LAMDA: Label Matching Deep Domain Adaptation

Trung Le [1] **Tuan Nguyen** [1] **Nhat Ho** [2] **Hung Bui** [3] **Dinh Phung** [1 3]

Abstract

Deep domain adaptation (DDA) approaches have recently been shown to perform better than their shallow rivals with better modeling capacity on complex domains (e.g., image, structural data, and sequential data). The underlying idea is to learn domain invariant representations on a latent space that can bridge the gap between source and target domains. Several theoretical studies have established insightful understanding and the benefit of learning domain invariant features; however, they are usually limited to the case where there is no label shift, hence hindering its applicability. In this paper, we propose and study a new challenging setting that allows us to use a Wasserstein distance (WS) to not only quantify the data shift but also to define the label shift directly. We further develop a theory to demonstrate that minimizing the WS of the data shift leads to closing the gap between the source and target data distributions on the latent space (e.g., an intermediate layer of a deep net), while still being able to quantify the label shift with respect to this latent space. Interestingly, our theory can consequently explain certain drawbacks of learning domain invariant features on the latent space. Finally, grounded on the results and guidance of our developed theory, we propose the Label Matching Deep Domain Adaptation (LAMDA) approach that outperforms baselines on real-world datasets for DA problems.

1. Introduction

The great achievement of machine learning in general and deep learning in particular can be attributed to the significant advancement of in computational power and large-scale annotated datasets. However, in many application domains, it is often prohibitively labor-expensive, error-prone, and time-consuming to collect and label high-quality data sufficiently large to train accurate deep models, such as in the domain of medicine or autonomous driving. Domain adaptation (DA) or transfer learning has emerged as a vital solution for this issue by transferring knowledge from a label-rich domain (a.k.a. source domain) to a label-scarce domain (a.k.a. target domain). Along with DA methods (Ganin & Lempitsky, 2015; Tzeng et al., 2015; Long et al., 2015; Shu et al., 2018; French et al., 2018; Nguyen et al., 2021b;a; 2019; 2020) achieved impressive performance on real-world datasets of various application domains, theoretical results (Mansour et al., 2009; Ben-David et al., 2010; Redko et al., 2017; Zhang et al., 2019a; Cortes et al., 2019) are abundant to provide rigorous and insightful understanding of various aspects of transfer learning.

Moving beyond using fixed features and taking advantage of deep nets in learning rich and meaningful representations, DDA aims to learn domain invariant representations, i.e., intermediate representations whose distribution is the same in source and target domains. While relying on invariant representations helps to reduce the data shift between the source and target domains, Zhao et al. (2019) found that this might seriously cause the label shift. More specifically, it was shown that if the marginal label distributions are significantly different between the source and target domains, enforcing learning domain invariant representations leads to an increase of the general loss on the target domain. Moreover, while *data shift* can be understood as a divergence between the source and target data distributions, the *label shift* is harder to quantify. It is commonly interpreted as the difference in labeling mechanisms of the source and target domains (i.e., $p^s(y \mid \mathbf{x})$ and $p^t(y \mid \mathbf{x})$), however, it is not an explicit definition for the label shift since the mechanic to indicate how a source example couple to a target example is missing. Another explanation is using a divergence between the marginal label distributions of the source and target domains (i.e., $p^s(y)$ and $p^t(y)$), nevertheless, this naive approach is simple and ignores individual conditional distributions of labels w.r.t. data examples.

