y}_s) &= H(\mathbf{y}_s) - H(\mathbf{y}_s|\mathbf{z}_s) \\ &= \mathbb{E}_{\mathbf{y}_s}\left[-\log p_s(\mathbf{y}_s)\right] - \mathbb{E}_{(\mathbf{z}_s,\mathbf{y}_s)}\left[-\log p_s(\mathbf{y}_s|\mathbf{z_s})\right] \\ &\approx \sum_{i=1}^{C}\left[-p_s(\bar{y}_i^s)\log p_s(\bar{y}_i^s)\right] \\ &\quad - \frac{1}{n_s}\sum_{i=1}^{n_s}\left[-\log \hat{p}_s(\mathbf{y}_s|\mathbf{z}_s)\right] \\ &\approx \sum_{i=1}^{C}\left[-p_s(\bar{y}_i^s)\log p_s(\bar{y}_i^s)\right] - \frac{1}{n_s}\sum_{i=1}^{n_s}\ell(f(\mathbf{x}_i^s), y_i^s) \\ &= \mathcal{L}_{div}^s - \mathcal{L}_{cls} \end{aligned} \tag{7}$$

where C is the number of classes, and $\bar{\mathbf{y}}_s = \frac{1}{n_s}\sum_{i=1}^{n_s}\hat{y}_i^s$ is the mean value of the prediction results of the batch source data. Note that $\mathcal{L}_{div}^s - \mathcal{L}_{cls}$ can be regarded as a empirical estimation of the mutual information on source domain, where $\mathcal{L}_{div}^s$ is the diversity loss and $\mathcal{L}_{cls}$ is the cross entropy loss.

Weighted Target Mutual Information: To effectively utilize the target domain, some existing methods use predictive pseudo labels to learn better target task-relevant representations. Due to the distribution discrepancy, the generated pseudo labels are usually noisy and inaccurate. Currently works improve the quality of pseudo labels by maximizing the mutual information on the target domain. Although mutual information maximization enforces the learning of task-relevant feature representations, it may still align the target feature representation to the wrong category. Thus, directly maximizing mutual information on the target domain may reduce the generalization performance of the learned model.

To cope with this problem, we reweighting the mutual information on the target domain by the confidence score of the outputs of the classifier. We take these output results as the classification confidence score. Intuitively, the higher the confidence score, the greater the possibility of correct classification of samples in the target domain. Therefore, a natural strategy is to select samples whose confidence score is higher than the threshold. However, this sample selection strategy is too strict to effectively use the target domain data. Therefore, we propose to use the classification accuracy of the current batch as the confidence score of mutual information in the target domain. We re-weighting the mutual information on target domain by the mean of the prediction confidence. First, we estimate the pseudo label $\tilde{y}_i^t = \arg\max_c p_c(\mathbf{x}_i^t)$ to sample $\mathbf{x}_i^t$ with the predict probability $\hat{y}_i^t = p(\mathbf{x}_i^t)$. Then, the weight w can be estimated by

$$\hat{w} = \frac{1}{n_t}\sum_{i=1}^{n_t}\mathbb{I}(p_{\tilde{y}_i^t}(\mathbf{x}_i^t) > \tau_0) \tag{8}$$

where τ_0 is the threshold, and $\mathbb{I}$ denoted as indicator function. For example, if 90% samples in mini-batch larger than the threshold τ_0, then the weight $\hat{w} = 0.9$. Based on the weighting strategy introduced above, we then define the weighted target domain

mutual information loss as follows.

$$\begin{aligned}\mathcal{L}_{mi}^{t} &= \hat{w} * I(\mathbf{h}_t; \mathbf{y}_t) = \hat{w} * (\mathcal{L}_{div}^{t} - \mathcal{L}_{ent}^{t}) \\ \mathcal{L}_{ent}^{t} &= H(\hat{\mathbf{y}}_t|\mathbf{h}_t) = -\frac{1}{n_t}\sum_{i=1}^{n_t}\sum_{c=1}^{C}\hat{y}_{ic}^{t}\log(\hat{y}_{ic}^{t}) \\ \mathcal{L}_{div}^{t} &= H(\bar{\mathbf{y}}_t) = -\sum_{c=1}^{C}\bar{y}_{c}^{t}\log(\bar{y}_{c}^{t}) \end{aligned} \tag{9}$$

where $\hat{y}_{i,c}^{t}$ is the predication probability of the class c at sample i for target domain, $\bar{\mathbf{y}}_t = \frac{1}{n_t}\sum_{i=1}^{n_t}\hat{y}_i^t$ is the mean value of the prediction results of the target domain, representing the probability distribution of the category. Although the predicted output can be inaccurate. Since MMD can effectively reduce the distribution discrepancy across domains, the prediction probability of the target domain is closer to the real probability with the increase of the number of iterations.

This work is related to the work that maximizing the mutual information on the target domain, such as SHOT [45], MEDM [48], DPN [34], CLIM [35]. Another related work is predictive reweighting [49], which uses additional discriminators to measure the similarity of samples in the target domain. Unlike this method, our method uses classifier output to estimate weights without introducing additional module. Different from these methods, we re-weighting the estimated mutual information by using the average of the prediction confidence in mini-batch. On one hand, this weighting strategy reduces the side-effect of noisy prediction, which is benefit the adaptation. On the other hand, enhancing the ability of learn the task-relevant features can be regarded as aligning the feature distribution implicitly. We would like note that our proposed method improves the prediction diversity by maximizing the entropy of the average prediction without additional structure or priors information, which shows that our proposed approach is simple and easy to implement.

C. Batch Label Distribution Priors

During the network training, a mini-batch is randomly sampled from the whole data set. Due to the limit space, for class C, the mini-batch may not contain samples from class C. However, the diversity L_{div} in information maximization forces the predicted label probabilities fit uniform distribution over all category, including classes not exist in the mini-batch, which may damage the adaptive performance. To solve the above challenge, we propose a priors regularization term for label distribution to prevent the predicted label distribution far away from the real label distribution.

The label distribution priors is defined as the empirical distribution of labels in mini-batch. Specifically, Let $\bar{\mathbf{p}}_s \in \mathbb{R}^C$ represent a priori probability distribution. If there are samples with label c in mini-batch, then $\bar{p}_s(c) = 1$. To guarantee $\sum_{i=1}^{B}\bar{p}_s(c)$ equal to 1, the corresponding label distribution priors $\bar{\mathbf{p}}_s = \{\bar{p}_s(c)\}_{i=1}^{C}$ is computed by $\bar{p}_s(c) = \bar{p}_s(c)/\sum_{i=1}^{B}\bar{p}_s(c)$.

To incorporate label distribution priors into the proposed method, we add a regularization term that minimizing the discrepancy between the empirical label distribution and the real label distribution with ℓ_1-norm distance. This regularization loss $\mathcal{L}_{reg}$ can be defined as

$$\mathcal{L}_{reg} = \|\bar{\mathbf{y}}_s - \bar{\mathbf{p}}_s\|_1 \tag{10}$$

where $\bar{\mathbf{y}}_s = \frac{1}{B}\sum_{i=1}^{B}\hat{y}_i^s$ is the average of the predicted output over the sample dimension for each source domain mini-batch. Note that the label distribution priors are calculated only in the source mini-batch. In this way, we can apply this regularization term to enhance the consistency between the predicted label distribution and the real label distribution in mini-batch, and to reduce the risk of diversity collapse. Experiments show that the label distribution priors can achieve better adaptation performance in most transfer tasks.

