Learning Disentangled Semantic Representation for Domain Adaptation

Ruichu Cai[1] , **Zijian Li**[1] , **Pengfei Wei**[2] , **Jie Qiao**[1] , **Kun Zhang**[3] and **Zhifeng Hao**[4]
[1]School of Computers, Guangdong University of Technology, China
[2]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[3]Department of Philosophy, Carnegie Mellon University, USA
[4]School of Mathematics and Big Data, Foshan University, China
cairuichu@gdut.edu.cn, leizigin@gmail.com, wpf89928@gmail.com, kunz1@cmu.edu,
qiaojie.chn@gmail.com, zfhao@gdut.edu.cn

Abstract

Domain adaptation is an important but challenging task. Most of the existing domain adaptation methods struggle to extract the domain-invariant representation on the feature space with entangling domain information and semantic information. Different from previous efforts on the entangled feature space, we aim to extract the domain invariant semantic information in the latent disentangled semantic representation (DSR) of the data. In DSR, we assume the data generation process is controlled by two independent sets of variables, i.e., the semantic latent variables and the domain latent variables. Under the above assumption, we employ a variational auto-encoder to reconstruct the semantic latent variables and domain latent variables behind the data. We further devise a dual adversarial network to disentangle these two sets of reconstructed latent variables. The disentangled semantic latent variables are finally adapted across the domains. Experimental studies testify that our model yields state-of-the-art performance on several domain adaptation benchmark datasets.

1 Introduction

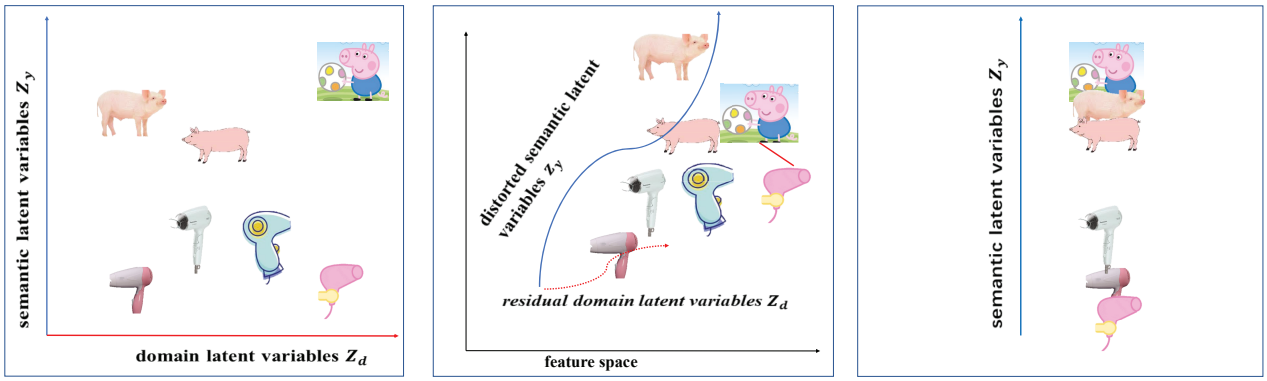
Domain adaptation is an important but challenging task. Since the acquisition of a large labeled data is usually either expensive or impractical, how to train on the unlabeled target domain with the help of labeled source domain has become a particular focus. However, this learning scheme suffers from a well-known phenomenon named *domain shift*, leading an urging motivation in building an adaptive classifier that can efficiently transfer the source labeled data under the domain shift, this problem is also known as *unsupervised domain adaptation*.

An essential approach in unsupervised domain adaptation is to understand what the domain-invariant representation across the domains is and how to find it [Zhang *et al.*, 2013; Pan *et al.*, 2011; Gong *et al.*, 2012]. Typical methodologies explored in the literature include the feature alignment approaches that extract domain-invariant representation by minimizing the discrepancy between the feature distributions inside deep feed-forward architectures [Tzeng *et al.*, 2014; Long *et al.*, 2015; 2016; 2017], and the adversarial learning approaches that extract the representation by deceiving the domain discriminators [Tzeng *et al.*, 2015; Ganin and Lempitsky, 2015; Ganin *et al.*, 2016; Long *et al.*, 2018]. Recently, a fine-grained semantic alignment has also been proposed in order to extract domain-invariant representation under the consideration of the semantic information [Xie *et al.*, 2018; Zhang *et al.*, 2018; Chen *et al.*, 2018]. However, most of them require target pseudo labels in order to minimize the discrepancies across domains within the same labels, and thus resulting the error accumulation due to the uncertainty of the pseudo-labeling accuracy.