We propose in this paper a new theoretical setting for unsupervised DA which enables us to study the data and label shifts under a more rigorous framework. Specifically, let $\mathcal{H}^s$ be the hypothesis class on source domain, we introduce a transformation T that maps the target to source domains,

[1]Department of Data Science and AI, Monash University, Australia [2]University of Texas, Austin, USA [3]VinAI Research, Vietnam. Correspondence to: Trung Le <trunglm@monash.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

and hence inducing a new hypothesis class on the target domain $\mathcal{H}^t := \{h^t : h^t = h^s \circ T\}$, where $\circ$ represents the function composition operator and $h^s \in \mathcal{H}^s$. Given a target example $\mathbf{x}$, our motivation is to use T to find its counterpart source example $T(\mathbf{x})$ and then use $h^s(T(\mathbf{x}))$ for a prediction. We note that this setting is different from current popular literature (Mansour et al., 2009; Ben-David et al., 2010; Redko et al., 2017; Zhang et al., 2019a; Cortes et al., 2019) in which the source and target hypothesis classes are decoupled. Moreover, by coupling the source and target domains via the transformation T, our theory developed in the sequel has two most important advantages: i) it enables us to explicitly quantify the label shift, and ii) the transformation T can be constructed explicitly, e.g., as a deep net, to yield tractable implementation.

Equipped with this setting, we demonstrate that the loss in performance of a source hypothesis and its relevant target hypothesis w.r.t. T can be upper-bounded by a Wasserstein (WS) distance (Villani, 2008; Santambrogio, 2015) between the source data distribution and the push-forward distribution of the target one (i.e., *data shift*), and the expectation of the divergence between $p^t(y \mid \mathbf{x})$ and $p^s(y \mid T(\mathbf{x}))$, where $\mathbf{x}$ is sampled from the target data distribution (i.e., *label shift*). This bears similarity to previous results (Mansour et al., 2009; Ben-David et al., 2010; Redko et al., 2017; Cortes et al., 2019), however, different from existing work, we conduct our theoretical analysis in a new setting with multi-class classification, probabilistic label assignment mechanism (Vapnik, 1999), continuous loss functions, and use a WS to describe the data shift laying a novel framework for describing the data and label shifts on a latent space.

In our work, the transformation T is a deep neural network, which can be decomposed into $T = T^2 \circ T^1$ where T^1 is a sub-network mapping data examples to a latent space (an intermediate layer of T). Under this assumption, we theoretically demonstrate that to resolve the data shift by means of learning T to minimize the aforementioned WS distance, we can simultaneously *match the gap* between source and target distributions for learning domain-invariant representations on the latent space and *minimize a reconstruction loss* w.r.t. the ground metric c of a WS distance (cf. Theorem 4). Further, grounded by theory developed for our theoretical setting, we find a trade-off of strictly forcing learning domain-invariant features, that is, enforcing domain-invariant latent representations gradually hurts target performance. Although this result is similar to (Zhao et al., 2019), our theoretical analysis is performed in a more general setting (i.e., multi-class classification, probabilistic labeling mechanism, and continuous loss function) than (Zhao et al., 2019) (i.e., binary classification, deterministic labeling mechanism, and absolute loss). Additionally, we make use of Wasserstein distance rather than JS distance (Endres & Schindelin, 2006) as in (Zhao et al., 2019).

Our work suggests that the key ingredient to remedy the label shift is to encourage target samples to move to suitable source class regions on the latent space while reducing the data shift. With this motivation, we propose ***LA**bel **M**atching **D**omain **A**daptation* (LAMDA) with the aim to minimize the discrepancy gap between two domains and simultaneously reduce the label mismatch on the latent space. Different from existing works, LAMDA employs a multi-class discriminator to be aware of source class regions and an optimal transport based cost to encourage target samples for moving to their matching source class region on the latent space. We conduct extensive experiments on real-world datasets to compare LAMDA with state-of-the-art baselines. The experimental results on the real-world datasets show that our LAMDA is able to reduce the label mismatch and hence achieving better performances.