D. Information Maximization Adaptation Networks With Label Distribution Priors

According to the analysis in Sections III-B. and III-C., we propose an information maximization adaptation network by integrating feature adaptation, batch label priors distribution and mutual information. Different from previous adaptation methods, our method employs the mutual information across domains to boost the generalization of the learned feature represents, and batch label distribution priors to prevent the problem of diversity collapse. In deep network, we need to minimize the domain discrepancy over the penultimate layer of the proposed network with LMMD [24]. Besides, we train the network by combing the label distribution priors regularization $\mathcal{L}_{reg}$ with the mutual information loss, i.e. $\mathcal{L}_{div}^{s} - \mathcal{L}_{cls}$ for source domain and $\mathcal{L}_{mi}^{t}$ for target domain. Therefore, the final object function can be formulated as:

$$\mathcal{L} = \mathcal{L}_{cls} - \mathcal{L}_{div}^{s} + \gamma\mathcal{L}_{reg} + \beta\hat{M}_l(\mathcal{D}_s, \mathcal{D}_t) - \lambda\mathcal{L}_{mi}^{t} \tag{11}$$

where and $\lambda, \gamma \geq 0$ are weighting factors. And the β is the coefficient of the domain adaptation loss $\hat{M}_l(\mathcal{D}_s, \mathcal{D}_t)$, we fixed it to 0.3 in our experiments following DSAN [24].

It is worth noting that enforcing the diversity may be misclassify the samples from majority classes as minority classes. Still, for labeled source domain data, minimizing both the regularization priors of label distribution and classification loss will punish the wrongly encouraged diversity in mini-batch. For target domain data, as the progress of training, the LMMD loss in our proposed method will gradually reduce the distribution discrepancy across domains. Based on this observation, entropy minimization plays the same role as minimizing classification loss. Besides, the confidence re-weighting mechanism can enhance the robustness to noisy label and improve the adaptation performance. We have verified the above viewpoint in the experimental analysis. Similar in spirit, BNM increases both the prediction diversity and discriminability by maximizing nuclear-norm of the classifier output matrix. Different from BNM, our method does

not need matrix decomposition, so its computational overload is lower than that of BNM.

E. Theoretical Analysis

In this section, we analysis the effectiveness of our method on the target domain base on the theory of domain adaptation [50], [51].

Theorem 1: ([50], [51]) Let H be the hypothesis class, given two different but related domains $\mathcal{D}_s$, $\mathcal{D}_t$, we have

$$\forall h \in \mathcal{H}, \mathcal{R}_t(h) \leq \mathcal{R}_s(h) + \frac{1}{2} d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_t) + C$$

$$wehere$$

$$C = \mathcal{R}_s(h^*) + \mathcal{R}_t(h^*)$$

$$h^* = \min_h \mathcal{R}_s(h, f_s) + \mathcal{R}_t(h, f_t) \tag{12}$$

Note that f_s and f_t denote the ground truth labeling function on the source domain and the target domain, respectively. Let $\mathcal{R}_s(h)$, $\mathcal{R}_t(h)$ are the corresponding expected error on for the source domain and target domain. This theorem show that the expected on the target domain upper bounded the following three items. The first term is achieved by minimizing classification loss in the training process. And the second term denoted as the discrepancy of two distinct domains, there are many methods to minimize $d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_t)$, such as MMD, LMMD. The last term C denote as the shared expected error for source and target domain with a little abuse of notations. Most existing methods assume that C is fixed when the domains is given. However, when the feature alignment is insufficient across domain, C will become larger. Considering the change of C, we further analyze its upper bound.

Unfortunately, the target domain label is not accessible in UDA. Therefore, we propose to use the pseudo label of the target domain to approximate estimate its upper bound. For labeling function f_1, f_2, and f_3, it have the following triangular inequality [15]: $\mathcal{R}(f_1, f_2) \leq \mathcal{R}(f_1, f_3) + \mathcal{R}(f_2, f_3)$. Thus, the upper bound of C can be further derived as :

$$\begin{aligned} C &= \min_{h \in H} \mathcal{R}_s(h, f_s) + \mathcal{R}_t(h, f_t) \\ &\leq \min_{h \in H} \mathcal{R}_s(h, f_s) + \mathcal{R}_t(h, f_s) + \mathcal{R}_t(f_s, f_t) \\ &\leq \min_{h \in H} \mathcal{R}_s(h, f_s) + \mathcal{R}_t(h, f_s) + \mathcal{R}_t(f_s, \hat{f}_t) \\ &\quad + \mathcal{R}_t(\hat{f}_t, f_t) \end{aligned} \tag{13}$$

where $\hat{f}_t$ is the pseudo label functions. The first two terms denote the disagreement between f_s and h on the source domain and the target domain, respectively. Due the function h learn from the source domain, the above two term would be minimized in training phase. The last term $\mathcal{R}_t(\hat{f}_t, f_t)$ represents the expected error of the pseudo labeling function on the target domain. As train proceeds, it can still be minimized by maximizing the mutual information and feature alignment in the target domain.

We mainly focus on the third term $\mathcal{R}_t(f_s, \hat{f}_t)$, which represents the difference between the source labeling function and the target pseudo labeling function. On the one hand, when the feature distribution across domains is aligned, the feature distance of samples from the same class across domains is expected to be close. On the other hand, since class imbalance, i.e. many samples exist in majority category, minimizing the loss of cross entropy is easy to be misled by the majority category and cause the degenerate solution. In our method, maximizing mutual information tradeoff the maximizing the label entropy and the minimizing the cross entropy. The former encourages the diversity of prediction, which can effectively eliminate the influence of class imbalance and avoid the degradation solutions. To conclude, ours can further reduce the generalization error of the learned model.

IV. Experiments

To validate the effectiveness of our proposed method, we carry out extensive experiments on three benchmark datasets, including Office-31, Office-Home, and VisDA-2017, and compared the proposed method with state-of-the-art domain adaptation methods. In addition, we analyze our method from six aspects, i.e., ablation study, feature visualization, distribution discrepancy, parameter sensitivity, batch sizes, and convergence. In our experiments, all domain adaptation tasks are denoted as source domain $\rightarrow$ target domain.

A. Setup

Office-31 [52] is the standard benchmark dataset in domain adaptation, which includes three different object recognition domains: Amazon (**A**), Webcam (**W**), and DSLR (**D**), comprising 4,110 images and 31 categories. From the domains mentioned above, two different domains are randomly selected as the source domain and the target domain, and construct six domain adaptation tasks.

Office-Home [53] is a more challenging dataset in domain adaptation, consists of 65 categories and 15,588 images, which is much larger than Office-31 in the number of images and categories. It contains four different domains: Artistic images (**A**), Clip Art (**C**), Product images (**P**), and Real-World images (**R**). Likewise, we construct 12 domain adaptation tasks among four domains.

VisDA-2017 is a simulation-to-real dataset for domain adaptation, which consist of two very distinct domains: synthetic, where the 3D model rendered from the composite data set under different angles and lighting conditions, and the real, natural images collected from MSCOCO. We use the training and the validation domains as the source domain and the target domain of domain adaptation task respectively.

