

# DOMAIN ADAPTATION VIA MAXIMIZING SURROGATE MUTUAL INFORMATION

**Haiteng Zhao, Chang Ma, Qinyu Chen, Zhihong Deng**

Department of Computer Science

Peking University

{zhaohaiteng, changma, chenqinyu, zhdeng}@pku.edu.cn

## ABSTRACT

Unsupervised domain adaptation (UDA), which is an important topic in transfer learning, aims to predict unlabeled data from target domain with access to labeled data from the source domain. In this work, we propose a novel framework called SIDA (Surrogate Mutual Information Maximization Domain Adaptation) with strong theoretical guarantees. To be specific, SIDA implements adaptation by maximizing mutual information (MI) between features. In the framework, a surrogate joint distribution models the underlying joint distribution of the unlabeled target domain. Our theoretical analysis validates SIDA by bounding the expected risk on target domain with MI and surrogate distribution bias. Experiments show that our approach is comparable with state-of-the-art unsupervised adaptation methods on standard UDA tasks.

## 1 INTRODUCTION

Inspired by human beings' ability to transfer knowledge across domains and tasks, transfer learning is proposed to leverage knowledge from source domain and task to improve performance on target domain and task (Pan and Yang, 2009). However, in practice, labeled data are often available on source domain, but limited on target domains. To address such situation, unsupervised domain adaptation (UDA), a category of transfer learning methods (Long et al., 2015; Tzeng et al., 2014; Long et al., 2016; 2017a; Ganin et al., 2016; Tzeng et al., 2017), attempts to enhance knowledge transfer from labeled source domain to target domain by leveraging unlabeled target domain data.

Most previous work is based on the data shift assumption, *i.e.*, the label space maintains the same across domains, but the data distribution conditioned on labels varies. Domain alignment and class-level method are two common practices for this situation. Domain alignment minimizes the discrepancy between the feature distributions of two domains to cope with data shift (Chen et al., 2020a; Zhang et al., 2019; Tang and Jia, 2020a; Wen et al., 2019; Cui et al., 2020). In recent years, class-level approaches based on pseudo-labels have emerged (Xie et al., 2018; Long et al., 2018; Kang et al., 2019; Gu et al., 2020; Xu et al., 2020; Liang et al., 2020; Venkat et al., 2020; Li et al., 2020a). Conditional alignment has achieved promising results by aligning conditional distributions from both domains (Xie et al., 2018; Long et al., 2018). However, the conditional distributions from different categories tend to mix together, leading to performance drop. Contrastive learning based methods resolve this issue by discriminating features from different classes (Kang et al., 2019; Chen et al., 2020a; Luo et al., 2020a), but still face the problem of pseudo-label precision.

Despite the success of class-level methods, most of them lack solid theoretical guidance. Prior theoretical results mainly focus on marginal distribution between domains (Ben-David et al., 2007; Redko et al., 2020). Other works (Chen et al., 2019a; Xie et al., 2018; Chen et al., 2019b) yield some intuitive explanations for conditional alignment and contrastive learning, but the relation between their objective functions and their theoretical analysis on cross-domain error remains unclear.

In this work, we propose Surrogate Information Domain Adaptation (SIDA), a general domain adaptation framework with strong theoretical guarantees. SIDA achieves adaptation by maximizing the mutual information (MI) between features within the same class, and ensuring the discriminability of features between different classes. Furthermore, a surrogate distribution is constructed to approximate the unlabeled target distribution, which improves flexibility for selecting data and assists MI

estimation. Also, our theoretical analyses directly establish a bound between MI of features and target expected risk, giving a proof that our model can improve generalization across domain.

Our novelties and contributions are summarized as follows:

- We propose a novel framework to achieve domain adaptation by maximizing surrogate MI.
- We establish an expected risk upper bound based on feature MI and surrogate distribution bias for UDA. This provides theoretical guarantee for our framework.
- Experiment results on three challenging benchmarks demonstrate that our method performs favorably against state-of-art class-level UDA models.

## 2 RELATED WORK

**Domain Adaptation** Prior works are based on two major assumptions: (1) the label shift hypothesis, where the label distribution changes, and (2) a more common data shift hypothesis where we only study the shift in conditional distribution under the premise that the label distribution is fixed. Our work focuses on the data shift hypothesis, and previous work following this line can be divided into two major categories: domain alignment methods which align marginal distributions, and class-level methods addressing the alignment of conditional distributions.

Domain alignment methods minimize the difference between feature distributions of source and target domains with various metrics, e.g. maximum mean discrepancy (MMD) (Long et al., 2015), JS divergence (Ganin et al., 2016) estimated by adversarial discriminator (Goodfellow et al., 2014), Wasserstein metric (Courty et al., 2016) and others. Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) has been applied to measure the discrepancy in marginal distributions (Tzeng et al., 2014; Long et al., 2015; 2016; 2017a; Sun and Saenko, 2016; Chen et al., 2020a). Adversarial domain adaptation plays a mini-max game to learn domain-invariant features (Ganin et al., 2016; Tzeng et al., 2017; Luo et al., 2020b; Wu and Guo, 2020; Li et al., 2020a).

Class-level methods align the conditional distribution based on pseudo-labels(Chen et al., 2020a; Luo et al., 2020a; Li et al., 2020b; Jiang et al., 2020; Hu et al., 2020; Liang et al., 2020; Venkat et al., 2020). Conditional alignment methods (Xie et al., 2018; Long et al., 2018) minimize the discrepancy between conditional distributions. In class-level methods, conditional distributions are assigned by pseudo-labels. The accuracy of pseudo-labels greatly influences performance and later works construct more accurate pseudo-labels (Gu et al., 2020; Wang and Breckon, 2020; Chen et al., 2020b). However, the major problem with this method is that error in conditional alignment leads to distribution overlap of features from different class, resulting in low discriminability on target domain. Contrastive learning addresses this problem by maximizing the discrepancy between different classes (Kang et al., 2019; Chen et al., 2020a; Luo et al., 2020a). However, the performance of contrastive learning also relies on pseudo-labeling.

In addition, previous class-level works provide weak theoretical support on cross-domain generalization. Prior works mainly focus on domain alignment (Ben-David et al., 2007; Redko et al., 2020). Chen et al. (2019a), Xie et al. (2018), and Chen et al. (2019b) consider optimal classification on both domains, which yields some intuitive explanation of conditional alignment and contrastive learning, but the relation between their objective function and theoretical cross-domain error remains unclear.

To emphasize these problems, we establish a novel framework based on MI of representations, and use theoretical results to explain the effect of MI on cross-domain generalization. MI emphasizes discriminability of features between different classes and guarantees the performance. In addition, surrogate distribution is constructed instead of pseudo-labels to provide better estimation of MI.

**Information Maximization Principle** Recently, mutual information maximization (InfoMax) for representation learning has attracted lots of attention (Chen et al., 2020c; Oord et al., 2018; Kolesnikov et al., 2019; Henaff, 2020; Tian et al., 2020; Hjelm et al., 2018; Bachman et al., 2019; Khosla et al., 2020). The intuition is that two features belonging to different classes should be discriminable while features of the same class should be similar. The InfoMax principle provides a general framework for learning informative representations, and provides consistent boosts in various downstream tasks.

We facilitate domain adaptation with MI maximization, i.e. maximizing the MI between features of the same class. Some works solve domain adaptation problem via information theoretical methods (Thota and Leontidis, 2021; Chen and Liu, 2020; Park et al., 2020), which maximize MI using InfoNCE estimation (Oord et al., 2018). As far as we know, we are the first to provide theoretical guarantee for the target domain expected risk based on MI. Compared with InfoNCE, the variational lower bound of MI we use is tighter (Poole et al., 2019). We also construct a surrogate distribution as a substitute for unlabeled target domain, which is more suitable for MI estimation.

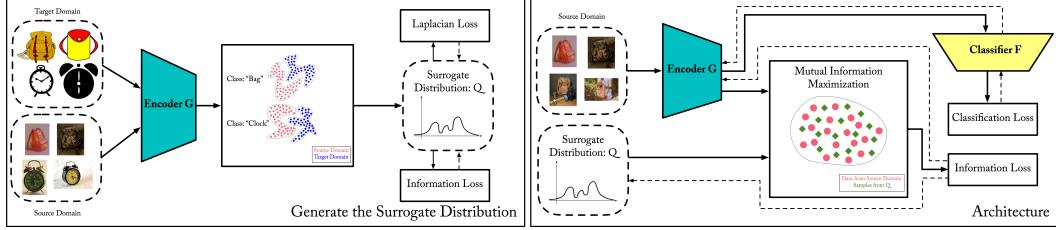


Figure 1: Overview of SIDA framework for training. Only encoder and classifier are involved in inference. The dashed arrow shows the path of the gradient backpropagation.