Due to the complex manifold structures underlying the data distributions, these methods mainly suffer a false alignment problem [Pei *et al.*, 2018; Xie *et al.*, 2018]. As shown in Figure 1(a), the data generation process is controlled by the disentangled domain latent variables and semantic latent variables in the latent manifold. The ideal *semantic* position of the two types of labels – pig and hair drier – should be placed relatively upper and lower on the manifold according to the semantic axis, and the ideal *domain* position within the same labels should be placed in the left and right according to the domain axis. However, as shown in Figure 1(b), once the domain information is not completely removed, samples are distorted on the feature manifold, leading to the false alignment problem, e.g., the Peppa Pig looks like a pink hair-drier and thus the feature of the Peppa Pig might be near to that of the hair-drier in the distorted feature manifold.

Motivated by the example of Figure 1(b), the underlying cause of the false alignment problem is the entanglement of the semantic and domain information. More specifically, samples are controlled by two sets of independent latent variables $\mathbf{z}_y$ and $\mathbf{z}_d$. However, these two sets of latent variables are highly tangled and distorted on the high dimensional feature manifold space. It is very challenging to remove the domain information while preserving the semantic information on a complex tangled feature manifold space.

In this work, motivated by the disentanglement property of the multiple explanatory factors in the representation learning literature [Bengio *et al.*, 2013; Dinh *et al.*, 2014], we propose a Disentangle Semantic Representation learning model (DSR in short) by assuming the independence between the semantic variables $\boldsymbol{z}_y$ and the domain variables $\boldsymbol{z}_d$. Our DSR reconstructs the disentangled latent space and simultaneously



(a) Latent manifold of the data generation. (b) Distorted feature manifold with residual domain information (c) Disentangle semantic manifold

Figure 1: A toy domain adaption example with "pig" and "hair drier" samples on the "art" and "camera" domain. (a) The data generation process is controlled by the disentangled domain latent variables and semantic latent variables in the latent manifold. (b) The samples are distorted on the feature manifold, and the residual domain information results in the false alignment between the Peppa Pig and the hair drier. (c) The samples are well distributed on the disentangled semantic manifold. (*best view in color*)

uses the semantic variables to predict the target labels. The underlying intuition, as shown in Figure 1(c), is that by using the disentangle semantic latent variables that are independent to the domain latent variables, we can easily classify the labels into two categories merely based on the semantic axis $\mathbf{z}_y$. We employ a variational auto-encoder to reconstruct the disentangled semantic and domain latent variables, with the help of a dual adversarial network. The extensive experimental studies demonstrate that DSR outperforms state-of-the-art unsupervised domain adaptation methods on standard domain adaptation benchmarks.

2 Related Work

Deep feature learning methods have been shown very effective for unsupervised domain adaptation. The key idea of the deep feature learning is to extract domain-invariant representation by aligning different domains. Some works utilize maximum mean discrepancy(MMD) to realize the domain alignment. [Tzeng *et al.*, 2014] learns domain-invariant representation by adding an adaptation layer and an additional domain confusion loss; [Long *et al.*, 2015] reduces the domain discrepancy by using an optimal multi-kernel selection method. [Long *et al.*, 2016] assumes that the source classifier and target classifier differ by a residual function and enable classifier adaptation by plugging several layers with reference to the target classifier. [Long *et al.*, 2017] learns domain-invariant representation by aligning the joint distributions of multiple domain-specific layers across domains based on a joint maximum mean discrepancy criterion. Other works introduce a domain adversial layer for the domain alignment. [Ganin and Lempitsky, 2015] introduces a gradient reversal layer to fool the domain classifier and extracts the domain-invariant representation, [Tzeng *et al.*, 2017] borrows the idea of generative adversarial network(GAN)[Goodfellow *et al.*, 2014] and proposes a novel unified framework for adversarial domain adaptation.

Recent studies also show the benefits of semantic alignment to unsupervised domain adaptation. With the assumption that the distance among the samples with the same label but from different domains should be as small as possible, [Xie *et al.*, 2018] learns semantic representation by aligning labeled source centroid and pseudo-labeled target centroid. [Chen *et al.*, 2018] aligns the discriminative features across domains progressively and effectively, via utilizing the intra-class variation in the target domain. [Deng *et al.*, 2018] proposes a similarity constrained alignment method which enforces a similarity-preserving constraint to maintain class-level relations among the source and target samples.

However, most of the semantic alignment methods require target pseudo labels to minimize the variety discrepancies across domains, and thus resulting the error accumulation due to the uncertainty of the pseudo-labeling accuracy. In this work, we employ the concept of variational auto-encoder [Kingma and Welling, 2013] and adversarial learning to extract the domain invariant semantic representation.

3 Disentangled Semantic Representation Model

In this work, we focus on the unsupervised domain adaptation problem that uses the labeled samples $D_S = \{\boldsymbol{x}_i^S, y_i^S\}_{i=1}^{n_S}$ on the source domain to classify the unlabeled samples $D_T = \{\boldsymbol{x}_j^T\}_{j=1}^{n_T}$ on the target domain. The goal of this paper is to understand:(1) what the domain-invariant representation across domains is, and (2) how to design a framework that can extract such a domain-invariant representation.