Related work. Several attempts have been proposed to characterize the gap between general losses of source and target domains in domain adaptation, notably (Mansour et al., 2009; Ben-David et al., 2010; Redko et al., 2017; Zhang et al., 2019a; Cortes et al., 2019). Ben-David & Urner (2014; 2012); Zhang et al. (2019a) study the impossibility theorems for domain adaptation, attempting to characterize the conditions under which it is nearly impossible to perform transferability between domains. PAC-Bayesian view on domain adaptation using weighted majority vote learning has been rigorously studied in (Germain et al., 2013; 2016). Zhao et al. (2019); Johansson et al. (2019) interestingly indicate the insufficiency of learning domain-invariant representation for successful adaptation. Specifically, Zhao et al. (2019) points out the degradation in target predictive performance if forcing domain invariant representations to be learned while two marginal label distributions of the source and target domains are overly divergent. Johansson et al. (2019) analyzes the information loss of non-invertible transformations and proposes a generalization upper bound that directly takes it into account. Optimal transport theory has been theoretically leveraged with domain adaptation (Courty et al., 2017). Moreover, our theory development, motivations, and obtained results in Section 2.3 are different from those in (Courty et al., 2017). In addition, we compare our proposed LAMDA to DeepJDOT (Damodaran et al., 2018) (a deep domain adaptation approach developed based on the theoretical foundation of (Courty et al., 2017)) and other OT-based DDA approaches, including SWD (Lee et al., 2019), DASPOT (Xie et al., 2019), ETD (Li et al., 2020) and RWOT (Xu et al., 2020) to demonstrate the capability of our proposed method.

2. Main Theoretical Results

2.1. Theoretical setting

Let the data spaces of the source and target domains be $\mathcal{X}^s$ and $\mathcal{X}^t$. These are endowed with data generation probability

distributions $\mathbb{P}^s$ and $\mathbb{P}^t$ with the densities $p^s(\mathbf{x})$ and $p^t(\mathbf{x})$ respectively. We also denote the probabilistic supervisor distributions that assign labels to data samples in the source and target domains by $p^s(y \mid \mathbf{x})$ and $p^t(y \mid \mathbf{x})$ (Vapnik, 1999). We consider the multi-class classification problem with the label set $\mathcal{Y} = \{1, 2, ..., C\}$.

Consider the hypothesis family on the source domain $\mathcal{H}^s := \{h^s : \mathcal{X}^s \to \Delta_C\}$, where $\Delta_C = \left\{\boldsymbol{\pi} \in \mathbb{R}^C : \|\boldsymbol{\pi}\|_1 = 1 \wedge \pi \geq \mathbf{0}\right\}$ is the C-simplex. Let $T : \mathcal{X}^t \to \mathcal{X}^s$ be a mapping.

The corresponding hypothesis family induced on the target domain via T is denoted as $\mathcal{H}^t := \{h^t : \mathcal{X}^t \to \Delta_C \mid h^t(\cdot) = h^s(T(\cdot)) \text{ for some } h^s \in \mathcal{H}^s\}$.

The intuition here is that with $\mathbf{x} \sim \mathbb{P}^t$, we apply the mapping T to reduce the difference between two domains and then use a hypothesis $h^s \in \mathcal{H}^s$ to predict the label of $\mathbf{x}$. This motivates us to seek the key properties of the transformation T in order to employ the hypothesis $h^t = h^s \circ T$ for accurately predicting labels of target data.

To formulate this, let $P^\# := T_\# \mathbb{P}^t$ be the push-forward distribution induced by transporting $\mathbb{P}^t$ via T, which consequently introduces a new domain, termed the *transport domain* having the density function $p^\#(\cdot)$ and probability distribution $\mathbb{P}^\#$. We further define the supervisor distribution for the transport domain as $p^\#(y \mid T(\mathbf{x})) = p^t(y \mid \mathbf{x})$ for any $\mathbf{x} \sim \mathbb{P}^t$. To ease the presentation, we denote the general expected loss:

$$R^{a,b}(h) := \int \ell(y, h(\mathbf{x})) p^b(y \mid \mathbf{x}) p^a(\mathbf{x}) dy d\mathbf{x},$$

where a, b are in the set $\{s, t, \#\}$ and $\ell(\cdot, \cdot)$ specifies a loss function. In addition, we shorten $R^{a,a}$ as R^a, and given a hypothesis $h^s \in \mathcal{H}^s$ and $h^t = h^s \circ T$, we measure the variance of general losses of h^s when predicting on the source domain and general losses of h^t when predicting on the target domain as:

$$\Delta R\left(h^s, h^t\right) := \left|R^t\left(h^t\right) - R^s\left(h^s\right)\right|.$$