### 3 PRELIMINARIES

#### 3.1 NOTATIONS AND PROBLEM SETTING

Let  $\mathcal{X}$  be the data space and  $\mathcal{Y}$  be the label space. In UDA, there is a source distribution  $P_S(X, Y)$  and a target distribution  $P_T(X, Y)$  on  $\mathcal{X} \times \mathcal{Y}$ . Note that distributions are also referred to as domains in UDA. Our work is based on the data shift hypothesis, which assumes  $P_S(X, Y)$  and  $P_T(X, Y)$  satisfy the following properties:  $P_T(Y) = P_S(Y)$  and  $P_T(X|Y) \neq P_S(X|Y)$ .

In our work, we focus on classification tasks. Under this setting, an algorithm has access to  $n_S$  labeled samples  $\{(x_S^i, y_S^i)\}_{i=1}^{n_S} \sim P_S(X, Y)$  and  $n_T$  unlabeled samples  $\{(x_T^i)\}_{i=1}^{n_T} \sim P_T(X)$ , and outputs a hypothesis composed of an encoder  $G$  and a classifier  $F$ . Let  $\mathcal{Z}$  be the feature space. The encoder maps data to feature space, denoted by  $G : \mathcal{X} \rightarrow \mathcal{Z}$ . Then the classifier maps the feature to a corresponding class,  $F : \mathcal{Z} \rightarrow \mathcal{Y}$ .

For brevity, given encoder  $G$  and data-label distribution  $P(X, Y)$ , denote the distribution of  $G$ -encoded feature and label by  $P_G^G$ , i.e.  $P_G^G(z, y) = P(x = G^{-1}(z), y)$ .

Let  $F$  be a hypothesis, and  $P$  be the distribution of feature and label. The expected risk of a  $F$  w.r.t.  $P$  is denoted as

$$\epsilon_P(F) \triangleq \mathbb{E}_{P(z)} |\delta_{F(z)} - P(y|z)|_1, \quad (1)$$

where  $\delta_{F(z)}(y)$  equals to 1 if  $y = F(z)$  and equals 0 in else cases. Our objective is to minimize the expected risk of  $F$  on target feature distribution encoded by  $G$ ,

$$\min_{G, F} \epsilon_{P_T^G}(F). \quad (2)$$

### 4 METHODOLOGY

#### 4.1 OVERVIEW

In UDA task, the model needs to generalize across different domains with varying distributions; thus the encoder needs to extract appropriate features that are transferable across domains. The challenges of class-level adaptation are two folds: learning transferable features, and modeling  $P_T^G(Z|Y)$  without label information.

To solve the first problem, we use MI based methods. Following the InfoMax principle, we maximize the mutual information between features from the same class on the target and source mixture distribution. This encourages resemblance of features from the same class and reduces confusion of features from different categories, and thus provides opportunities for transferring classifier across domains.

As for the second challenge, we first revisit the data shift hypothesis. The distribution of labels  $P(Y)$  remains independent of domains, therefore the key is to model the conditional distribution  $P(Z|Y)$  on the target domain. However, modeling  $P(Z|Y)$  is intractable, since labels on the target domain are inaccessible. To tackle this problem, we model a surrogate distribution  $Q(Z|Y)$  instead.

We introduce the goal of maximizing MI in section 4.2, and theoretically explain how MI affects domain adaptation risk. In Section 4.3, we will introduce the model in detail, including the variational estimation of MI, the modeling of the surrogate distribution, and the optimization of the loss function of the model.

#### 4.2 MUTUAL INFORMATION MAXIMIZATION

MI measures the degree to which two variables can predict each other. Inspired by InfoMax principle (Hjelm et al., 2018), we maximize the MI between the features within the same class. It encourages features from different classes to be discriminable from each other.

We maximize MI between features on both source and target domain, regardless of which domain they come from. So we introduce mixture distribution  $S + T$  of both domain, which is

$$P_{S+T}(x, y) \triangleq \frac{1}{2}(P_S(x, y) + P_T(x, y)). \quad (3)$$

Note that because  $P_S(y) = P_T(y) = P_{S+T}(y)$ ,  $P_{S+T}(x|y) = \frac{1}{2}P_S(x | y) + \frac{1}{2}P_T(x | y)$ . Define the distribution of features from the same class as

$$P_{S+T}^G(z_1, z_2|y) \triangleq P_{S+T}^G(z_1|y)P_{S+T}^G(z_2|y), \quad P_{S+T}^G(z_1, z_2) = \sum_y P_{S+T}^G(y)P_{S+T}^G(z_1, z_2|y). \quad (4)$$

which means the feature  $z_1$  and  $z_2$  are sampled independently from the conditional distribution of the same class, with equal probability from source domain and target domain.

MI between features is maximized within the mixture distribution, as formalized below:

$$\arg \max_G I_{S+T}^G(Z_1; Z_2) = \int P_{S+T}^G(z_1, z_2) \log \frac{P_{S+T}^G(z_1, z_2)}{P_{S+T}^G(z_1)P_{S+T}^G(z_2)} dz_1 dz_2. \quad (5)$$

However, due to the lack of target domain labels,  $P_{S+T}^G$  is hard to model and thereby it is infeasible to estimate  $I_{S+T}^G$  directly. To address this problem, we propose a surrogate joint distribution  $Q(Z, Y)$  as the substitute for target domain  $P_T^G$ . Then the mixture distribution becomes  $P_{S+Q}^G = \frac{1}{2}(P_S^G + Q)$ , and the objective becomes maximizing  $I_{S+Q}^G(Z_1; Z_2)$ . The construction and optimization of the surrogate joint distribution is explained in Section 4.3.2.

##### 4.2.1 THEORETICAL MOTIVATION FOR MI MAXIMIZATION

We use theoretical bound to demonstrate the motivation for using MI maximization. Our theoretical results prove that minimizing the expected risk on the target domain can be naturally transformed into MI maximization and expected risk minimization on the source domain, which explains why MI maximization is pivotal to our framework. The proofs are in appendix.

**Definition 1** ( $\mathcal{H}\Delta\mathcal{H}$ -Divergence). *Let  $F_1 \in \mathcal{H}, F_2 \in \mathcal{H}$  be two hypotheses in hypothesis space  $\mathcal{H} : \mathcal{Z} \rightarrow \mathcal{Y}$ . Define  $\epsilon_P(F_1, F_2)$  as the disagreement between hypotheses  $F_1, F_2$  w.r.t. distribution  $P$  on  $\mathcal{Z}$ ,  $\epsilon_P(F_1, F_2) \triangleq \mathbb{E}_{z \sim P} |\delta_{F_1(z)} - \delta_{F_2(z)}|$ .  $\mathcal{H}\Delta\mathcal{H}$ -divergence, which is the discrepancy of two distributions  $P_1, P_2$  w.r.t. any hypothesis  $F_1 - F_2$  where  $F_1, F_2 \in \mathcal{H}$ , is defined as  $d_{\mathcal{H}\Delta\mathcal{H}}(P_1, P_2) \triangleq 2 \sup_{F_1, F_2 \in \mathcal{H}} |\epsilon_{P_1}(F_1, F_2) - \epsilon_{P_2}(F_1, F_2)|$ .*

**Theorem 4.1** (Bound of Target Domain expected risk). *The expected risk on target domain can be upper-bounded by the negative MI between features, and  $\mathcal{H}\Delta\mathcal{H}$ -divergence between features of two domains:*

$$\epsilon_{P_T^G}(F) \leq \epsilon_{P_S^G}(F) - 4I_{S+T}^G(Z_1; Z_2) + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) + 4H(Y). \quad (6)$$

The proof is in appendix. We give an explanation of the conditions for the upper bound to be equal.  $I_{S+T}^G(Z_1; Z_2)$  is a lower bound of  $I_{S+T}^G(Z; Y)$ , and it measures how much uncertainty of  $Y$  is reduced by knowing the feature, and it's equal to  $H(Y)$  if and only if  $P_{S+T}^G(Y|Z)$  is deterministic, i.e.,  $P_{S+T}^G(Y|Z)$  is  $\delta$  distribution, which means  $P_S^G(Y|Z) = P_T^G(Y|Z) = \delta_{Y(Z)}$ . Thus if the  $\mathcal{H}\Delta\mathcal{H}$ -divergence is zero, i.e.,  $P_S^G(Z) = P_T^G(Z)$ , then it's ensured that  $P_S^G(Z, Y) = P_T^G(Z, Y)$ , and  $\epsilon_{P_T^G}(F) = \epsilon_{P_S^G}(F)$ .