We compare the proposed method with start-of-the-art deep learning and domain adaptation methods: Res-Net [1], Deep Adaptation Network (DAN) [10], Domain Adversarial Neural Network (DANN) [13], Joint Adaptation Network (JAN) [9], Multi-Adversarial Domain Adaptation (MADA) [16], Maximum Classifier Discrepancy (MCD) [38], Conditional Domain Adversarial Network (CDAN and CDAN+E) [25], Deep Subdomain Adaptation Network (DSAN) [24], Batch Spectral Penalization for Adversarial Domain Adaptation (BSP+CDAN) [22], Entropy Minimization (EntMin) [30], Maximum Square (MaxSquare) [39], Batch Nuclear-Norm

TABLE I
CLASSIFICATION ACCURACY (%) ON OFFICE-31 FOR UDA (RESNET-50)

Method	A→ W	D→ W	W→ D	A→ D	D→ A	W→ A	Average
ResNet [1]	68.4±0.5	96.7±0.5	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.1
PRDA [49]	78.9±0.2	92.4±0.1	96.8±0.1	87.6±0.1	64.7±0.2	63.1±0.1	80.6
DPN [34]	91.5±0.4	**99.5**±0.5	**100.0**±0.0	94.0±0.9	72.2±1.3	68.1±0.1	87.6
DAN [10]	86.3±0.3	97.2±0.2	99.6±0.1	82.1±0.3	64.6±0.4	65.2±0.3	82.5
DANN [13]	82.0±0.4	96.9±0.2	99.1±0.1	79.7±0.4	68.2±0.4	67.4±0.5	82.2
JAN [9]	85.4±0.3	97.4±0.2	99.8±0.2	84.7±0.3	68.6±0.3	70.0±0.4	84.3
MADA [16]	90.0±0.1	97.4±0.1	99.6±0.1	87.8±0.2	70.3±0.3	66.4±0.3	85.2
CDAN [25]	93.1±0.2	98.2±0.2	**100.0**±0.0	89.8±0.3	70.1±0.4	68±0.4	86.6
CDAN+E [25]	94.1±0.1	98.6±0.1	**100.0**±0.0	92.9±0.2	71.0±0.3	69.3±0.3	87.7
BSP+CDAN [22]	93.3±0.2	98.2±0.2	**100.0**±0.0	93.0±0.2	73.6±0.3	72.6±0.3	88.5
GLS [23]	94.5	99.3	**100.0**	90.1	73.1	71.8	88.1
DSAN [24]	93.6±0.2	98.3±0.1	**100.0**±0.0	90.2±0.7	73.5±0.5	74.8±0.4	88.4
MCC [32]	95.5±0.2	98.6±0.1	**100.0**±0.0	94.4±0.3	72.9±0.2	74.9±0.3	89.4
DANN+MCC [32]	**95.6**±0.3	98.6±0.1	99.3±0.0	93.8±0.4	74.0±0.3	75.0±0.4	89.4
CDAN+MCC [32]	94.7±0.2	98.6±0.1	**100.0**±0.0	**95.0**±0.1	73.0±0.2	73.6±0.3	89.2
DWL [21]	89.2	99.2	**100.0**	91.2	73.1	69.8	87.1
SCDA [19]	94.2	98.7	99.8	95.2	75.7	76.2	**90.0**
SENTRY*(w/o. class balancing) [42]	80.1	98.5	99.8	76.5	67.8	71.7	82.4
SENTRY* [42]	89.7	97.9	99.8	91.2	72.8	72.7	87.3
EntMin [30]	89.0±0.1	99.0±0.1	**100.0**±0.0	86.3±0.3	67.5±0.2	63.0±0.1	84.1
MaxSquare [39]	92.4±0.5	99.1±0.1	**100.0**±0.0	90.0±0.2	68.1±0.4	64.2±0.2	85.6
BNM [31]	91.5	98.5	**100.0**	90.3	70.9	71.6	87.1
CDAN+BNM [31]	92.8	98.8	**100.0**	92.9	73.5	73.8	88.6
Ours(w/o. re-weighting)	93.1±0.2	98.8±0.1	**100.0**±0.0	90.8±0.4	74.7±0.7	74.6±0.2	88.7
Ours	93.3±0.5	99.0±0.1	**100.0**±0.0	93.2±0.3	**76.7**±0.2	**76.2**±0.3	89.7
CAN(intra only) [29]	93.2±0.2	98.4±0.2	99.8±0.2	92.9±0.2	76.5±0.3	76.0±0.3	89.5
CAN [29]	94.5±0.3	99.1±0.2	99.8±0.2	95.0±0.3	78.0±0.3	77.0±0.3	90.6

(BNM and CDAN+BNM) [31], Minimum Class Confusion (MCC, DANN+MCC, and CDAN+MCC) [32], Generalized Label Shift (GLS) [23], Transferrable Prototypical Networks (TPN) [28], Semantic Concentration for Domain Adaptation (SCDA) [19],Selective Entropy Optimization (SENTRY, SENTRY(w/o. class balancing)) [42], Adversarial Domain Adaptation with Domain Mixup (DM-ADA) [20], Dynamic Weighted Learning (DWL) [21], Prediction Reweighting Domain Adaptation(PRDA) [49], Domain Preservation Nets((DPN) [34], and Contrastive Adaptation Network (CAN and CAN(intra only)) [29].

We use **ResNet** [1] pre-trained on ImageNet to learn the transferable representation. In order to perform safe transferable representation learning, we use the same strategy as CDAN, adding a bottleneck layer of 256 neural units after the final average pooling layer. To make a fair comparison, the same network structure was used in all experiments (we use ResNet50 for Office31 and Office-Home, and use ResNet101 for VisDA-2017). Following the standard protocol of UDA, we use all labeled source domain data and unlabeled target domain data as the training dataset. We fine-tune all the convolution layers and pooling layers and trained the classifier layer from scratch via the back-propagation algorithm. Since the classifier layer is trained from scratch, we set the learning rate to be ten times that of the fine-tuning layers. All experiments in this paper use the mini-batch stochastic gradient descent algorithm with momentum of 0.9 and the learning rate annealing strategy in Revgrad. Due to the high computational cost of grid search, the learning rate is adjusted dynamically by the following formula [13]: $\eta_\theta = \eta_0/(1+a\theta)^b$, where θ is the training progress linearly changing from 0 to 1, $\eta_0 = 0.003$ for VisDA-2017, $\eta_0 = 0.01$ for others, $a = 10$, and $b = 0.75$. This learning rate updating strategy promote convergence of algorithm and low error on the source domain. For Office-31, $\lambda = 0.2$, while for Office-Home and VisDA-2017, $\lambda = 0.3$. The γ selected is 0.5. The threshold τ_0 is set 0.9 for Office-31, 0.6 for Office-Home, and 0.7 for VisDA2017, respectively. To suppress noise activations in the early stages of training, we do not fix the adaptation factor, but gradually change it from 0 to 1 with a progressive schedule: $\lambda_\theta = 2/\exp(-\rho\theta) - 1$, and $\rho = 10$ is fixed through the experiments. For the LMMD used in our approach, we use Gaussian kernel with the bandwidth set to the paired square distance of the median of the training data [54]. Our method is implemented by PyTorch. We run three random experiments and report the average. For the purpose of fair comparisons, the experiment results are report directly from their original paper, if available. Note that * indicates our reproduce results based on the code given by the corresponding paper.