We note that our theoretical setting is different from popular literature (Mansour et al., 2009; Ben-David et al., 2010; Redko et al., 2017; Zhang et al., 2019a; Cortes et al., 2019). By introducing the transformation T, we couple target examples and hypotheses with source examples and hypotheses which enables us to define the label shift explicitly.

2.2. Gap between target and source domains

To investigate the variance $\Delta R(h^s, h^t)$ and derive a relation between $R^t(h^t)$ and $R^s(h^s)$, we make the following assumptions w.r.t. loss function:

- (A.1) $M := \sup_{h^s \in \mathcal{H}^s, \mathbf{x} \in \mathcal{X}^s, y \in \mathcal{Y}} |\ell(y, h^s(\mathbf{x}))| < \infty$.
- (A.2) ℓ is a k-Lipschitz function w.r.t. a norm $\|\cdot\|$ over Δ_C, that is, $|\ell(y, \mathbf{a}) - \ell(y, \mathbf{b})| \leq k \|\mathbf{a} - \mathbf{b}\|$ for all $y \in \mathcal{Y}$ and $\mathbf{a}, \mathbf{b} \in \Delta_C$.

We note that these assumptions are easily satisfied when ℓ is a bounded loss, e.g., logistic or 0-1 loss, or when ℓ is any continuous loss, $\mathcal{X}^s$ is compact, and $\sup_{\mathbf{x} \in \mathcal{X}^s} |h^s(\mathbf{x})| < \infty$. Equipped with Assumption (A.1), we have the following key result to upper bound the gap $\Delta R(h^s, h^t)$:

Theorem 1. *Given Assumption (A.1), then for any hypothesis $h^s \in \mathcal{H}^s$, the following inequality holds:*

$$\Delta R\left(h^s, h^t\right) \leq M\left(\mathrm{W}_{c_{0/1}}\left(\mathbb{P}^s, \mathbb{P}^\#\right) + \mathbb{E}_{\mathbb{P}^t}\left[\|\Delta p(\cdot \mid \mathbf{x})\|_1\right]\right),$$

where $\Delta p(\cdot \mid \mathbf{x}) := \left\|\left[p^t(y = i \mid \mathbf{x}) - p^s(y = i \mid T(\mathbf{x}))\right]_{i=1}^C\right\|_1$, *and $W_{c_{0/1}}(\cdot, \cdot)$ is the Wasserstein distance with respect to the cost function $c_{0/1}(\mathbf{x}, \mathbf{x}') = \mathbf{1}_{\mathbf{x} \neq \mathbf{x}'}$, returning 1 if $\mathbf{x} \neq \mathbf{x}'$ and 0 otherwise.*

Remark 2. We have some observations in order.

- The quantity $\Delta p(\cdot \mid \mathbf{x})$ quantifies the label shift. Note that by coupling a target example $\mathbf{x}$ with a source example $T(\mathbf{x})$ using a transformation T, we can reasonably define and tackle the label shift as the divergence between $p^t(y \mid \mathbf{x})$ and $p^s(y \mid T(\mathbf{x}))$.
- In addition, when $\Delta p(\cdot \mid \mathbf{x}) = 0$ (i.e., $p^s(y \mid T(\mathbf{x})) = p^t(y \mid \mathbf{x})$ for $\mathbf{x} \sim \mathbb{P}^t$) and $W_{c_{0/1}}\left(\mathbb{P}^s, \mathbb{P}^\#\right) = 0$ (i.e., $T_\# \mathbb{P}^t = \mathbb{P}^s$), Theorem 1 shows that a perfect transfer learning without loss of performance can be achieved. Hence, if we can instrument a suitable mapping T, the adaptation is achievable.