Regarding the first point, we start from the causal mechanism behind the data generation process as shown in Figure. 2. Given $\boldsymbol{x}$, it is generated from two independent latent variables, .i.e, $\boldsymbol{z}_d$ encodes the domain information and $\boldsymbol{z}_y$ encodes the semantic information. $\mathbf{z}_d \in \mathbb{R}^{K_d}$ and $\mathbf{z}_y \in \mathbb{R}^{K_y}$ denote the semantic latent variables and the domain latent variables respectively. Considering that the domain information

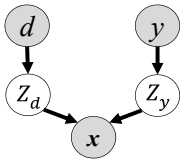


Figure 2: The causal model of data generation process, which are controlled by the latent variables $\boldsymbol{z}_d$ and $\boldsymbol{z}_y$.

may differ considerably across domains, we induce that the semantic latent variables play an important role in extracting the domain-invariant representation. Let $\boldsymbol{z} = \{\mathbf{z}_y, \mathbf{z}_d\}$. By further developing the independence property in the latent space, we also assume that $\mathbf{z}_y \perp\!\!\!\perp \mathbf{z}_d$.

Regarding the second point, with the above data generative mechanism, we propose a disentangled semantic representation (DSR) domain adaptation framework by first reconstructing the two independent latent variables via variational auto-encoder and then disentangling them through a dual adversarial training network. The key structure of the proposed framework is given in Figure. 3.

As shown in the upper part of Figure. 3, the "Reconstruction" architecture, we first obtain a feature via a backbone feature extractor $G(.)$ .e.g ResNet, and then use the VAE-like scheme to reconstruct the feature by first encoding it into the latent variables $\boldsymbol{z}$, then use the latent variables to reconstruct the feature $G(\boldsymbol{x})$ from two independent latent variables $\boldsymbol{z}_y$ and $\boldsymbol{z}_d$. However, unlike the vanilla VAE, we further design a "Disentanglement" architecture as shown in the green dot box of Figure. 3. In this architecture, two adversarial modules are placed under the semantic latent variables and domain latent variables respectively. For the label adversarial learning module on the left side, it aims to pull all the semantic information into $\boldsymbol{z}_y$ and push all the domain information from $\boldsymbol{z}_d$. For the domain adversarial learning module on the right side, it aims to pull all the domain information into $\boldsymbol{z}_d$ and push all the semantic information from $\boldsymbol{z}_y$. By doing so, we can obtain those domain-invariant semantic information without the contamination of the domain information.

We introduce more technical details of our proposed framework in the following section.

3.1 Semantic Latent Variables Reconstruction

For the reconstruction architecture in DSR framework, we follow the configuration in VAE. We denote $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ as the *encoder* with respect to $\boldsymbol{\phi}$ to approximate the intractable true posterior $p(\boldsymbol{z}|\boldsymbol{x})$. The variational lower bound of the marginal likelihood is given as follow

$$\begin{aligned}&\mathcal{L}_{\mathrm{ELBO}}(\boldsymbol{\phi}, \theta_r) = \\ &- D_{KL}\left(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||P(\boldsymbol{z})\right) + \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log P_{\theta_r}(\boldsymbol{x}|\boldsymbol{z})\right], \quad (1)\end{aligned}$$

where $P_{\theta_r}(\boldsymbol{x}|\boldsymbol{z})$ denotes the *decoder* with respect to the parameters θ_r and $P(\boldsymbol{z})$ is the prior distribution. Then, we further decompose the latent variables $\boldsymbol{z}$ into $\boldsymbol{z_y}$ and $\boldsymbol{z_d}$. Eq. (1)

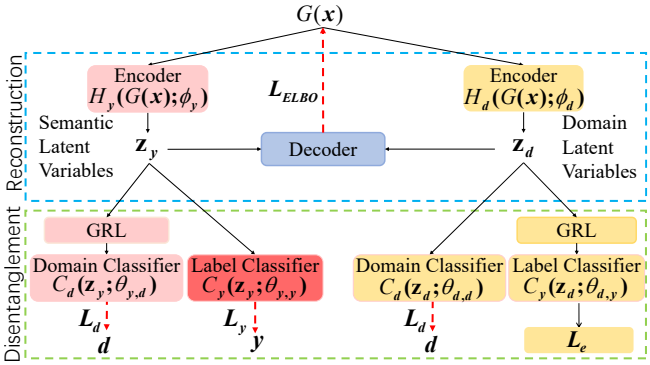


Figure 3: The framework of the Disentangled Semantic Representation model. In the reconstruction block (marked with the blue dashed lines), the variational auto-encoder is used to recover the semantic latent variables ($\boldsymbol{z}_y$) and the domain latent variables ($\boldsymbol{z}_d$). In the disentanglement block (marked with the green dashed lines), a dual adversarial network is used to disentangle the latent variables. H_y and H_d are the encoders for the semantic and domain information respectively. C_y and C_d are the classifiers for the label and domain respectively. GRL is a gradient reversal layer that multiplies the gradient by a negative constant. (*best view in color*)

can be derived as follows:

$$\begin{aligned}\mathcal{L}_{\mathrm{ELBO}}(\phi_y, \phi_d, \theta_r) = &-D_{KL}(q_{\phi_y}(\boldsymbol{z}_y|G(\boldsymbol{x}))\|P(\boldsymbol{z}_y)) \\ &-D_{KL}(q_{\phi_d}(\boldsymbol{z}_d|G(\boldsymbol{x}))\|P(\boldsymbol{z}_d)) \quad (2) \\ &+\mathbb{E}_{q_{\phi_{y,d}}(\boldsymbol{z}_y,\boldsymbol{z}_d|G(\boldsymbol{x}))}[\log P_{\theta_r}(G(\boldsymbol{x})|\boldsymbol{z}_y,\boldsymbol{z}_d)].\end{aligned}$$

Here, we assume that $P(\boldsymbol{z}_y), P(\boldsymbol{z}_d) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and ϕ_y and ϕ_d are the parameters of the encoder. Similar to VAE, by applying a reparameterization trick, we use MLP $H_y(G(\mathbf{x}); \phi_y)$ and $H_d(G(\mathbf{x}); \phi_d)$ as the universal approximator of q to encode the data into $\boldsymbol{z}_y$ and $\boldsymbol{z}_d$ respectively.

3.2 Semantic Latent Variables Disentanglement

For the disentanglement architecture of the DSR framework, it consists of two adversarial modules working together following the typical configuration in [Ganin and Lempitsky, 2015]. On the left of the Figure. 3 under the semantic latent variables $\boldsymbol{z}_y$ is the *label adversarial learning module* that fuses the semantic information and excludes all the domain information. This is done by using a label classifier C_y and a domain classifier C_d. To exclude the domain information, we use a gradient reversal layer (GRL) for C_d. As a result, the parameters ϕ_y in H_y are learned by maximizing the loss L_d of C_d and simultaneously minimizing the loss L_y of C_y. The parameters $\theta_{y,d}$ of C_d are learned by L_d. The overall objective function of label adversarial learning module is shown as follows:

$$\begin{aligned}&\mathcal{L}_{sem}(\phi_y, \theta_{y,y}, \theta_{y,d}) = \\ &\quad \frac{\delta}{n_S} \sum_{x_i^s \in D_S} L_y\left(C_y\left(H_y\left(G(\boldsymbol{x}); \phi_y\right); \theta_{y,y}\right), y_i\right) \\ &\quad -\frac{\lambda}{n} \sum_{x_i \in (D_S, D_T)} L_d\left(C_d\left(H_y\left(G(\boldsymbol{x}); \phi_y\right); \theta_{y,d}\right), d_i\right), \quad (3)\end{aligned}$$

where $n = n_S + n_T$, λ are a trade-off parameters that balance the two objectives and δ is the parameter that controls the weight of L_y. A bigger δ enables the label classifier to learn more semantic information. The default value of δ is 1, and we try different values in order to validate the individual contributions of the domain adversarial learning module in the section 4.

Similarly, on the right-side adversarial module is the *domain adversarial learning module* that fuses the domain information to $\boldsymbol{z}_d$ and excludes the semantic information from $\boldsymbol{z}_d$. The GRL is placed on the label classifier above in order to absorb all the domain information from $\boldsymbol{z}_y$. However, unlike the semantic module, we do not use cross-entropy as the label loss, because of the unsupervised learning in the target domain. In order to utilize the data in the target domain, we employ maximum entropy loss L_e for the label classifier C_y. As a result, the parameters ϕ_d in H_d are learned by maximizing the loss L_e of label classifier C_y but minimizing the loss L_d of label classifier C_d. In addition, the parameters $\theta_{d,y}$ of label classifier C_y are learned by minimizing its own loss L_e. The objective of the domain adversarial learning module is shown as follow

$$\begin{aligned}\mathcal{L}_{dom}(\phi_d, \theta_{d,d}, \theta_{d,y}) = \\ \frac{1}{n} \cdot \sum_{x_i \in (D_S, D_T)} L_d \left(C_d \left(H_d \left(G(\boldsymbol{x}); \phi_d\right); \theta_{d,d}\right), d_i\right) \\ -\frac{\omega}{n} \cdot \sum_{x_i \in (D_S, D_T)} L_E \left(C_y \left(H_d \left(G(\boldsymbol{x}); \phi_d\right); \theta_{d,y}\right)\right),\end{aligned} \tag{4}$$

where ω is the trade-off parameter between the two objectives that shapes the feature during.

3.3 Model Summary

By combining the reconstruction and the disentanglement, we summarize the model as follows.

The total loss of the proposed disentangled semantic representation learning for domain adaptation model is formulated as:

$$\begin{aligned}\mathcal{L}(\phi_y, \theta_{y,d}, \theta_{y,y}, \phi_d, \theta_{d,d}, \theta_{d,y}, \theta_r) = \\ \mathcal{L}_{\text{ELOB}} + \beta \mathcal{L}_{sem} + \gamma \mathcal{L}_{dom},\end{aligned} \tag{5}$$

where β and γ are the hyper-parameters that is not very sensitive and we set β=1 and γ=1.