B. Results

As show in Tables I–III, the classification accuracy results of Office-31, Office-Home and VisDA-2017 based on ResNet network are given, respectively. Our approach outperforms all comparison algorithms on most transfer tasks, and significantly improves the classification accuracy of difficult transfer tasks, where the baseline prediction accuracy is relatively low. Taking the Office31 dataset as an example, our approach achieves better

TABLE II
CLASSIFICATION ACCURACY (%) ON OFFICE-HOME FOR UDA (RESNET50)

METHOD	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	AVERAGE
RESNET [1]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN [10]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
DANN [13]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
JAN [9]	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
PRDA [49]	49.8	72.3	73.9	45.7	66.0	66.3	50.6	45.0	74.0	58.1	50.4	76.9	60.7
DPN [34]	51.8	75.3	79.4	66.6	74.8	74.6	63.8	51.7	81.5	74.0	58.0	84.3	69.7
CDAN [25]	49.0	69.3	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
CDAN+E [25]	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
BSP+CDAN [22]	52.0	68.6	76.1	58.0	70.3	70.2	58.6	50.2	77.6	72.2	59.3	81.9	66.3
GLS(IWCDAN) [23]	52.3	74.5	78.5	60.3	70.8	71.5	62.6	50.7	78.9	72.4	57.8	81.3	67.6
DSAN [24]	54.4	70.8	75.4	60.4	67.8	68.0	62.6	55.9	78.5	73.8	60.6	83.1	67.6
DWL* [21]	45.6	63.9	72.2	55.5	60.8	64.6	58.0	46.8	73.9	69.3	52.0	78.0	61.7
SCDA [19]	57.5	76.9	**80.3**	65.7	74.9	74.5	65.5	53.6	79.8	**74.5**	59.6	83.7	70.5
SENTRY*(W/O. CLASS BALANCING) [19]	**59.8**	**78.6**	79.3	63.5	74.2	74.4	66.9	61.1	80.2	73.1	65.8	84.2	71.8
SENTRY [42]	61.8	77.4	80.1	66.3	71.6	74.7	66.8	**63.0**	80.9	74.0	**66.3**	84.1	**72.2**
ENTMIN [30]	43.2	68.4	78.4	61.4	69.9	71.4	58.5	44.2	78.2	71.1	47.6	81.8	64.5
BNM [31]	52.3	73.9	80.0	63.3	72.9	74.9	61.7	49.5	79.7	70.5	53.6	82.2	67.9
CDAN+BNM [31]	56.2	73.7	79.0	63.1	73.6	74.0	62.4	54.8	80.7	72.4	58.9	83.5	69.4
OURS(W/O. RE-WEIGHTING)	57.7	77.0	79.8	66.8	74.1	**75.0**	**67.5**	56.7	81.3	74.2	60.7	**84.7**	71.3
OURS	59.6	**77.3**	79.5	**67.4**	**75.9**	74.6	66.1	56.4	**81.0**	**74.5**	61.4	84.4	71.5

TABLE III
CLASSIFICATION ACCURACY (%) ON VISDA-2017 FOR UDA (RESNET101)

METHOD	PLANE	BCYBL	BUS	CAR	HORSE	KNIFE	MCYLE	PERSN	PLANT	SKTB	TRAIN	TRUCK	AVERAGE
RESNET [1]	72.3	6.1	63.4	91.7	52.7	7.9	80.1	5.6	90.1	18.5	78.1	25.9	49.4
DANN [13]	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
DAN [10]	68.1	15.4	76.5	87.0	71.1	48.9	82.3	51.5	88.7	33.2	88.9	42.2	62.8
JAN [9]	75.7	18.7	82.3	**86.3**	70.2	56.9	80.5	53.8	92.5	32.2	84.5	**54.5**	65.7
MCD [38]	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
BSP+CDAN [22]	92.4	61.0	81.0	57.5	89.0	80.6	90.1	77.0	84.2	77.9	82.1	38.4	75.9
DSAN [24]	90.9	66.9	75.7	62.4	88.9	77.0	**93.7**	75.1	92.8	67.6	89.1	39.4	75.1
MCC [32]	88.1	80.3	80.5	71.5	90.1	93.2	85.0	71.6	89.4	73.8	85.0	36.9	78.8
DANN+MCC [32]	90.4	79.8	72.3	55.1	90.5	86.8	86.6	80.0	94.2	76.9	90.0	49.6	79.4
CDAN+MCC [32]	94.5	80.8	78.4	65.3	90.6	79.4	87.5	82.2	**94.7**	81.0	86.0	44.6	80.4
TPN [28]	93.7	85.1	69.2	81.6	93.5	61.9	89.3	81.4	93.5	81.6	84.5	49.9	80.4
DWL [21]	90.7	80.20	86.1	67.6	92.4	81.5	86.8	78.0	90.6	57.1	85.6	28.7	77.1
DM-ADA [20]	-	-	-	-	-	-	-	-	-	-	-	-	75.6
SCDA* [19]	95.3	77.6	80.6	56.4	95.0	6.5	85.0	82.0	89.4	80.9	84.6	47.5	73.4
SENTRY*(W/O. CLASS BALANCING) [42]	94.5	**87.3**	75.7	45.1	95.9	**94.8**	85.8	80.7	92.1	**95.6**	**90.3**	53.5	82.6
SENTRY* [42]	**95.6**	86.0	**88.8**	68.2	**96.5**	93.5	89.9	**82.3**	92.9	94.7	84.7	41.3	**84.5**
OURS(W/O. RE-WEIGHTING)	95.0	83.2	75.9	65.8	95.2	80.3	87.6	79.4	93.1	76.5	87.5	53.2	81.0
OURS	94.5	85.4	77.2	65.2	94.8	82.3	86.1	81.4	93.0	77.4	88.6	50.5	81.4
CAN(INTRA ONLY) [29]	96.5	72.1	80.9	70.8	94.6	98.0	91.7	84.2	90.3	89.8	89.4	47.9	83.9
CAN [29]	97.0	87.2	82.5	74.3	97.8	96.2	90.8	80.7	96.6	96.3	87.5	59.9	87.2

average performance compared with state-of-the-art methods, and significantly improves the classification accuracy on two of six transfer tasks (i.e. **D**→**A**, **W**→**A** in Office-31). The experimental results show that our approach can learn more transferable and discriminable features, especially on difficult transfer tasks. From the experimental results, we can make several interesting findings.

1) The performance of our method is better than ResNet on all transfer tasks. Deep neural network has better learning ability and can learn abstract representation. This also confirms that deep learning can reduce the distribution discrepancy between domains, but cannot remove it.

2) In our experiment, MaxSquare, EntMin, and BNM can improve the performance of the model, and our approach is significantly better than them on most transfer tasks. In addition, compared with complex mechanism methods, such as CDAN+BNM, ours also achieves more signification improvement, and significantly improves the performance of difficult transfer tasks (e.g. **W**→**A** on Office-31 and **P**→**C** on Office-Home). This further proves that our approach can effectively learn more transferable features for domain adaptation.