This upper bound decomposes the cross-domain generalization error into the divergence of feature marginal distribution and MI of features. It emphasizes that in addition to the divergence of the feature marginal distributions, only a MI term is enough for knowledge transfer across domains.

In this work, we minimize the expected risk on the source domain and maximize MI, for minimizing the upper bound of expected risk on target domain. Due to the lack of labels on target domain, we estimate MI based on surrogate distribution  $Q$ . The expected risk upper bound based on surrogate MI is further derived as follows.

**Definition 2** ( $L_1$ -distance). Define  $L_1$ -distance of  $P_1, P_2$  as  $d_1(P_1, P_2) \triangleq 2 \sup_{B \in \mathcal{B}} |\Pr_{P_1}[B] - \Pr_{P_2}[B]|$  where  $B$  is the set of measurable subsets under  $P_1$  and  $P_2$ .

**Theorem 4.2** (Bound Estimation with Surrogate Distribution). Let  $B \triangleq d_1(P_T^G(Z), Q(Z)) + \epsilon_{P_T^G}(Q(Y|Z))$  be the bias of surrogate distribution  $Q$  w.r.t target distribution. The expected risk on target domain can be upper-bounded by the negative surrogate MI between features,  $\mathcal{H}\Delta\mathcal{H}$ -divergence between source and target domain, and additional bias of surrogate domain:

$$\epsilon_{P_T^G}(F) \leq \epsilon_{P_S^G}(F) - 4I_{S+Q}^G(Z_1; Z_2) + B + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) + 4H(Y). \quad (7)$$

The proof is in appendix. This theorem supports the feasibility of domain adaptation via maximizing surrogate MI  $I_{S+Q}^G(Z_1; Z_2)$ . The bias of surrogate distribution is expressed in terms  $d_1(P_T^G(Z), Q(Z)) + \epsilon_{P_T^G}(Q(Y|Z))$ , where the first term is the distance between the surrogate and target feature marginal distribution, and the second term is the risk of conditional label surrogate distribution. To minimize the upper bound, the bias of the surrogate distribution should be small.

Bias equal to zero if and only if surrogate feature distribution and conditional label distribution are the same as target distribution, i.e.,  $P_T^G = Q$ , where surrogate distribution does not introduce errors.

### 4.3 SIDA FRAMEWORK

We employ MI maximization and surrogate distribution in our SIDA framework, as shown in Figure 1. During training, a surrogate distribution is first built from target and source data via optimizing w.r.t. Laplacian and MI. Then a mixture data distribution is created by encoding source data to features and sampling target features from the surrogate distribution. The encoder is optimized by maximizing MI, and minimizing classification error. The overall loss is:

$$L_{model} = L_{Classify} + \alpha_1 L_{MI} + \alpha_2 L_{Auxiliary} + L_{Laplacian}. \quad (8)$$

We elaborate each module in the following sections, and introduce the optimization of surrogate distribution in the last sections.

#### 4.3.1 MUTUAL INFORMATION ESTIMATION

Several MI estimation and optimization methods are proposed in deep learning (Poole et al., 2019). In this work, we use the following variational lower bound of MI as proposed in (Nguyen et al., 2010):

$$I(Z_1; Z_2) \geq \mathbb{E}_{P(z_1, z_2)}[f(z_1, z_2)] - e^{-1} \mathbb{E}_{P(z_1)}[\mathbb{E}_{P(z_2)}[e^{f(z_1, z_2)}]], \quad (9)$$

where  $f$  is a score function in  $\mathcal{Z} \times \mathcal{Z} \rightarrow R$ . The equality holds when  $\frac{e^{f(z_1, z_2)}}{\mathbb{E}_{P(z_1)} e^{f(z_1, z_2)}} = \frac{P(z_1|z_2)}{P(z_1)}$  and  $\mathbb{E}_{P(z_1)} \mathbb{E}_{P(z_2)} e^{f(z_1, z_2)} = e$ . The proof is in appendix. Therefore maximizing MI can be transformed into maximizing its lower bound, and the loss is:

$$L_{MI} = -\mathbb{E}_{P_{S+Q}^G(y)} \mathbb{E}_{P_{S+Q}^G(z_1|y)} \mathbb{E}_{P_{S+Q}^G(z_2|y)} [f(z_1, z_2)] + e^{-1} \mathbb{E}_{P_{S+Q}^G(z_1)} [\mathbb{E}_{P_{S+Q}^G(z_2)} [e^{f(z_1, z_2)}]], \quad (10)$$

where  $f(z_1, z_2)$  is constructed as  $T_{m_1}^{m_2}(|z_1 - z_2|_2)$ .  $T_{m_1}^{m_2}$  is a threshold function, i.e.,  $T_{m_1}^{m_2}(a) = \max(m_1, \min(m_2, a))$ .

#### 4.3.2 SURROGATE DISTRIBUTION CONSTRUCTION

We decompose the surrogate distribution  $Q(Z, Y)$  into two factors  $Q(Z, Y) = Q(Y)Q(Z|Y)$ , and describe the construction of two factors individually.

According to the data shift assumption,  $P_T(Y)$  is similar to  $P_S(Y)$ , thus  $Q(Y)$  should be similar to  $P_S(Y)$ . However, source distribution may suffer from the class imbalance problem, which will harm the performance on classes with fewer data. A common solution to this problem is class-balanced sampling, which samples data on each class uniformly. In this work, for the balance across different classes, the marginal distribution  $P_S(Y)$  and  $Q(Y)$  are both considered as uniform distribution.

As for the second term, the conditional surrogate distribution  $Q(Z|Y)$  is constructed by weighted sampling method (Junsomboon and Phientrakul, 2017). We need to construct the  $Q(Z|Y)$  to calculate Eq. 10, which takes the form of expectation, and only needs samples from  $Q(Z|Y)$  to estimate. Instead of explicitly modeling  $Q(Y|Z)$ , we use the ideas of importance sampling. For each class, the surrogate conditional distribution  $Q(Z|y_j)$  is constructed by weighted sampling from target features. Thus  $Q(Z|Y)$  is a distribution on target features  $\{G(x_T^i)\}_{i=1}^{n_T}$ , and parameterized by  $W \in R^{n_T \times n_Y}$ , where  $n_Y$  is the number of labels:

$$Q(G(x_T^i)|y_j) = W_{ij}, \text{ s.t. } W_{ij} \in [0, 1], \sum_i W_{ij} = 1, \forall j. \quad (11)$$

Compared with Pseudo-labeling, our estimation method has the following advantages: (1) The surrogate marginal distribution of feature  $Q(Z) = \sum_Y Q(Z|Y)P(Y)$  is not fixed, which enables us to select features more flexibly. (2) The construction process of the surrogate distribution makes MI estimation  $I(Z_1, Z_2)$  more convenient. Our surrogate distribution  $Q(Z|Y)$  provides weights so that weighted sampling can be performed directly.

The challenge is to optimize the sampling probability weights  $W_{ij}$  so as to minimize the bias of the surrogate distribution. We propose to optimize this distribution via Laplacian regularization as well as MI, which is explained in details in the following section.

#### 4.3.3 SURROGATE DISTRIBUTION LOSS

Inspired by semi-supervised learning (Zhu and Ghahramani, 2002), we expect that the surrogate distribution is consistent with the clustering structure of the feature distribution, based on the assumption that the feature is well-structured and clustered according to class, regardless of domains. We employ Laplacian regularization to capture the manifold clustering structure of feature distribution.

Let  $A \in R^{n_T \times n_T}$  be the adjacent matrix of target features, where the entry  $A_{ij}$  measures how similar  $G(x_T^i)$  and  $G(x_T^j)$  are, and  $D = \text{Diag}(A\mathbf{1})$  is the degree matrix, i.e.  $D_{ii} = \sum_j A_{ij}$  and  $D_{ij} = 0, \forall i \neq j$ . We construct A as K-nearest graph on target features, and the Laplacian regularization of  $W$  is defined as

$$L_{\text{Laplacian}} = \text{Tr}(W^T L W) = \frac{1}{2} \sum_k \sum_{i,j} A_{ij} \left( \frac{W_{ik}}{D_{ii}} - \frac{W_{jk}}{D_{jj}} \right)^2, \quad (12)$$

where L is the normalized Laplacian matrix  $L = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ . This regularization encourages  $W_{ik}$  and  $W_{jk}$  to be similar if feature  $G(x_T^i)$  is similar to  $G(x_T^j)$ . It also enables the conditional surrogate distribution to spread uniformly on a connected region.