To arrive at a stronger result presented in Theorem 3 below, we consider a Wasserstein distance between $\mathbb{P}^s$ and $\mathbb{P}^\#$ w.r.t. a ground metric c over $\mathcal{X}^s \times \mathcal{X}^s$ and $p \geq 1$ as

$$W_{c,p}\left(\mathbb{P}^s, \mathbb{P}^\#\right) = \inf_{\gamma \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^\#\right)} \mathbb{E}_{(\mathbf{x}_s, \mathbf{x}_\#) \sim \gamma}\left[c(\mathbf{x}_s, \mathbf{x}_\#)^p\right]^{1/p},$$

where $\gamma \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^\#\right)$ is a joint distribution admitting $\mathbb{P}^s, \mathbb{P}^\#$ as its marginals.

Furthermore, given a decreasing function $\phi : \mathbb{R} \to [0, 1]$, a hypothesis h^s is said to be ϕ-Lipschitz transferable (Courty et al., 2017) w.r.t. a joint distribution $\gamma \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^\#\right)$, the metric c, and the norm $\|\cdot\|$ if for all $\lambda > 0$, we have

$$\mathbb{P}_{(\mathbf{x}_s, \mathbf{x}_\#) \sim \gamma}\left[\|h^s(\mathbf{x}_s) - h^s(\mathbf{x}_\#)\| > \lambda c(\mathbf{x}_s, \mathbf{x}_\#)\right] \leq \phi(\lambda).$$

Theorem 3. *Assume that Assumptions (A.1) and (A2) hold, the hypothesis h^s satisfies ϕ-Lipschitz transferable w.r.t the optimal joint distribution (transport plan) $\gamma^* \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^\#\right)$, c and $\|\cdot\|$, the following inequality holds for all $\lambda > 0$:*

$$\Delta R\left(h^s, h^t\right) \leq M\left(\mathbb{E}_{\mathbb{P}^t}\left[\|\Delta p(\cdot \mid \mathbf{x})\|_1\right] + 2\phi(\lambda)\right) + kC\lambda \mathrm{W}_{c,p}\left(\mathbb{P}^s, \mathbb{P}^\#\right).$$

Detailed proofs and further technical descriptions are given in the supplementary material.

2.3. Data shift via Wasserstein metric

Theorems 1 and 3 suggest that we need to construct a map that transports the target to source distributions and makes two supervisor distributions identical via this map for a perfect transfer learning. This is consistent with what is achieved in Theorem 1 for which the upper bound of the loss variance $\Delta R\left(h^s, h^t\right)$ vanishes.

In particular, the upper bounds in Theorems 1 and 3 consist of two terms: the first term (i.e., $W_{c,p}\left(\mathbb{P}^s, \mathbb{P}^{\#}\right)$) quantifies the *data shift*, while the second term (i.e., $\mathbb{E}_{\mathbb{P}^t}\left[\left\|\Delta p\left(y \mid \mathbf{x}\right)\right\|_1\right]$) reflects the *label shift*. Our strategy is then to find the best hypothesis h^s_* by minimizing the general loss $R^s\left(h^s\right)$, and the optimal transformation T^* by minimizing $W_{c,p}\left(\mathbb{P}^s, \mathbb{P}^{\#}\right)$ and $\mathbb{E}_{\mathbb{P}^t}\left[\left\|\Delta p\left(y \mid \mathbf{x}\right)\right\|_1\right]$.