3) In UDA, the performance of the LMMD-based methods (e.g. DSAN [24], Ours) is better than the MMD-based methods mentioned above (e.g. DAN [10], JAN [9]). This shows that aligning the domain distribution with the same category across domains is the crucial for domain adaptation. The main reason is that the MMD-based methods aligning the global distribution without considering the relationship between the category, which confirms the effectiveness of LMMD in domain adaptation.

4) Compared with the adversarial-based adaptation methods [14], [15], [16], [17], our approach achieves better result on most transfer tasks, and obtains comparable performance in other tasks. It should be emphasized that our method does not need additional structure and discriminator, which shows that the proposed approach is simple and easy to implement.

5) We compare our approach with LMMD-based methods, such as DSAN. The results show that our method has achieved significant performance improvement on most transfer tasks. Especially on the hard transfer tasks, it shows that the diversity of prediction is important for the performance of the algorithm. Note that LMMD-based methods require pseudo labels in the target domain to align the data distribution with the same category. Compare to these methods, our method can effectively learn representations with task-relevant, and is more robust to noisy pseudo labels.

Generally speaking, it is difficult to measure the superiority of a method in UDA. Indeed, we find not a single method

TABLE IV
THE AVERAGE CLASSIFICATION ACCURACY (%) OF CAN AND OURS FOR SIX TASKS ON OFFICE-31, AND SYNTHETIC-TO-REAL TASK ON VISDA-2017 ARE REPORTED

Dataset	w/o. AO	w/o. CAS	CAN	Ours
Office-31	88.1	89.1	90.6	89.7
VisDA-2017	77.5	81.6	87.2	81.4

outperform state of the art on all benchmarks. In addition, although CAN [29] has achieved remarking performance, it relies on alternating optimization (AO) and class-aware sampling (CAS). In contrast, the mean accuracy of our method (89.7%) outperforms the method "CAN without AO" (88.1%) by 1.6% and the method "CAN without CAS" (89.1%) by 0.6% on Office31 respectively, and achieves superior or comparable performance on VisDA-2017. The detailed results are shown in Table IV. Note that our algorithm does not need to calculate the inter-class discrepancy, thus it is simple and easy to implement. We want to emphasize that our method is slightly worse than SCDA [19] on the Office31 dataset. However, for Office-Home and VisDA2017, our prediction accuracy is significantly better than that of SCDA. This shows that our method can learn discriminative, domain-invariant feature representations. In addition, due to the class imbalance used by SENTRY [42], the performance of sentry algorithm is better than our method on Office-Home and VISDA, but the performance on Office31 home is worse than ours. To make a fair comparison, we compared our method with Sentry (w/o class balancing), which trained without class balancing. It is worth mentioning that the results in Tables I–III show that our method is slightly worse than SENTRY (w/o class balancing) on Office-Home and VisDA2017, but significantly better than it on Office31.

Domain adaptation theory proves that the distribution discrepancy across domains plays a key role in domain adaptation, i.e., the larger the distribution discrepancy, the better the transfer performance. Take Office-31 data as an example. Through domain alignment, domain **W** is similar to domain **D**, but they are quite dissimilar to domain **A**. The experimental results as shown in Table I confirm this finding: when the distribution discrepancy is large, the model generalization performance is poor; when the distribution discrepancy is small, the transfer performance is good. Interestingly, we find the asymmetric property of domain adaptation: the difficulty of transfer from source domain **S** to target domain **T** is different from that of **T** to **S**. Specifically, when the source domain is large, the transfer task is easier. For example, **A**→**W** is easier than **W**→**A**.

In summary, experimental results on different domain datasets show that the performance of our algorithm is superior to the existing methods in most transfer tasks, especially in difficult transfer tasks where the baseline prediction accuracy is low. The main reason is that existing methods learning domain invariant representations by aligning the distribution discrepancy across domain. However, under the label shift, GLS show that the reduction of distribution discrepancy will damage the generalization performance of the model. In contrast, our method can reduce the negative impact of label shift to a certain extent by learning the feature representations with task-relevant information.

C. Empirical Analysis

Ablation Study: To explore the contribution of each component in our method, such as the diversity in source mutual information $\mathcal{L}_{div}$, weighted target mutual information $\mathcal{L}^t_{mi}$, and label distribution priors $\mathcal{L}_{reg}$. We conducted ablation study on six adaptation tasks of Office-31 dataset. We compare its performance with the model adding different losses to verify the effectiveness of each loss. For instance, the first row in Table V shows the domain adaptation method using only classification loss and LMMD, called ours (w/ LMMD). Experimental results show that each part of our approach has its indispensable contribution, and our method achieves the best performance. It proves that our method can improve the classification performance of the model and is benefit to adaptation.

Interestingly, when only add $\mathcal{L}_{div}$ loss term into the ours (w/ LMMD) method (the second row), the method also outperforms ours (w/ LMMD) adaptation method. This proves that the prediction diversity can make the LMMD metric robust to noisy label to some extent, and maximizing mutual information is an effective way for learning domain invariant representation.

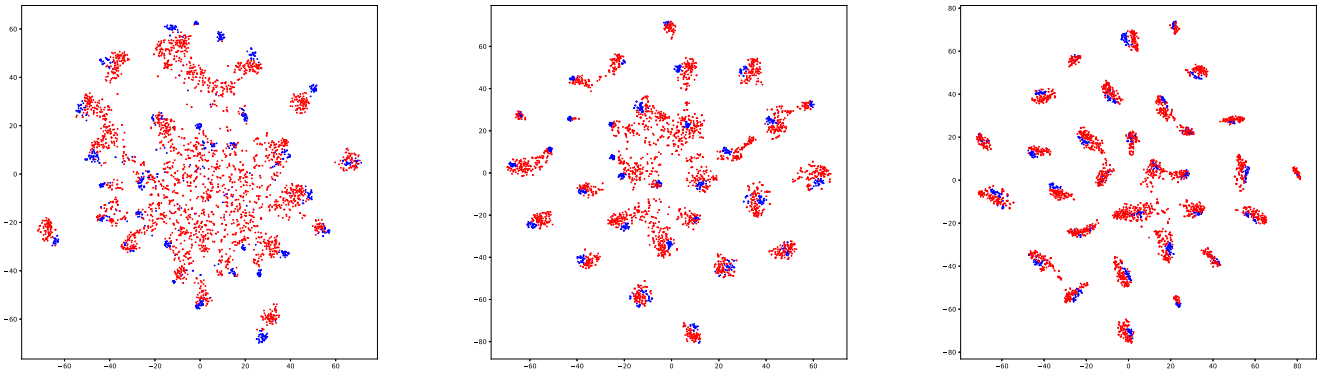
The experimental results in Tables I–III examine the weighted target domain mutual information in our method. We leave one-component-out of our method to perform ablation study at a time. Ours (w/o reweighting) directly uses the target mutual information, which means we train the model without reweighting the target mutual information. The experimental results validate the contribution of weighted target mutual information in our method.