#### 4.3.4 CLASSIFICATION AND AUXILIARY LOSS

The model is optimized in supervised manner on the source domain. The classification loss is the standard cross-entropy loss via class-balanced sampling.

$$L_{\text{Classify}} = -\frac{1}{n_Y} \sum_y E_{P_S(x|y)} \log P(F(G(x)) = y). \quad (13)$$

And we use auxiliary classification loss on pseudo-labels from the surrogate distribution, as the classifier will benefit from label information of the surrogate distribution. We use mean square error (MSE) for pseudo-labels, which is more robust to noise than cross entropy loss.

$$L_{Auxiliary} = \frac{1}{n_Y} \sum_y E_{Q(x|y)}(1 - P(F(G(x)) = y))^2. \quad (14)$$

#### 4.3.5 OPTIMIZATION OF SURROGATE DISTRIBUTION

We optimize both  $L_{\text{Laplacian}}$  and  $L_{\text{MI}}$  w.r.t.  $W$  for a structured and informative surrogate distribution. At the beginning of each epoch,  $W$  is initialized by K-means clustering and filtered by the distance to the clustering centers, i.e.  $\widetilde{W}_{i,j} = \mathbf{1}_{\mu_j \text{ nearest to } G(x_i)} \mathbf{1}_{d(G(x_i), \mu_j) < \theta}$ , where  $\mu_j$  is the  $j$ -th clustering center during clustering, and normalized as  $W_{i,j} = \frac{\widetilde{W}_{i,j}}{\sum_i \widetilde{W}_{i,j}}$ .

To minimize two losses w.r.t  $W$ , the gradients are derived analytically. The derivation is in appendix.

Based on the gradient of these two losses, we perform T-step descent update of  $W$  with learning rate  $\eta_1$  and  $\eta_2$  respectively, and each step we project  $W$  back to the probability simplex (Wang and Carreira-Perpinán, 2013). See appendix for details.

## 5 EXPERIMENTS

In this section, We evaluate the proposed method on three public domain adaptation benchmarks, compared with recent state-of-the-art UDA methods. We conduct extensive ablation study to discuss our method.

### 5.1 DATASETS

Office-31 (Saenko et al., 2010) is a commonly used dataset for UDA, where images are collected from three distinct domains: Amazon (A), Webcam (W) and DSLR (D). The dataset consists of 4,110 images belonging to 31 classes, and is imbalanced across domains, with 2,817 images in A domain, 795 images in W domain, and 498 images in D domain. Our method is evaluated on all six transfer tasks. We follow the standard protocol for UDA (Ganin and Lempitsky, 2015; Long et al., 2017a) to use all labeled source samples and all unlabeled target samples as the training data.

Office-Home (Venkateswara et al., 2017) is another classical dataset with 15,500 images of 65 categories in office and home settings, consisting of 4 domains including Artistic images (Ar), Clip Art images (Cl), Product images (Pr) and Real-World images (Rw). Following the common protocol, all 65 categories from the four domains are used for evaluation of UDA, forming 12 transfer tasks.

VisDA-2017 (Peng et al., 2017) is a challenging benchmark for UDA with the domain shift from synthetic data to real imagery. It contains 152,397 training images and 55,388 validation images across 12 classes. Following the training and testing protocol in (Long et al., 2017b), the model is trained on labeled training and unlabeled validation set and tested on the validation set.

### 5.2 IMPLEMENTATION DETAILS

For each transfer task, mean ( $\pm$ std) over 5 runs of the test accuracy are reported. We use the ImageNet (Deng et al., 2009) pre-trained ResNet-50 (He et al., 2016) without final classifier layer as the encoder network  $G$  for Office-31 and Office-Home, and ResNet-101 for VisDA-2017. The details of experiments are in appendix. Code is attached in supplementary materials.

### 5.3 BASELINES

We compare our approach with the state of the arts. Domain alignment methods include DAN (Long et al., 2015), DANN (Ganin et al., 2016), JAN (Long et al., 2017a). Class-level methods include conditional alignment methods (CDAN (Long et al., 2018), DCAN (Li et al., 2020b)), ALDA (Chen et al., 2020b)), and contrastive methods (DRMEA (Luo et al., 2020a), ETD (Li et al., 2020c), DADA (Tang and Jia, 2020b), SAFN Xu et al. (2019)). We only report available results in each baseline.

Table 1: Accuracy(%) on Office-31

Methods	A→W	D→W	W→D	A→D	D→A	W→A	avg
ResNet-50	68.4±0.2	96.7±0.1	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.1
DAN	80.5±0.4	97.1±0.2	99.6±0.1	78.6±0.2	63.6±0.3	62.8±0.2	80.4
DANN	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	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
CDAN	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
DCAN	95.0	97.5	100.0	92.6	77.2	74.9	89.5
ALDA	95.6±0.5	97.7±0.1	100.0±0.0	94.0±0.4	72.2±0.4	72.5±0.2	88.7
ETD	92.1	100.0	100.0	88.0	71.0	67.8	86.2
DADA	92.3±0.1	99.2±0.1	100.0±0.0	93.9±0.2	74.4±0.1	74.2±0.1	89.0
SAFN	90.3	98.7	100.0	92.1	73.4	71.2	87.6
SIDA (ours)	94.5±0.6	99.2±0.1	100.0±0.0	95.7±0.3	76.6±0.6	76.2±0.4	<b>90.4</b>

Table 2: Accuracy (%) on Office-Home

Methods	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
ResNet-50	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	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	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	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
CDAN	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
DCAN	54.5	75.7	81.2	67.4	74.0	76.3	67.4	52.7	80.6	74.1	59.1	83.5	70.5
ALDA	53.7	70.1	76.4	60.2	72.6	71.5	56.8	51.9	77.1	70.2	56.3	82.1	66.6
DRMEA	52.3	73.0	77.3	64.3	72.0	71.8	63.6	52.7	78.5	72.0	57.7	81.6	68.1
ETD	51.3	71.9	85.7	57.6	69.2	73.7	57.8	51.2	79.3	70.2	57.5	82.1	67.3
SAFN	54.4	73.3	77.9	65.2	71.5	73.2	63.6	52.6	78.2	72.3	58.0	82.1	68.5
SIDA(ours)	57.2	79.1	81.7	67.1	74.5	77.3	67.2	53.9	82.5	71.4	58.7	83.3	<b>71.2</b>

## 5.4 RESULTS AND COMPARATIVE ANALYSIS

In this section we will present our results and compare with other methods for evaluation on three standard benchmarks mentioned earlier. We report average classification accuracies with standard deviations. Results of other methods are collected from original papers or the follow-up work.

**Office-31** The unsupervised adaptation results on six Office-31 transfer tasks based on ResNet-50 are reported in Table 1. As the data reveals, the average accuracy of SIDA is 90.4, the best among all compared methods. It is noteworthy that our proposed method substantially improves the classification accuracy on hard transfer tasks, e.g. W→A, A→D, and D→A, where source and target data are similar. Our model also achieves comparable classification performance on easy transfer tasks, e.g. D→W, W→D, and A→W. Our improvements are mainly on hard settings.

**Office-Home** Results on Office-Home using ResNet-50 backbone are reported in Table 2. It can be observed that SIDA exceeds all compared methods on most transfer tasks with an average accuracy of 71.2. The performance reveals the importance of maximizing MI between feature in difficult domain-adaptation tasks which contain more categories.

**VisDA-2017** Table 3 summarizes our experimental results on the more challenging VisDA-2017 dataset. For fair comparison, all methods listed here use ResNet-101 as the backbone network. Note that SIDA still outperforms state-of-the-art models with an average accuracy of 84.0, surpassing the previous best result reported by +4%.

In summary, our surrogate MI maximization approach achieves competitive performance compared to traditional alignment based methods and recent pseudo-label based methods for UDA. It underlines the validity of using information theory methods for UDA via MI maximization.