Under the above objective function our model is trained on the source domain using the following procedure

$$\begin{aligned}(\hat{\phi_y}, \hat{\theta_{y,y}}, \hat{\phi_d}, \hat{\theta_{d,y}}, \hat{\theta_r}) = \\ \underset{\phi_y, \theta_{y,y}, \phi_d, \theta_{d,y}, \theta_r}{\arg\min} \mathcal{L}(\phi_y, \theta_{y,d}, \theta_{y,y}, \phi_d, \theta_{d,d}, \theta_{d,y}, \theta_r) \\ (\hat{\theta_{y,d}}, \hat{\theta_{d,d}}) = \underset{\theta_{y,d}, \theta_{d,d}}{\arg\max} \mathcal{L}(\phi_y, \theta_{y,d}, \theta_{y,y}, \phi_d, \theta_{d,d}, \theta_{d,y}, \theta_r).\end{aligned} \tag{6}$$

The following classifier with the trained optimal parameters is adapted to the target domains.

$$y = C_y \left(H_y \left(G(\boldsymbol{x}); \hat{\phi_y}\right); \hat{\theta_{y,y}}\right). \tag{7}$$

3.4 Analysis

In this section, we first show that the error will be reduced under the domain-invariant feature space, and second we further develop an upper bound for the target generalization error.

Following the instruction in [Ben-David *et al.*, 2007]. Let $R : \mathcal{X} \to \mathcal{Z}$ denote the representation function, where $R \triangleq H \circ G$ follows the definition in Section 2. We denote D_S as the source distribution over $\mathcal{X}$ and $\tilde{D}_S$ as the induced distribution over the feature space $\mathcal{Z}$, i.e., $\Pr_{\tilde{\mathcal{D}}_S}[B] \stackrel{\text{def}}{=} \Pr_{\mathcal{D}_S}\left[\mathcal{R}^{-1}(B)\right]$ for any measurable event B. Similarly, we use parallel notation, D_T, $\tilde{D}_T$ for the target domain. The error on the source domain with a hypothesis h is defined as

$$\begin{aligned}\epsilon_S(h) =& \mathrm{E}_{\mathbf{z} \sim \tilde{D}_S}[\mathrm{E}_{y \sim \tilde{f}(\mathbf{z})}[y \neq h(\mathbf{z})]] \\ =& \mathrm{E}_{\mathbf{z} \sim \tilde{D}_S}[C(\mathbf{z}) - h(\mathbf{z})],\end{aligned} \tag{8}$$

where $C : \mathcal{Z} \to [0, 1]$ is the label classier defined on $\tilde{D}_S$.

Theorem 1. *Assume that the semantic and the domain factors are independent, i.e.,* $\mathbf{z}_y \perp\!\!\!\perp \mathbf{z}_d$. *Let* $\mathbf{z} = \{\mathbf{z}_y, \mathbf{z}_d\}$, *and the error on the disentangled source and target domain with a hypothesis* h *is*

$$\begin{aligned}\epsilon_S^y(h) = \epsilon_S(h) - \alpha_S, \\ \epsilon_T^y(h) = \epsilon_T(h) - \alpha_T,\end{aligned} \tag{9}$$

where $\alpha_{\mathrm{S}} := \mathrm{E}_{\mathbf{z}_d \sim \tilde{\mathcal{D}}_S}[C(\mathbf{z}_d) - h(\mathbf{z}_d)]$ *and* $\epsilon_{\mathrm{S}}^{\mathrm{Y}}(\mathrm{h}) \coloneqq \mathrm{E}_{\mathbf{z}_y \sim \tilde{\mathcal{D}}_S}[C(\mathbf{z}_y) - h(\mathbf{z}_y)]$ *denotes the error of DSR with respect to* h *in the source domain, while* $\epsilon_{\mathrm{T}}^{\mathrm{Y}}(\mathrm{h})$ *denotes the error of DSR in target domain.*

Proof. Since $\mathbf{z}_y \perp\!\!\!\perp \mathbf{z}_d$, we can further derive Eq. (8) as follow,

$$\begin{aligned}\epsilon_S(h) =& \mathrm{E}_{(\mathbf{z}_y, \mathbf{z}_d) \sim \tilde{\mathcal{D}}_S}[C(\mathbf{z}) - h(\mathbf{z})] \\ =& \underbrace{\mathrm{E}_{z_y \sim \tilde{\mathcal{D}}_{S_y}}[C(\mathbf{z}_y) - h(\mathbf{z}_y)]+}_{\epsilon_{\mathrm{S}}^{\mathrm{Y}}(\mathrm{h})} \\ & \underbrace{\mathrm{E}_{z_d \sim \tilde{\mathcal{D}}_{S_d}}[C(\mathbf{z}_d) - h(\mathbf{z}_d)]}_{\alpha_{\mathrm{S}}} \\ =& \epsilon_{\mathrm{S}}^{\mathrm{Y}}(\mathrm{h}) + \alpha_{\mathrm{S}}.\end{aligned} \tag{10}$$

In the second equality, based on the independence property between $\mathbf{z}_y$ and $\mathbf{z}_d$, the distribution of $\tilde{\mathcal{D}}_S$ can be decomposed into two part so as to the error.