Feature Visualization: We use t-SNE [55] to analysis visualize the output of transfer features learned by DAN, DSAN, and ours on task **W**→**A**, respectively. Compared with DAN and DSAN, our model is more discriminative, as show in Fig. 1(a)–(c). It also shows that our method can effectively learn domain invariant representations for UDA. The blue points indicates the source domain samples, and the red dots indicate the target domain samples. Fig. 1(a) shows the results of DAN, a method to align global distribution discrepancy using MMD. We find that most classes between the source domain and the target domain are not well aligned. The obvious difference is that Fig. 1(b) aligns the feature distribution among the same category. The clustering between same classes is relatively small, and the distance between different classes is large, but there are still data that are hard to classify. Compared with the above algorithm, it is obviously that there are fewer difficult classification samples. This result shows that our model can effectively learn domain invariant features and ensure the discriminability of features, as shown Fig. 1(c).

Distribution Discrepancy: Ben David et al. proposed $\mathcal{A}$-distance to measure the discrepancy across domains [50], [51], which can be defined as $d_{\mathcal{A}} = 2(1 - 2\epsilon)$. Where ϵ is the generalization error of the trained domain classifier to distinguish the source domain from the target domain. Fig. 2(a) shows the $d_{\mathcal{A}}$ distance on the **A**→**W** and **W**→**A** task for Office-31, respectively. We compare our approach to the DSAN and DAN. From the figure, we can observe that $d_{\mathcal{A}}$ using our model is smaller than $d_{\mathcal{A}}$ using DAN, which means that our model can reduce the gap between domains effectively, which also

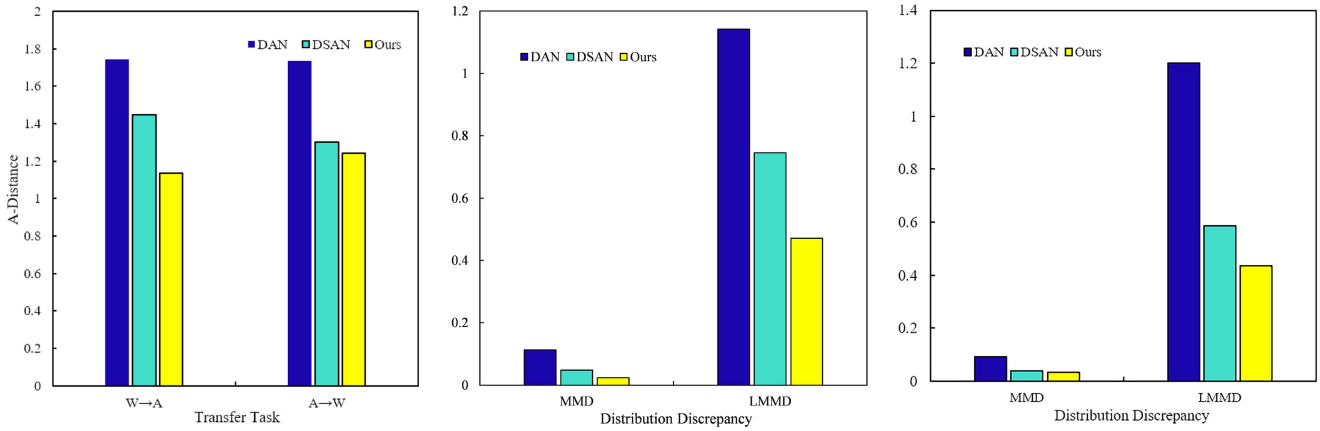
TABLE V
ABLATION STUDY TO INVESTIGATE THE EFFECT OF EACH COMPONENT ON OFFICE-31 FOR UDA (RESNET-50)

$\mathcal{L}_{div}^{s}$	$\mathcal{L}_{mi}^{t}$	$\mathcal{L}_{reg}$	A→ W	D→ W	W→ D	A→ D	D→ A	W→ A	Average
			92.5	98.1	100.0	90.4	74.8	72.5	88.0
✓			92.5	99.1	100.0	91.6	76.3	75.0	89.1
✓	✓		93.3	99.0	100.0	91.4	76.3	75.6	89.3
✓		✓	93.0	99.1	100.0	91.6	76.8	76.1	89.4
✓	✓	✓	93.3	99.0	100.0	93.2	76.7	76.2	89.7



(a) DAN (b) DSAN (c) Ours

Fig. 1. The t-SNE visualization of transferable features (ResNet) generated by DAN (a), DSAN (b), and Ours (c) on task W → A, respectively. Blue Points are source samples and red are target sample.



(a) A-Distance (b) Distribution Discrepancy (c) Distribution Discrepancy

Fig. 2. (a) $\mathcal{A}$-distance on task A → W and W→ A. (b) MMD and LMMD on task W → A. (c) MMD and LMMD on task D → A.

shows that LMMD can effectively align the data distribution. Interestingly, compared with DSAN, the results are close. But our method is significantly outperforming than DSAN in classification accuracy. The above findings show that our proposed algorithm not only focuses on reducing the distribution discrepancy across domains, but also can learn the features related to task. In other words, only aligning distribution discrepancy is not enough for domain adaptation to ensure the discriminability of the learned features.

MMD and LMMD can measure the global distribution difference and the category distribution discrepancy, respectively. We calculate the MMD and LMMD on **W**→**A** and **D**→**A** tasks by using ground-truth labels and the features extracted from DAN, DSAN and our method, respectively. As shown in Fig. 2(b)–(c), our method is smaller on LMMD and MMD, which shows that our method can effectively reduce the global and local distribution discrepancy. In addition, LMMD is larger than MMD, which means that LMMD is a more rigorous measure. In other words, this further show that only aligning the global distribution is not enough for domain adaptation.

Hyper-Parameter Analysis: The weight factors γ and λ are the two hyper-parameters of our method. We study the sensitivity of the proposed method to these two super parameters on two adaptation tasks **W**→**A** and **D**→**A** of office-31. As

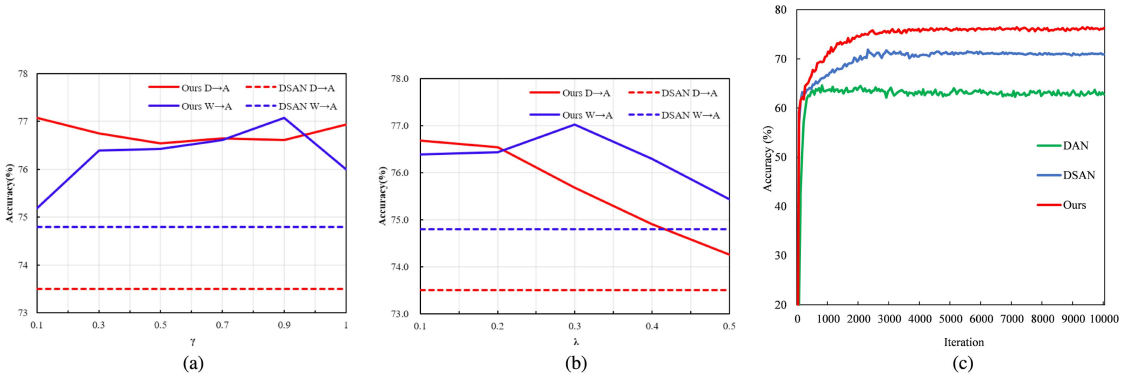


Fig. 3. (a)–(b) Hyper-parameter sensitivity of task W → A. (Left is the sensitivity of accuracy of ours approach to γ, right to λ.) (c) The convergence on task W → A .