## 5.5 VISUALIZATION

Our visualization experiment is carried out on  $D \rightarrow A$  and  $W \rightarrow A$  tasks in the dataset Office-31, which are the two most difficult tasks in Office-31. The baselines we chose were ResNet-50 pre-trained on ImageNet and a typical conditional alignment method, CDAN (Long et al., 2018). We

Table 3: Accuracy (%) on VisDA-2017

Methods	Plane	Bcycl	Bus	Car	Horse	Knife	Mcycle	Person	Plant	Sktbrd	Train	Truck	Avg
ResNet-101	55.1	53.3	61.9	59.1	80.6	17.9	79.7	31.2	81.0	26.5	73.5	8.5	52.4
DAN	87.1	63.0	76.5	42.0	90.3	42.9	85.9	53.1	49.7	36.3	85.8	20.7	61.1
DANN	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
CDAN	85.2	66.9	83.0	50.8	84.2	74.9	88.1	74.5	83.4	76.0	81.9	38.0	73.9
ALDA	93.8	74.1	82.4	69.4	90.6	87.2	89.0	67.6	93.4	76.1	87.7	22.2	77.8
DRMEA	92.1	75.0	78.9	75.5	91.2	81.9	89.0	77.2	93.3	77.4	84.8	35.1	79.3
DADA	92.9	74.2	82.5	65.0	90.9	93.8	87.2	74.2	89.9	71.5	86.5	48.7	79.8
SAFN	93.6	61.3	84.1	70.6	94.1	79.0	91.8	79.6	89.9	55.6	89.0	24.4	76.1
SIDA(ours)	95.4	83.1	77.1	64.6	94.5	97.2	88.7	78.4	93.8	89.9	85.2	59.4	<b>84.0</b>

Table 5: Ablation Study

MI	SDO	A→W	A→D	D→A	W→A	Avg
×	×	$90.25 \pm 0.2$	$92.37 \pm 0.1$	$74.21 \pm 0.2$	$74.09 \pm 0.1$	82.7
×	✓	$92.08 \pm 0.3$	$94.28 \pm 0.3$	$74.23 \pm 0.9$	$74.74 \pm 0.8$	83.8
✓	×	$94.03 \pm 0.1$	$95.28 \pm 0.1$	$75.86 \pm 0.4$	$75.72 \pm 0.5$	85.2
✓	✓	$94.52 \pm 0.6$	$95.68 \pm 0.1$	$76.62 \pm 0.6$	$76.22 \pm 0.4$	85.8

take representation before the final linear classification layer as feature vectors. We use t-SNE to visualize the features. Details and more results are in appendix.

Figure 2 shows that SIDA can make the features of different categories more distinguishable, which is the consequence of maximizing MI among features from the same category. Discriminable features can be easier for classification, as the visualization shows.

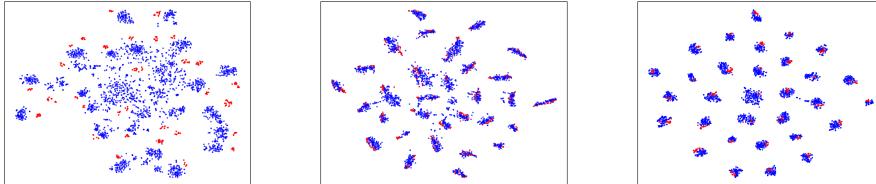


Figure 2: From left to right are the feature visualizations on task D→A of ResNet50, CDAN, and SIDA, respectively. Red represents source domain, and blue represents the target domain.

## 5.6 ABLATION STUDY

In this section, to evaluate how different components of our work contribute to the final performance, we conduct ablation study for SIDA on Office-31. We mainly focus on harder transfer tasks, e.g. A→W, A→D, D→A and W→A. We investigate different combinations of two components: MI maximization and surrogate distribution optimization(SDO). The average classification accuracy on four tasks are in Table 5.

From the results, we can observe that the model with MI maximization outperforms the base model without the two components by about 2.5% on average, which demonstrates the effectiveness of the maximization strategy. The surrogate distribution optimization also improves the average performance by 1.1% compared to base model, confirming that the optimization improves the quality of surrogate distribution. The combination of two components yields the highest improvement.

## 6 CONCLUSION AND FUTURE WORK

In this work, we introduce a novel framework of unsupervised domain adaptation and provide theoretical analysis to validate our optimization objectives. Experiments show that our approach gives competitive results compared to state-of-the-art unsupervised adaptation methods on standard domain adaptation tasks. One unresolved problem is to integrate the domain discrepancy in target risk upper bound into mutual information framework. This problem is left for future work.

## 7 ACKNOWLEDGEMENT

### REFERENCES

- Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.
- Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015.
- Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.
- Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *NIPS*, 2016.
- Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *International conference on machine learning*, pages 2208–2217. PMLR, 2017a.
- Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.
- Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.
- Chao Chen, Zhihang Fu, Zhihong Chen, Sheng Jin, Zhaowei Cheng, Xinyu Jin, and Xian-Sheng Hua. Homm: Higher-order moment matching for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3422–3429, 2020a.
- Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413. PMLR, 2019.
- Hui Tang and Kui Jia. Discriminative adversarial domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5940–5947, 2020a.
- Jun Wen, Risheng Liu, Nenggan Zheng, Qian Zheng, Zhefeng Gong, and Junsong Yuan. Exploiting local feature patterns for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5401–5408, 2019.
- Shuhao Cui, Shuhui Wang, Junbao Zhuo, Chi Su, Qingming Huang, and Qi Tian. Gradually vanishing bridge for adversarial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12455–12464, 2020.
- Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *International conference on machine learning*, pages 5423–5432. PMLR, 2018.
- Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, 2018.
- Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4893–4902, 2019.
- Xiang Gu, Jian Sun, and Zongben Xu. Spherical space domain adaptation with robust pseudo-label loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9101–9110, 2020.

- Renjun Xu, Pelen Liu, Liyan Wang, Chao Chen, and Jindong Wang. Reliable weighted optimal transport for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4394–4403, 2020.
- Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020.
- Naveen Venkat, Jogendra Nath Kundu, Durgesh Singh, Ambareesh Revanur, and Venkatesh Babu R. Your classifier can secretly suffice multi-source domain adaptation. In *Advances in Neural Information Processing Systems*, volume 33, pages 4647–4659, 2020.
- Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9641–9650, 2020a.
- You-Wei Luo, Chuan-Xian Ren, Pengfei Ge, Ke-Kun Huang, and Yu-Feng Yu. Unsupervised domain adaptation via discriminative manifold embedding and alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5029–5036, 2020a.
- Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19:137, 2007.
- Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. A survey on domain adaptation theory: learning bounds and theoretical guarantees. *arXiv e-prints*, pages arXiv–2004, 2020.
- Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 627–636, 2019a.
- Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *International conference on machine learning*, pages 1081–1090. PMLR, 2019b.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.
- Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.
- Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.
- Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. Adversarial style mining for one-shot unsupervised domain adaptation. *Advances in Neural Information Processing Systems*, 33, 2020b.
- Yuan Wu and Yuhong Guo. Dual adversarial co-learning for multi-domain text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6438–6445, 2020.
- Shuang Li, Chi Liu, Qiuxia Lin, Binhu Xie, Zhengming Ding, Gao Huang, and Jian Tang. Domain conditioned adaptation network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11386–11393, 2020b.
- Xiang Jiang, Qicheng Lao, Stan Matwin, and Mohammad Havaei. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 4816–4827. PMLR, 2020.

- Lanqing Hu, Meina Kan, Shiguang Shan, and Xilin Chen. Unsupervised domain adaptation with hierarchical gradient synchronization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4043–4052, 2020.
- Qian Wang and Toby Breckon. Unsupervised domain adaptation via structured prediction based selective pseudo-labeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6243–6250, 2020.
- Minghao Chen, Shuai Zhao, Haifeng Liu, and Deng Cai. Adversarial learned loss for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3521–3528, 2020b.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020c.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1920–1929, 2019.
- Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR, 2020.
- Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer, 2020.
- R. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2018.
- Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, volume 32, pages 15509–15519, 2019.
- Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673, 2020.
- Mamatha Thota and Georgios Leontidis. Contrastive domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2209–2218, 2021.
- Qingchao Chen and Yang Liu. Structure-aware feature fusion for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10567–10574, 2020.
- Changhwa Park, Jonghyun Lee, Jaeyoon Yoo, Minhoe Hur, and Sungroh Yoon. Joint contrastive learning for unsupervised domain adaptation. *arXiv preprint arXiv:2006.10297*, 2020.
- Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180. PMLR, 2019.
- XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.
- Nutthaporn Junsomboon and Tanasanee Phienthrakul. Combining over-sampling and under-sampling techniques for imbalance dataset. In *Proceedings of the 9th International Conference on Machine Learning and Computing, ICMLC 2017*, page 243–247, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450348171. doi: 10.1145/3055635.3056643. URL <https://doi.org/10.1145/3055635.3056643>.

- X. Zhu and Z. Ghahramani. Learning from labels and unlabeled data with label propagation. *Tech Report*, 3175(2004):237–244, 2002.
- Weiran Wang and Miguel A Carreira-Perpinán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *arXiv preprint arXiv:1309.1541*, 2013.
- Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.
- Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.
- Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017.
- Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017.
- Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *arXiv preprint arXiv:1705.10667*, 2017b.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- Mengxue Li, Yi-Ming Zhai, You-Wei Luo, Peng-Fei Ge, and Chuan-Xian Ren. Enhanced transport distance for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13936–13944, 2020c.
- H. Tang and K. Jia. Discriminative adversarial domain adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(4):5940–5947, 2020b.
- Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1426–1435, 2019.