Similarly, we have $\epsilon_T^y(h) = \epsilon_T(h) - \alpha_T$, by decomposing the $\tilde{\mathcal{D}}_T$ using the independence property. □

Theorem 1 shows that the disentanglement of the representation space is helpful and might also necessary for obtaining less classify error. Then, in the following Theorem 2, we show that the less classify error on the source domain will tighten the error bound at the target domain.

Theorem 2. *Let* $R(x)$ *be a fixed representation function from* $\mathcal{X}$ *to* $\mathcal{Z}$ *and* $\mathcal{H}$ *be a hypothesis space. Let* $h^* = \arg\min_{h \in \mathcal{H}}(\epsilon_T(h), \ \epsilon_S(h))$, *and let* λ_S, λ_T *be the errors of* h^* *with respect to* $\mathcal{D}_S$ *and* $\mathcal{D}_T$ *respectively, i.e.,* $\lambda_T \coloneqq \epsilon_T(h^*)$ *and* $\lambda_S \coloneqq \epsilon_S(h^*)$. *We have*

$$\epsilon_T^y(h) \leq \eta + \epsilon_T^y(S) + d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right), \tag{11}$$

where $\eta \coloneqq \epsilon_T^y(h^*) + \alpha_T^* + \epsilon_S^y(h^*) + \alpha_S^* + \alpha_S - \alpha_T$.

Proof. Applying [Ben-David *et al.*, 2007, Theorem 1], we have

$$\epsilon_T(h) \leq \lambda_T + \lambda_S + \epsilon_S(h) + d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right), \tag{12}$$

where

$$d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right) = 2 \sup_{h \in \mathcal{H}} \left|\Pr_{D_S}[\mathcal{Z}_h \triangle \mathcal{Z}_{h^*}] - \Pr_{D_T}[\mathcal{Z}_h \triangle \mathcal{Z}_{h^*}]\right|.$$

Then based on Theorem 1, combining with Eq. (12) and Eq. (10), we further obtain

$$\begin{aligned} \epsilon_T^y(h) &\leq \lambda_T + \lambda_S + \epsilon_S(h) + d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right) - \alpha_T \\ &\leq \lambda_T + \lambda_S + \epsilon_S(h) + d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right). \end{aligned} \tag{13}$$

Note that based on Eq. (10), we denote $\lambda_T = \epsilon_T(h^*) = \epsilon_T^y(h^*) + \alpha_T^*$. and the error upper bound for the target domain can be further derived as follow,

$$\begin{aligned} \epsilon_T^y(h) &\leq \lambda_T + \lambda_S + \epsilon_S(h) + d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right) - \alpha_T \\ &\leq \epsilon_T^y(h^*) + \alpha_T^* + \epsilon_S^y(h^*) + \alpha_S^* + \epsilon_S^y(h) \\ &\quad + \alpha_S + d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right) - \alpha_T \\ &\leq \eta + \epsilon_T^y(S) + d_{\mathcal{H}}\left(\tilde{\mathcal{D}}_S, \tilde{\mathcal{D}}_T\right), \end{aligned} \tag{14}$$

where $\eta \coloneqq \epsilon_T^y(h^*) + \alpha_T^* + \epsilon_S^y(h^*) + \alpha_S^* + \alpha_S - \alpha_T$. □

4 Experiments and Results

4.1 Setup

Office-31 is a standard benchmark for visual domain adaptation, which contains 4,652 images and 31 categories from three distinct domains: Amazon (A), Webcam (W) and DSLR (D).

Office-Home is a more challenging domain adaptation dataset than Office-31, which consists of around 15,500 images from 65 categories of everyday objects. This dataset is organized into four domains: Art (Ar), Clipart (Cl), Product (Pr) and Real-world (Rw).

Compared Approaches

Beside the classical approaches, we also compare our disentangled semantic representation model with some deep transfer learning methods. Two recently proposed semantic enhanced methods, CDAN [Long *et al.*, 2018] and MSTN [Xie *et al.*, 2018], are also compared in the experiment. Note that, CDAN conditions the adversarial adaptation models on discriminative information conveyed in the classifier predictions, and MSTN learns semantic representation by aligning labeled source centroid and pseudo-labeled target centroid.