TABLE VI
EFFECT OF BATCH SIZES: AVERAGE ACCURACY (%) OF OURS METHOD ON OFFICE-31 AND OFFICE-HOME ARE REPORTED

Batch Size	Office31	Office-Home
24	88.5	70.2
28	89.3	70.8
32	89.7	71.5
36	89.5	71.5
40	89.0	71.7
48	88.7	71.8
56	88.4	71.9

shown in Fig. 3(a)-(b), in a large range of parameters, the performance of our method is better than that of the baseline method DSAN. Specifically, our method is more robust to the change of hyper-parameter γ. For example, when the gamma value is [0.3-0.9], the algorithm achieves better performance, and the accuracy decreases when the value is too small or too large. On the contrary, our method is sensitive to the super parameter λ. With the increase of λ, the accuracy increases steadily before it decreases. The main reason is that the smaller lambda will lead to the vanish of mutual information in the target domain, and the larger λ reduces the accuracy of mutual information estimation in the target domain as it is easily affected by domain shift. The above hyper-parametric variation curves show the regularization effect of both mutual information and label distribution a priors in the target domain.

Effect of Batch Sizes: To verify the effect of batch sizes on the performance of our method, we conducted ablation study of batch sizes from 24 to 56 on Office31 and Office-Home datasets, respectively. Note that we use the same parameter settings in all batch sizes experiments as that in the comparative experiments.

Table VI shows the hyper-parameter sensitivity of our method based on ResNet-50 on Office31 and Office-Home datasets when using different batch size training. Our method works well in a relatively large range of batch sizes, especially for difficult tasks where the accuracy of baseline is relatively low (Office-Home). Specifically, batch sizes of 32 and 56 achieved the best performance on Office31 and Office-Home, respectively. Interestingly, the behavior of batch sizes on Office31 dataset is significantly different from that on Office-Home. With the increase of batch sizes, the accuracy of our method on Office31 first increases and then decreases, while that of Office-Home shows a linear increasing trend. The main reason is that the small bath size hinders the model alignment performance due to the lack of sufficient numbers. Larger batch sizes indirectly increase the learning rate. For simple tasks, larger batch size is difficult to converge. Meanwhile, such batch size introduce more label prior information, which can effectively improve the performance of the model for difficult tasks. This further confirms the effectiveness of our method.

Convergence Performance: We testify their convergence performance on a difficult transfer task **W**→**A**. Fig. 3(c) demonstrates the classification accuracy of different methods on task **W**→**A**, which suggests that our method runs faster than DSAN under the same step situations. Compared with DAN and DSAN, our model is more stable and converges faster, which can speed up the training process. The main reason is that our model will pay attention to the relevant information between tasks and features in the training process, which effectively improve the convergence speed of the model to a certain extent.

V. CONCLUSION

In this paper, we propose an information maximization adaptation network with label distribution priors. We revisit feature alignment from the perspective of distribution alignment in UDA. We also theoretically prove that we can learn discriminative feature representations by maximizing the source mutual information and aligning feature distribution across domains. Based on this observation, we propose to an information maximization network to learn discriminant features by maximizing mutual information between the outputs of the classifier and feature representations. We further propose a reweighted target mutual information by the prediction confidence score to improve the quality of the predicted pseudo labels from target domain. In addition, to prevent the collapse of diversity, a regularization term based on the distribution priors of labels is further introduced to encourage the consistency between the estimated label distribution and the real distribution of labels in mini-batch. Experimental results on three real data sets demonstrate the effectiveness of the proposed method. Our study provides new insights into designing UDA methods.

In further work, we will explore the effect of our proposed method in more vision adaptation task, such as detection and segmentation. Another interesting direction is to extend contrastive learning to UDA, which may improve the adaptive performance as these methods can be regarded as an approximate estimation of mutual information.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 770–778.

[2] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, 2015, pp. 3431–3440.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol.*, Minneapolis, MN, USA, 2019, pp. 4171–4186.

[4] Y. Yang, X. Li, P. Wang, Y. Xia, and Q. Ye, "Multi-source transfer learning via ensemble approach for initial diagnosis of Alzheimer's disease," *IEEE J. Transl. Eng. Health Med.*, vol. 8, pp. 1–10, Apr. 2020.

[5] Y. Yang et al., "Reservoir hosts prediction for COVID-19 by hybrid transfer learning model," *J. Biomed. Inform.*, vol. 117, May 2021, Art. no. 103736.

[6] Y. Yang, Y. Hu, X. Zhang, and S. Wang, "Two-stage selective ensemble of cnn via deep tree training for medical image classification," *IEEE Trans. Cybern.*, vol. 52, no. 9, pp. 9194–9207, Sep. 2022.

[7] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[8] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.

[9] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, Sydney, NSW, Australia, 2017, pp. 2208–2217.

[10] M. Long, Y. Cao, Z. Cao, J. Wang, and M. I. Jordan, "Transferable representation learning with deep adaptation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 12, pp. 3071–3085, Dec. 2019.

[11] W. Zellinger, T. Grubinger, E. Lughofer, T. Natschläger, and S. Saminger-Platz, "Central moment discrepancy (CMD) for domain-invariant representation learning," in *Proc. Int. Conf. Learn. Representation*, Toulon, France, 2017, pp. 1–13.

[12] X. Wu, J. Chen, F. Yu, M. Yao, and J. Luo, "Joint learning of multiple latent domains and deep representations for domain adaptation," *IEEE Trans. Cybern.*, vol. 51, no. 5, pp. 2676–2687, Jun. 2021.

[13] Y. Ganin et al., "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 59, pp. 1–35, Apr. 2016.

[14] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 2962–2971.

[15] S. Xie, Z. Zheng, L. Chen, and C. Chen, "Learning semantic representations for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, Stockholmsmässan, Stockholm, Sweden, 2018, pp. 5419–5428.

[16] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *Proc. AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, 2018, pp. 3934–3941.

[17] W. Zhang, W. Ouyang, W. Li, and D. Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 3801–3809.

[18] T. Shermin, G. Lu, S. W. Teng, M. M. Murshed, and F. Sohel, "Adversarial network with multiple classifiers for open set domain adaptation," *IEEE Trans. Multimedia*, vol. 23, pp. 2732–2744, 2021.

[19] S. Li et al., "Semantic concentration for domain adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Montreal, QC, Canada, 2021, pp. 9082–9091.

[20] M. Xu et al., "Adversarial domain adaptation with domain mixup," in *Proc. AAAI Conf. Artif. Intell.*, New York, NY, USA, 2020, pp. 6502–6509.

[21] N. Xiao and L. Zhang, "Dynamic weighted learning for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Virtual Event, 2021, pp. 15242–15 251.

[22] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, Long Beach, CA, USA, 2019, pp. 1081–1090.

[23] R. T. des Combes, H. Zhao, Y. Wang, and G. J. Gordon, "Domain adaptation with conditional distribution matching and generalized label shift," in *Proc. Adv. Neural Inf. Process. Syst.*, Virtual Event, 2020, pp. 19276–19289.

[24] Y. Zhu et al., "Deep subdomain adaptation network for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 4, pp. 1713–1722, Apr. 2021.

[25] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, Montréal, Canada, 2018, pp. 1647–1657.

[26] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 945–954.