## 8 APPENDIX

### 8.1 PROOF FOR THEOREM 4.1

**Theorem 8.1** (Bound of Target Domain expected risk). *The expected risk on target domain can be upper-bounded by the negative MI between features, and  $\mathcal{H}\Delta\mathcal{H}$ -divergence between features of two domains:*

$$\epsilon_{P_T^G}(F) \leq \epsilon_{P_S^G}(F) - 4I_{S+T}^G(Z_1; Z_2) + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) + 4H(Y) \quad (15)$$

*Proof.* The risk can be relaxed by triangle inequality

$$\begin{aligned} \epsilon_{P_T^G}(F) &\leq \epsilon_{P_T^G}(F') + \epsilon_{P_T^G}(F, F') \\ &= \epsilon_{P_T^G}(F') + \epsilon_{P_T^G}(F, F') + \epsilon_{P_S^G}(F, F') - \epsilon_{P_S^G}(F, F') \\ &\leq \epsilon_{P_T^G}(F') + \epsilon_{P_T^G}(F, F') + \epsilon_{P_S^G}(F) + \epsilon_{P_S^G}(F') - \epsilon_{P_S^G}(F, F') \\ &\leq \epsilon_{P_S^G}(F) + \epsilon_{P_T^G}(F') + \epsilon_{P_S^G}(F') + |\epsilon_{P_T^G}(F, F') - \epsilon_{P_S^G}(F, F')| \\ &\leq \epsilon_{P_S^G}(F) + \epsilon_{P_T^G}(F') + \epsilon_{P_S^G}(F') + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) \end{aligned} \quad (16)$$

For the term  $\epsilon_{P_T^G}(F') + \epsilon_{P_S^G}(F')$ , we have

$$\begin{aligned}
\epsilon_{P_T^G}(F') + \epsilon_{P_S^G}(F') &= \mathbb{E}_{P_S^G(z)} |P_S^G(y|z) - \delta_{F'(z)}|_1 + \mathbb{E}_{P_T^G(z)} |P_T^G(y|z) - \delta_{F'(z)}|_1 \\
&= 2\mathbb{E}_{P_S^G(z)} (1 - P_S^G(F'(z)|z)) + 2\mathbb{E}_{P_T^G(z)} (1 - P_T^G(F'(z)|z)) \\
&= 2(2 - \sum_z P_S^G(z)(P_S^G(F'(z)|z)) - \sum_z P_T^G(z)(P_T^G(F'(z)|z))) \\
&= 2(2 - \sum_z P_S^G(F'(z), z) - \sum_z P_T^G(F'(z), z)) \\
&= 2(2 - 2 \sum_z P_{S+T}^G(F'(z), z)) \\
&= 4(1 - \sum_z P_{S+T}^G(z)P_{S+T}^G(F'(z)|z)) \\
&= 4(\sum_z P_{S+T}^G(z)(1 - P_{S+T}^G(F'(z)|z))) \\
&= 4\mathbb{E}_{P_{S+T}^G(z)} (1 - P_{S+T}^G(F'(z)|z)) \\
&\leq -4\mathbb{E}_{P_{S+T}^G(z)} \log P_{S+T}^G(F'(z)|z)
\end{aligned} \tag{17}$$

While  $F'$  can be any classifier, define  $F'$  as the optimal classifier on  $P_{S+T}^G$ , i.e.  $F'(z) = \arg \max_y P_{S+T}^G(y|z)$ .

Recall the definition of MI,

$$\begin{aligned}
I_{S+T}^G(Z; Y) &= H_{S+T}(Y) - H_{S+T}(Y|Z) \\
&= H_{S+T}(Y) + \mathbb{E}_{P_{S+T}^G(z)} \mathbb{E}_{P_{S+T}^G(y|z)} \log P_{S+T}^G(y|z) \\
&\leq H_{S+T}(Y) + \mathbb{E}_{P_{S+T}^G(z)} \mathbb{E}_{P_{S+T}^G(y|z)} \log P_{S+T}^G(F'(z)|z) \\
&= H_{S+T}(Y) + \mathbb{E}_{P_{S+T}^G(z)} \log P_{S+T}^G(F'(z)|z)
\end{aligned} \tag{18}$$

Which means that

$$\begin{aligned}
\epsilon_{P_T^G}(F') + \epsilon_{P_S^G}(F') &\leq -4\mathbb{E}_{P_{S+T}^G(z)} \log P_{S+T}^G(F'(z)|z) \\
&\leq -4I_{S+T}^G(Z; Y) + 4H_{S+T}(Y)
\end{aligned} \tag{19}$$

According to the MI chain rule,  $I(Z_1; Y, Z_2) = I(Z_1; Y) + I(Z_1; Z_2|Y) = I(Z_1; Z_2) + I(Z_1; Y|Z_2)$ . Since  $Z_1$  and  $Z_2$  are two samples from class  $Y$ ,  $Z_1$  and  $Z_2$  are independent for a given  $Y$ , i.e.,  $I(Z_1; Z_2|Y) = 0$ . So we can get  $I(Z_1; Y) = I(Z_1; Z_2) + I(Z_1; Y|Z_2)$ . Because  $I(Z_1; Y|Z_2) \geq 0$ , we finally get

$$I(Z_1; Y) \geq I(Z_1; Z_2) \tag{20}$$

Note that  $I(Z_1; Y)$  is  $I(Z; Y)$ , because  $Z_1$  and  $Z_2$  both follow distribution  $P(Z|Y)$ .

So now we get the conclusion by

$$\epsilon_{P_T^G}(F') + \epsilon_{P_S^G}(F') \leq -4I_{S+T}^G(Z_1; Z_2) + 4H_{S+T}(Y) \tag{21}$$

We give an explanation of the conditions for the upper bound to be equal.  $I_{S+T}^G(Z_1; Z_2)$  is a lower bound of  $I_{S+T}^G(Z; Y)$ , and it measures how much uncertainty of  $Y$  is reduced by knowing the feature, and it's equal to  $H(Y)$  if and only if  $P_{S+T}^G(Y|Z)$  is deterministic, i.e.,  $P_{S+T}^G(Y|Z)$  is  $\delta$  distribution, which means  $P_S^G(Y|Z) = P_T^G(Y|Z) = \delta_{Y(Z)}$ . Thus if the  $\mathcal{H}\Delta\mathcal{H}$ -divergence is zero, i.e.,  $P_S^G(Z) = P_T^G(Z)$ , then it's ensured that  $P_S^G(Z, Y) = P_T^G(Z, Y)$ , and  $\epsilon_{P_T^G}(F) = \epsilon_{P_S^G}(F)$ .

□

## 8.2 PROOF FOR THEOREM 4.2

**Theorem 8.2** (Bound Estimation with Surrogate Distribution). *Let  $B = d_1(P_T^G(Z), Q(Z)) + \epsilon_{P_T^G}(Q(Y|Z))$  be the bias of surrogate distribution  $Q$  w.r.t target distribution. The expected risk on target domain can be upper-bounded by the negative surrogate MI between features,  $\mathcal{H}\Delta\mathcal{H}$ -divergence between source and target domain, and additional bias of surrogate domain:*

$$\epsilon_{P_T^G}(F) \leq \epsilon_{P_S^G}(F) - 4I_{S+Q}^G(Z_1; Z_2) + B + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) + 4H(Y) \quad (22)$$

*Proof.* The expected expected risk can be relaxed by triangle inequality:

$$\begin{aligned} \epsilon_{P_T^G}(F) &\leq \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_T^G}(F, Q(Y|Z)) \\ &\leq \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_T^G}(F', Q(Y|Z)) + \epsilon_{P_T^G}(F, F') \\ &= \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_T^G}(F', Q(Y|Z)) + \epsilon_{P_T^G}(F, F') + \epsilon_{P_S^G}(F, F') - \epsilon_{P_S^G}(F, F') \\ &\leq \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_T^G}(F', Q(Y|Z)) + \epsilon_{P_T^G}(F, F') + \epsilon_{P_S^G}(F) + \epsilon_{P_S^G}(F') - \epsilon_{P_S^G}(F, F') \\ &\leq \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_S^G}(F) + \epsilon_{P_T^G}(F', Q(Y|Z)) + \epsilon_{P_S^G}(F') + |\epsilon_{P_T^G}(F, F') - \epsilon_{P_S^G}(F, F')| \\ &\leq \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_S^G}(F) + \epsilon_{P_T^G}(F', Q(Y|Z)) + \epsilon_{P_S^G}(F') + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) \\ &= \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_S^G}(F) + \epsilon_{P_T^G}(F', Q(Y|Z)) - \epsilon_Q(F') + \\ &\quad \epsilon_Q(F') + \epsilon_{P_S^G}(F') + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) \\ &\leq \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_S^G}(F) + \int (P_T^G(z) - Q(z))|\delta_{F'(z)} - Q(Y|z)|_1 dz + \\ &\quad \epsilon_Q(F') + \epsilon_{P_S^G}(F') + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) \\ &\leq \epsilon_{P_T^G}(Q(Y|Z)) + \epsilon_{P_S^G}(F) + d_1(P_T^G(Z), Q(Z)) + \epsilon_Q(F') + \epsilon_{P_S^G}(F') + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) \\ &= \epsilon_{P_S^G}(F) + B + \epsilon_Q(F') + \epsilon_{P_S^G}(F') + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(P_S^G(Z), P_T^G(Z)) \end{aligned} \quad (23)$$

By the same method as the previous proof, terms  $\epsilon_Q(F') + \epsilon_{P_S^G}(F')$  can be deduced into MI  $-4I_{S+Q}^G(Z_1; Z_2) + 4H(Y)$ .