Due to the lack of target labels, we focus on minimizing the first term $W_{c,p}\left(\mathbb{P}^s, \mathbb{P}^{\#}\right)$ by answering the following question: *among the transformations T that transport the target to source distributions, which transformation incurs the minimal label shift* $\mathbb{E}_{\mathbb{P}^t}\left[\left\|\Delta p\left(y \mid \mathbf{x}\right)\right\|_1\right] = \left\|\left[p^t\left(y=i \mid \mathbf{x}\right) - p^s\left(y=i \mid T\left(\mathbf{x}\right)\right)\right]_{i=1}^{C}\right\|$? Given the ground metric c and $p \geq 1$, this is formulated as:

$$\min_{T} W_{c,p}\left(T_{\#}\mathbb{P}^t, \mathbb{P}^s\right). \tag{1}$$

Let $\mathcal{Z}$ be an intermediate space (i.e., the latent space $\mathcal{Z} = \mathbb{R}^m$). We consider the composite mapping : $T\left(\mathbf{x}\right) = T^2\left(T^1\left(\mathbf{x}\right)\right)$ where T^1 is a mapping from the target domain $\mathcal{X}^t$ to the latent space $\mathcal{Z}$ and T^2 maps from the latent space $\mathcal{Z}$ to the source domain $\mathcal{X}^s$ (note that if $\mathcal{Z} = \mathcal{X}^s$ then $T^2 = id$ is the identity function). The optimization problem (OP) in (1) becomes:

$$\min_{T^1,T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#}\mathbb{P}^t, \mathbb{P}^s\right). \tag{2}$$

In the following theorem, we show that the above OP can be transformed into another form involving the latent space (see Figure 1 for an illustration of that theorem).

Theorem 4. *The optimal objective value of the OP (2) is equal to that of the OP (3), that is*

$$\min_{T^1,T^2} \mathrm{W}_{c,p}\left(\left(T^2 \circ T^1\right)_{\#}\mathbb{P}^t, \mathbb{P}^s\right) = \min_{T^1,T^2} \min_{G^1: T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} \tag{3}$$

where G^1 is a map from $\mathcal{X}^s$ to $\mathcal{Z}$.

We can interpret G^1 and T^1 as two generators that map the source and target domains to the common latent space $\mathcal{Z}$.

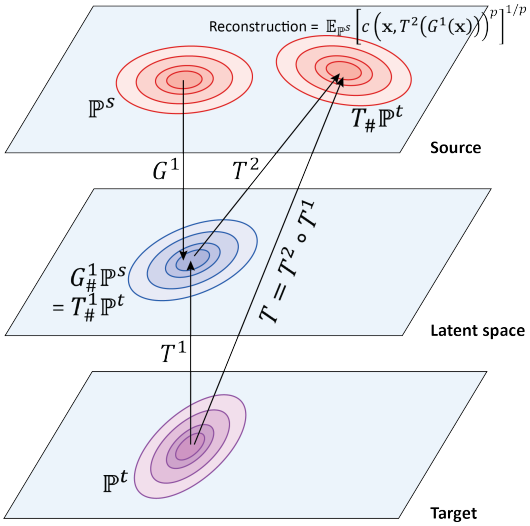

Figure 1. $T = T^2 \circ T^1$ maps from the target to source domains. We minimize $D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$ to close the discrepancy gap of the source and target domains on the latent space and minimize the reconstruction terms to avoid the mode collapse.

The constraint $T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s$ enforces the gap between the source and target distributions to be closed in the latent space. Furthermore, T^2 maps from the latent space to the source domain and aims to reconstruct G^1. Similar to (Tolstikhin et al., 2018), we do a relaxation and arrive at

$$\min_{T^1,T^2,G^1}\left(\mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} + \alpha D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)\right), \tag{4}$$

where $D\left(\cdot,\cdot\right)$ specifies a divergence between two distributions over the latent space and $\alpha > 0$. When α approaches $+\infty$, the solution of the relaxation problem in Eq. (4) approaches the optimal solution in Eq. (3).