4.2 Result

Office-31 Result

The classification accuracies on the Office-31 dataset for unsupervised domain adaptation on ResNet-50 are shown in Table 1. Our DSR model significantly outperforms all other baselines on most of the transfer tasks. It is remarkable that our method promotes the classification accuracies substantially on hard transfer tasks, e.g., D→A, W→A, and produces comparable results on the other relatively simple tasks, e.g., A→W, D→W. However, the result of DSR on the W→D and A→D tasks are lower than that of some compared approaches. This is because domain *DSLR* (D) only has total 498 images for 31 classes and some classes even only have less than 10 samples, it's insufficient for our method to reconstruct the disentangled semantic representation. We conduct the Wilcoxon signed-rank test [Lowry, 2014] on the reported accuracies, the results are as follows: on the Office-31 data set, our method is comparable with CDAN-M, and significantly outperforms all the other baselines with the p-value threshold 0.05.

Office-home Result

Similar to the results on Office-31, the DSR model also outperforms all other baselines on most of the tasks, as reported in Table 2. Note that our method achieves a great improvement when the source domain is Art (Ar) or Clipart (Cl) and performs slightly worse than CDAN-M when the source domain is Real World (RW). This is because the Ar and the clipart (Cl) domain contain relatively simpler pictures and more complex scenarios than the other domains and our DSR model can extract such semantic representation easily and thus achieve good performance on this source domain. However, in the Real World (RW) domain, the pictures are taken in real life and there a lot of ambiguous samples, .e.g., pictures with monitor, computer and laptop are tagged with the same label, which implies that the semantic information is difficult to be disentangled and extracted on this domain. We also conduct the Wilcoxon signed-rank test [Lowry, 2014] on the reported accuracies, our method significantly outperforms the baselines, with the p-value threshold 0.05.

The Study of the Disentangled Semantic Representation

To study the effectiveness of the disentangled semantic representation, we compare our methods with two approaches using similarly adversarial learning strategy but different representations on the task **Ar→Cl**. Figure. 4 (a)-(c) show the visualization of the extracted features using t-SNE. As shown in the figure, DSR obtains the best alignment among the three representations. Both DANN and MSTN have a large number of samples are falsely aligned. This result verifies the effectiveness of DSR.

Such results also can be observed in Table 1 and Table 2. For example, DSR achieves remarkably outstanding results on some transfer tasks, e.g. D→A, W→A on the Office-31 dataset and all the tasks except for those using RW as the source domain on the Office-home dataset. These results show a common phenomenon that the samples of the source domain are more complex than that of the target domain, i.e., the source domain has more scenarios than the target domain. Such common phenomenon also shows the advantage of our disentangled semantic representation.

Mehtods	$A \to W$	$D \to W$	$W \to D$	$A \to D$	$D \to A$	$W \to A$	Avg
ResNet-50 [He *et al.*, 2016]	68.4	96.7	99.3	68.9	62.5	60.7	76.1
TCA [Pan *et al.*, 2011]	72.7	96.7	99.6	74.1	61.7	60.9	77.6
GFK [Gong *et al.*, 2012]	72.8	95.0	98.2	74.5	63.4	61.0	77.5
DAN [Long *et al.*, 2015]	80.5	97.1	99.6	78.6	63.6	62.8	80.4
RTN [Long *et al.*, 2016]	84.5	96.8	99.4	77.5	66.2	64.8	81.6
DANN [Ganin *et al.*, 2016]	82.0	96.9	99.1	79.7	68.2	67.4	82.2
ADDA [Tzeng *et al.*, 2017]	86.2	96.2	98.4	77.8	69.5	68.9	82.9
JAN [Long *et al.*, 2017]	85.4	97.4	99.8	84.7	68.6	70.0	84.3
MSTN [Xie *et al.*, 2018]	86.9	96.7	99.9	87.3	66.9	68.4	84.3
CDAN-M [Long *et al.*, 2018]	**93.1**	98.6	**100.0**	**93.4**	71.0	70.3	87.7
DSR_DM($\delta = 1$)	90.7	97.1	99.2	88.8	71.5	71.2	86.4
DSR_DM($\delta = 2$)	91.5	97.2	99.5	88.5	71.5	72.1	86.7
Ours	**93.1**	**98.7**	99.8	92.4	**73.5**	**73.9**	**88.6**

Table 1: Accuracy (%) on Office-31 for unsupervised domain adaptation (ResNet)

Method	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 [He *et al.*, 2016]	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 [Long *et al.*, 2015]	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 [Ganin *et al.*, 2016]	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 [Long *et al.*, 2017]	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
MSTN [Xie *et al.*, 2018]	49.3	67.6	74.7	49.6	63.7	64.5	50.7	43.7	73.0	62.7	52.2	77.9	60.9
CDAN-M [Long *et al.*, 2018]	50.6	65.9	73.4	55.7	62.7	64.2	51.8	49.1	74.5	**68.2**	**56.9**	**80.7**	62.8
DSR_DM($\delta = 1$)	50.5	69.2	75.5	52.9	64.0	65.6	53.0	45.0	73.5	63.2	50.8	78.3	61.8
DSR_DM($\delta = 2$)	52.3	70.9	76.5	54.0	65.2	67.0	53.5	45.0	73.7	62.7	50.7	78.7	62.5
DSR	**53.4**	**71.6**	**77.4**	**57.1**	**66.8**	**69.3**	**56.7**	**49.2**	**75.7**	68.0	54.0	79.5	**64.9**