□

## 8.3 PROOF FOR THE EQUALITY CONDITION OF MI ESTIMATION

**Proposition 8.3.** *The following MI lower bound holds*

$$I(Z_1; Z_2) \geq \mathbb{E}_{P(z_1, z_2)}[f(z_1, z_2)] - e^{-1}\mathbb{E}_{P(z_1)}[\mathbb{E}_{P(z_2)}[e^{f(z_1, z_2)}]] \quad (24)$$

where  $f$  is arbitrary function in  $\mathcal{Z} \times \mathcal{Z} \rightarrow R$ . The equality holds when  $\frac{e^{f(z_1, z_2)}}{\mathbb{E}_{P(z_1)}e^{f(z_1, z_2)}} = \frac{P(z_1|z_2)}{P(z_1)}$  and  $\mathbb{E}_{P(z_1)}\mathbb{E}_{P(z_2)}e^{f(z_1, z_2)} = e$ .

*Proof.* The proof is as follows:

$$I(Z_1; Z_2) = \mathbb{E}_{P(z_1, z_2)} \left[ \log \frac{q(z_1 | z_2)}{P(z_1)} \right] + \mathbb{E}_{P(z_2)}[KL(P(z_1 | z_2) \| q(z_1 | z_2))] \geq \mathbb{E}_{P(z_1, z_2)} \left[ \log \frac{q(z_1 | z_2)}{P(z_1)} \right] \quad (25)$$

, where  $q$  is arbitrary variational distribution. Let  $q(z_1 \mid z_2) = \frac{P(z_1)}{Z(z_2)} e^{f(z_1, z_2)}$ , where  $Z(z_2) = \mathbb{E}_{P(z_1)} [e^{f(z_1, z_2)}]$  is the normalization constant.

Then

$$I(Z_1; Z_2) \geq \mathbb{E}_{P(z_1, z_2)}[f(z_1, z_2)] - \log \mathbb{E}_{P(z_2)}[Z(z_2)] \quad (26)$$

By  $\log(x) \leq \frac{x}{a(x)} + \log(a(x)) - 1$ , which is tight when  $a(x) = x$ ,

$$I(Z_1; Z_2) \geq \mathbb{E}_{P(z_1, z_2)}[f(z_1, z_2)] - \mathbb{E}_{P(z_2)} \left[ \frac{\mathbb{E}_{P(z_1)} [e^{f(z_1, z_2)}]}{a(z_2)} + \log(a(z_2)) - 1 \right] \quad (27)$$

Let  $a(z_2) = e$ , we get the final form of lower bound:

$$I(Z_1; Z_2) \geq \mathbb{E}_{P(z_1, z_2)}[f(z_1, z_2)] - e^{-1} \mathbb{E}_{P(z_2)} [\mathbb{E}_{P(z_1)} [e^{f(z_1, z_2)}]] \quad (28)$$

□

#### 8.4 DETAILS FOR SURROGATE DISTRIBUTION OPTIMIZATION

Let  $P$  be the conditional distribution matrix of source domain, i.e  $P_{ij} = P_S(x_S^i | y_S^j)$ , and  $W$  be the conditional distribution matrix of surrogate distribution  $Q$ . Let  $M = \frac{1}{2} \begin{bmatrix} P \\ W \end{bmatrix}$  be the conditional distribution matrix of mixture distribution  $P_{S+T}$ . Let  $S$  is the score function matrix, i.e.  $S_{i,j} = f(G(x^i), G(x^j))$ ,  $i, j = 1, \dots, n_S + n_T$ , where  $f$  is the score function of the MI lower bound. With class-balanced sampling,  $L_{MI}$  can be represented as follows:

$$L_{MI} = -\left(\frac{1}{n_Y} \text{Tr}(M^T S M) - \frac{1}{en_Y^2} \mathbf{1}^T M^T e^S M \mathbf{1}\right) \quad (29)$$

The gradient w.r.t  $M$  is

$$\nabla_M L_{MI} = -2\left(\frac{1}{n_Y} S M - \frac{1}{en_Y^2} e^S M \mathbf{1} \mathbf{1}^T\right) \quad (30)$$

And thus the gradient w.r.t  $W$  is

$$\nabla_W L_{MI} = -\left(\frac{1}{n_Y} (\mathbf{0}, I) S M - \frac{1}{en_Y^2} (\mathbf{0}, I) e^S M \mathbf{1} \mathbf{1}^T\right) \quad (31)$$

Where  $(\mathbf{0}, I) \in R^{n_T, n_S + n_T}$ ,  $I$  is identity matrix with size  $n_T$ .

In practice, we find it harmful to minimize  $L_{MI}$  by  $\nabla_W L_{MI}$  directly, because it will encourage the distribution to concentrate on only a few samples rapidly. We thus adjust the decent direction of  $L_{MI}$  to update the distribution slowly. Let  $|\nabla_W L_{Laplacian}|$  be the entry-wise absolute value of  $\nabla_W L_{Laplacian}$ . The descent directions are:  $d_1 = -\nabla_W L_{Laplacian}$  and  $d_2 = -|\nabla_W L_{Laplacian}| \odot (\nabla_W L_{MI})$ , where  $\odot$  is entry-wise multiplication. Due the property of  $\nabla_W L_{Laplacian}$  which is high on the margin of each conditional distribution, this yield a diffusion-like update of  $W$ , which prevent rapid collapse of surrogate distribution.

Therefore, the update rule of  $W$  is  $W^{k+1} = \Pi(W^k + \eta_1 d_1^k + \eta_2 d_2^k)$ , where  $\Pi$  is the projection operator onto probability simplex for each column of  $W$ .  $\eta_1$  and  $\eta_2$  are learning rate. The iteration is performed  $T$  times.

## 8.5 IMPLEMENTATION DETAILS

For each transfer task, mean ( $\pm$ std) over 5 runs of the test accuracy are reported. We use the ImageNet (Deng et al., 2009) pre-trained ResNet-50 (He et al., 2016) without final classifier layer as the encoder network  $G$  for Office-31 and Office-Home, and ResNet-101 for VisDA-2017. Following Kang et al. (2019), the final classifier layer of ResNet is replaced with the task-specific fully-connected layer to parameterize the classifier  $F$ , and domain-specific batch normalization parameters are used. Code is attached in supplementary materials.

The model is trained in the finetune protocol, where the learning rate of the classifier layer is 10 times that of the encoder, by mini-batch stochastic gradient descent (SGD) algorithm with momentum of 0.9 and weight decay of  $5e - 4$ . The learning rate schedule follows (Long et al., 2017a; 2015; Ganin and Lempitsky, 2015), where the learning rate  $\eta_p$  is adjusted following  $\eta_p = \frac{\eta_0}{(1+ap)^b}$ , where  $p$  is the normalized training progress from 0 to 1.  $\eta_0$  is the initial learning rate, i.e. 0.001 for the encoder layers and 0.01 for the classifier layer. For Office-31 and Office-Home,  $a = 10$  and  $b = 0.75$ , while for VisDA-2017,  $a = 10$  and  $b = 2.25$ . The coefficients of  $L_{MI}$  and  $L_{Auxiliary}$  are  $\alpha_1 = 0.3, \alpha_2 = 0.1$  for Office-31,  $\alpha_1 = 1.3, \alpha_2 = 1.0$  for Office-Home, and  $\alpha_1 = 3.0, \alpha_2 = 1.0$  for VisDA-2017. The hyperparameters of surrogate distribution optimization include K-nearest Graph  $K = 3$ , number of iterations  $T = 3$ , learning rate  $\eta_1 = 0.5$  and  $\eta_2 = 0.05$ .

Experiments are conducted with Python3 and Pytorch. The model is trained on single NVIDIA GeForce RTX 2080 Ti graphic card. For Office-31, each epoch takes 80 seconds and 10 seconds to perform inference. Code is attached in supplementary materials.