[27] H. Yan et al., "Weighted and class-specific maximum mean discrepancy for unsupervised domain adaptation," *IEEE Trans. Multimedia*, vol. 22, no. 9, pp. 2420–2433, Sep. 2020.

[28] Y. Pan et al., "Transferrable prototypical networks for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, 2019, pp. 2239–2247.

[29] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, 2019, pp. 4893–4902.

[30] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2004, pp. 529–536.

[31] S. Cui et al., "Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2020, pp. 3940–3949.

[32] Y. Jin, X. Wang, M. Long, and J. Wang, "Minimum class confusion for versatile domain adaptation," in *Proc. Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 464–480.

[33] W. Hu, T. Miyato, S. Tokui, E. Matsumoto, and M. Sugiyama, "Learning discrete representations via information maximizing self-augmented training," in *Proc. Int. Conf. Mach. Learn.*, Sydney, NSW, Australia, 2017, pp. 1558–1567.

[34] J. Chen, J. Wang, W. Lin, K. Zhang, and C. W. de Silva, "Preserving domain private representation via mutual information maximization," Jan. 2022. [Online]. Available: https://arxiv.org/abs/2201.03102

[35] T. Li, X. Chen, S. Zhang, Z. Dong, and K. Keutzer, "Cross-domain sentiment classification with contrastive learning and mutual information maximization," in *IEEE Int. Conf. Speech Signal Process.*, Toronto, ON, Canada, 2021, pp. 8203–8207.

[36] B. Sun and K. Saenko, "Deep CORAL: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comput. Vis. Workshops*, Amsterdam, The Netherlands, 2016, pp. 443–450.

[37] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, Australia, 2013, pp. 2200–2207.

[38] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 3723–3732.

[39] M. Chen, H. Xue, and D. Cai, "Domain adaptation for semantic segmentation with maximum squares loss," in *Proc. IEEE Int. Conf. Comput. Vis.,* Seoul, South Korea, 2019, pp. 2090–2099.

[40] G. French, M. Mackiewicz, and M. H. Fisher, "Self-ensembling for visual domain adaptation," in *Proc. Int. Conf. Learn. Representation*, Vancouver, BC, Canada, 2018, pp. 1–15.

[41] R. Xu, G. Li, J. Yang, and L. Lin, "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Seoul, South Korea, 2019, pp. 1426–1435.

[42] V. Prabhu, S. Khare, D. Kartik, and J. Hoffman, "SENTRY: Selective entropy optimization via committee consistency for unsupervised domain adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Montreal, QC, Canada, 2021, pp. 8538–8547.

[43] P. Morerio, R. Volpi, R. Ragonesi, and V. Murino, "Generative pseudo-label refinement for unsupervised domain adaptation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis*., Snowmass Village, CO, USA, 2020, pp. 3119–3128.

[44] T. He, L. Shen, Y. Guo, G. Ding, and Z. Guo, "SECRET: Self-consistent pseudo label refinement for unsupervised domain adaptive person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, *Virtual Event*, 2022, pp. 879–887.

[45] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, Virtual Event, 2020, pp. 6028–6039.
[46] S. M. Ahmed, D. S. Raychaudhuri, S. Paul, S. Oymak, and A. K. Roy-Chowdhury, "Unsupervised multi-source domain adaptation without access to source data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Virtual Event, 2021, pp. 10103–10112.
[47] C. Park, J. Lee, J. Yoo, M. Hur, and S. Yoon, "Joint contrastive learning for unsupervised domain adaptation," Oct. 2020. [Online]. Available: https://arxiv.org/abs/2006.10297
[48] X. Wu et al., "Entropy minimization vs. diversity maximization for domain adaptation," Feb. 2020. [Online]. Available: https://arxiv.org/abs/2002.01690
[49] S. Li, S. Song, and G. Huang, "Prediction reweighting for domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 7, pp. 1682–1695, Jul. 2017.
[50] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2006, pp. 137–144.
[51] S. Ben-David et al., "A theory of learning from different domains," *Mach. Learn.*, vol. 79, no. 1/2, pp. 151–175, 2010.
[52] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. Eur. Conf. Comput. Vis.*, Heraklion, Crete, Greece, 2010, pp. 213–226.
[53] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 5385–5394.
[54] A. Gretton et al., "Optimal kernel choice for large-scale two-sample tests," in *Proc. Adv. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, 2012, pp. 1214–1222.
[55] J. Donahue et al., "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, Beijing, China, 2014, pp. 647–655.

Pei Wang received the B.S. degree in 2014 from the Jianghuai college, Anhui University, Hefei, China, the M.S. degree in 2018 from the School of Information Science and Technology, Yunnan Normal University, Kunming, China. He is currently working toward the Ph.D. degree with the School of Information Science and Engineering, Yunnan University, Kunming, China. His research interests include transfer learning and large-scale data mining.

Yun Yang received the B.Sc. (Hons.) degree in information technology and telecommunication from Lancaster University, Lancaster, U.K., in 2004, the M.Sc. degree in advanced computing from Bristol University, Bristol, U.K., in 2005, and the M.Phil. degree in informatics and the Ph.D. degree in computer science from the University of Manchester, Manchester, U.K., in 2006 and 2011, respectively. From 2012 to 2013, he was a Research Fellow with the University of Surrey, Guildford, U.K. He is currently a Full Professor of machine learning with the National Pilot School of Software, Yunnan University, Kunming, China, the Director of Yunnan Education Department Key Laboratory of Data Science and Intelligent Computing, and Kunming Key Laboratory of Data Science and Intelligent Computing, He is an Associate Editor for the Journal of Yunnan University (Natural Sciences Edition). His research interests include machine learning, data mining, pattern recognition, and temporal data process and analysis.

Yuelong Xia received the M.S. degree in computer software and theory from Yunnan Normal University, Kunming, China, in 2014. He is currently working toward the Ph.D. degree with the Yunnan University, Kunming, China. His research interests include multi-label learning, deep learning, ensemble learning, natural language processing, knowledge graph, and pattern recognition.

Kun Wang was born in Yunnan province, China, in 1989. He received the M.S. degree from the Fuzhou University, Fuzhou, China, in 2015. He received the Ph.D. degree from the Yunnan University, Kunming, China. His research interests include image processing, signal processing, fuzzy set theory and its application, and partial differential equations.

Xingyi Zhang (Senior Member, IEEE) received the B.Sc. degree from Fuyang Normal College, Fuyang, China, in 2003, and the M.Sc. and Ph.D. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2009, respectively. He is currently a Professor with the School of Computer Science and Technology, Anhui University, Hefei, China. His research interests include unconventional models and algorithms of computation, multi-objective optimization, and membrane computing. He was the recipient of the 2018 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award.

Song Wang (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Illinois Urbana-Champaign (UIUC), Champaign, IL, USA, in 2002. From 1998 to 2002, he was a Research Assistant with the Image Formation and Processing Group, Beckman Institute, UIUC. In 2002, he joined the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC, USA, where he is currently a Professor. His research interests include computer vision, image processing, and machine learning. Dr. Wang is currently the Publicity/Web Portal Chair of the Technical Committee of Pattern Analysis and Machine Intelligence of the IEEE Computer Society, an Associate Editor for IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *Pattern Recognition Letters*, and *Electronics Letters*. He is a member of IEEE Computer Society.