Table 2: Accuracy (%) on Office-home for unsupervised domain adaptation (ResNet)

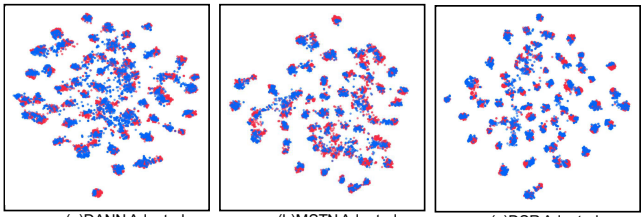


Figure 4: The t-SNE visualization of deep features extracted by DANN (a), MSTN (b) and DSR (c). The red points are source domain samples and the blue points are target domain samples.

Ablation Study of the Dual Adversarial Learning

To study the effectiveness of the dual adversarial learning module, we first train the standard DSR model until convergence, then train the model without the domain adversarial learning module. Such ablated model is named DSR_WD in the experiment. The experiment results are shown in Table 1 and Table 2. Comparing the result of DSR and DSR_WD (δ=1), we find that the performance drops because the semantic information is drained away without disentanglement. Even when we use $\delta = 2$, the performance of the ablated model is still worse than the original one. These results verify that the dual adversarial learning module can push the semantic information into the semantic latent variables $\mathbf{z}_y$, and simultaneously, push the domain information into the domain latent variables $\mathbf{z}_d$.

5 Conclusion

This paper presents a disentangled semantic representation model for the unsupervised domain adaptation task. Different from previous work , our approach extracts the disentangled semantic representation on the recovered latent space, following the causal model of the data generation process. Our approach is also featured with the variational auto-encoder based latent space recovery and the dual adversarial learning based disentangle of the representation. The success of the proposed approach not only provides an effective solution for the domain adaptation task, but also opens the possibility of disentanglement based learning methods.

Acknowledgments

This research was supported in part by NSFC-Guangdong Joint Found (U1501254), Natural Science Foundation of China (61876043), Natural Science Foundation of Guangdong (2014A030306004, 2014A030308008), Guangdong High-level Personnel of Special Support Program (2015TQ01X140) and Pearl River S&T Nova Program of Guangzhou (201610010101). Kun Zhang would like to acknowledge the support by National Institutes of Health (NIH) under Contract No. NIH-1R01EB022858-01, FAINR01EB022858, NIH-1R01LM012087, NIH5-5U54HG008540-02, and FAIN- U54HG008540, by the United States Air Force under Contract No. FA8650-17-C-7715, and by National Science Foundation (NSF) EAGER Grant No. IIS-1829681. The NIH, the U.S. Air Force, and the NSF are not responsible for the views reported here. We would like to thank Dr Tom Fu from ADSC and professor Ke Yiping from Nanyang Technological University for their help and supports on this work.

References

[Ben-David *et al.*, 2007] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in neural information processing systems*, pages 137–144, 2007.

[Bengio *et al.*, 2013] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[Chen *et al.*, 2018] Chaoqi Chen, Weiping Xie, Tingyang Xu, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. *arXiv preprint arXiv:1811.08585*, 2018.

[Deng *et al.*, 2018] Weijian Deng, Liang Zheng, and Jianbin Jiao. Domain alignment with triplets. *arXiv preprint arXiv:1812.00893*, 2018.

[Dinh *et al.*, 2014] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[Ganin and Lempitsky, 2015] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*, pages 1180–1189. JMLR. org, 2015.

[Ganin *et al.*, 2016] 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.

[Gong *et al.*, 2012] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2066–2073. IEEE, 2012.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[He *et al.*, 2016] 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.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Long *et al.*, 2015] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I Jordan. Learning transferable features with deep adaptation networks. *arXiv preprint arXiv:1502.02791*, 2015.

[Long *et al.*, 2016] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016.

[Long *et al.*, 2017] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2208–2217. JMLR. org, 2017.

[Long *et al.*, 2018] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1647–1657, 2018.

[Lowry, 2014] Richard Lowry. Concepts and applications of inferential statistics. 2014.

[Pan *et al.*, 2011] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

[Pei *et al.*, 2018] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2018.

[Tzeng *et al.*, 2014] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

[Tzeng *et al.*, 2015] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015.

[Tzeng *et al.*, 2017] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 4, 2017.

[Xie *et al.*, 2018] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 5419–5428, 2018.

[Zhang *et al.*, 2013] Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain adaptation under target and conditional shift. In *International Conference on Machine Learning*, pages 819–827, 2013.

[Zhang *et al.*, 2018] Yexun Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. Domain-invariant adversarial learning for unsupervised domain adaption. *arXiv preprint arXiv:1811.12751*, 2018.