## 8.6 VISUALIZATION

Our visualization experiment is carried out on  $D \rightarrow A$  and  $W \rightarrow A$  tasks in the data set Office31, which are the two most difficult tasks in Office31. The baselines we chose were ResNet-50 pre-trained on ImageNet and CDAN. We chose CDAN because it is a typical conditional domain alignment method. Pre-trained Resnet-50 is fine-tuned on the source domain and then tested on the target domain. Results of CDAN are obtained by running the official code. We train all the models until convergence, then encode the data of source domain and target domain with the model, and take representation before the final linear classification layer as feature vectors. We use t-SNE to visualize the features, using the t-SNE function of scikit-learn with default parameters. The results are in the link.

Figure 3 shows the results of task  $D \rightarrow A$ . From top to bottom are the feature visualizations on task  $D \rightarrow A$  of ResNet-50, CDAN, and SIDA, respectively. The left column is the feature comparison of the source and target domains. Red represents the source domain, and blue represents the target domain. The results show that SIDA emphasises discriminability of features. The right column shows the feature of different classes on the target domain. SIDA makes target features better distinguishable.

Figure 4 shows the results of task  $W \rightarrow A$ . The results on task  $W \rightarrow A$  are similar to task  $D \rightarrow A$ .

The visualization results show that SIDA can make the features of different categories more distinguishable, a natural consequence of maximizing MI among features from the same category. Thus features can be easier for classification, as the visualization shows.

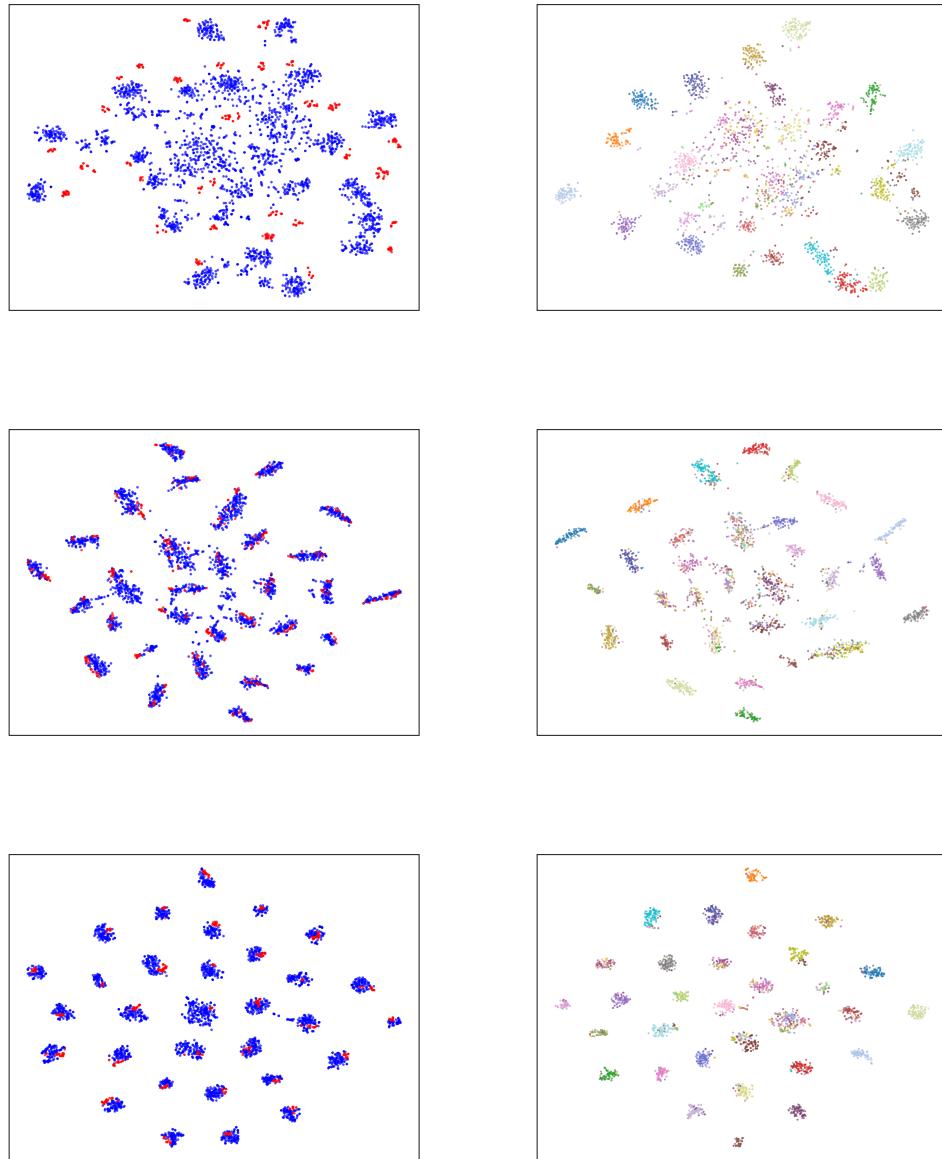


Figure 3: From top to bottom are the feature visualizations on task D→A of ResNet50, CDAN, and SIDA, respectively.

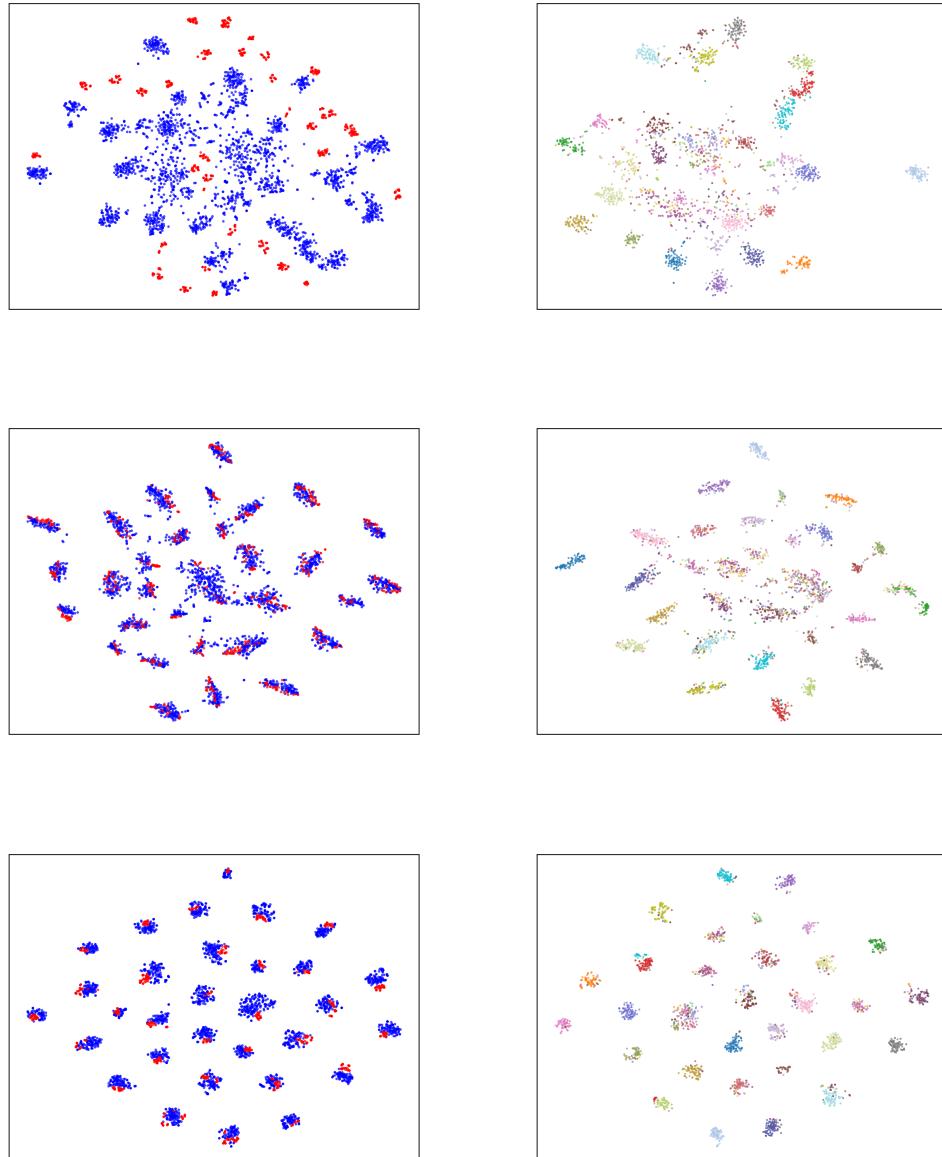


Figure 4: The result of task  $W \rightarrow A$  is similar to task  $D \rightarrow A$ .