Let $\mathcal{D}^s = \left\{\left(\mathbf{x}^s_1, y_1\right), ..., \left(\mathbf{x}^s_{N_s}, y_{N_s}\right)\right\}$, to enable the transfer learning, we can train a supervised classifier $\mathcal{A}$ on $G^1\left(\mathcal{D}^s\right) = \left\{\left(G^1\left(\mathbf{x}^s_1\right), y_1\right), \ldots, \left(G^1\left(\mathbf{x}^s_{N_s}\right), y_{N_s}\right)\right\}$. Our final OP becomes

$$\min_{T^1,T^2,G^1}\left(\beta\mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} + \alpha D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right) + \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^s}\left[\ell\left(y, \mathcal{A}\left(G^1\left(\mathbf{x}\right)\right)\right)\right]\right), \tag{5}$$

where $\beta > 0$ and we overload $\mathcal{D}^s$ to represent the empirical distribution over the source training set. Moreover, to reduce the discrepancy gap $D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$ in Eq. (5), one can use the adversarial learning framework (Goodfellow et al., 2014) to implicitly minimize a Jensen-Shannon (JS) divergence or explicitly minimize other divergences and distances (e.g., a maximum mean discrepancy (Gretton et al., 2007) or WS distance). Note that if we employ a

JS divergence or $f-$divergence for D, the OP in (5) can be further rewritten in a min-max form (Goodfellow et al., 2014; Nowozin et al., 2016).

It is also worth mentioning that with regard to the latent space and the above equipment for $T = T^2 \circ T^1$, we have the following formulations for the source classifier (i.e., h^s) and target classifier (i.e., h^t) now become:

$$h^s(\mathbf{x}) = \mathcal{A}\left(G^1(\mathbf{x})\right) \text{ and } h^t(\mathbf{x}) = \mathcal{A}\left(G^1(T(\mathbf{x}))\right). \quad (6)$$

2.4. Label shift via Wasserstein metric

Since G^1 and T^1 are two mappings from the source and target domains to the latent space, we can further define the source and target supervisor distributions on the latent space as $p^{\#,s}\left(y \mid G^1(\mathbf{x})\right) = p^s(y \mid \mathbf{x})$ for $\mathbf{x} \sim \mathbb{P}^s$ and $p^{\#,t}\left(y \mid T^1(\mathbf{x})\right) = p^t(y \mid \mathbf{x})$ for $\mathbf{x} \sim \mathbb{P}^t$. With respect to the latent space, the second term of the upper bound in Theorem 1 can be rewritten as in the following corollary.

Corollary 5. *The second term of the upper bound in Theorem 1 can be rewritten as*

$$\mathbb{E}_{\mathbb{P}^t}\left[\left\|p^{\#,s}\left(\cdot \mid G^1\left(T^2\left(T^1(\mathbf{x})\right)\right)\right) - p^{\#,t}\left(\cdot \mid T^1(\mathbf{x})\right)\right\|_1\right]. \quad (7)$$

We now analyze the ideal scenario to formulate the data distribution and label shifts in the latent space $\mathcal{Z}$. For sufficiently powerful G^1 and T^1, the OP in (3) peaks its minimization at 0 when $G^1_{\#}\mathbb{P}^s = T^1_{\#}\mathbb{P}^t$ and $G^1 \circ T^2 = id$ (i.e., the identity function), which further implies that

$$\begin{aligned} T_{\#}\mathbb{P}^t &= T^2_{\#}\left(T^1_{\#}\mathbb{P}^t\right) = T^2_{\#}\left(G^1_{\#}\mathbb{P}^s\right) \\ &= \left(G^1 \circ T^2\right)_{\#}\mathbb{P}^s = \mathbb{P}^s, \end{aligned}$$

and $W_{c,p}\left(T_{\#}\mathbb{P}^t, \mathbb{P}^s\right) = 0$. Under that ideal scenario, the label mismatch term in Eq. (7) reduces to

$$\mathbb{E}_{\mathbb{P}^t}\left[\left\|p^{\#,s}\left(\cdot \mid T^1(\mathbf{x})\right) - p^{\#,t}\left(\cdot \mid T^1(\mathbf{x})\right)\right\|_1\right]. \quad (8)$$

We note that because $G^1_{\#}\mathbb{P}^s = T^1_{\#}\mathbb{P}^t$, $T^1(\mathbf{x})$ with $\mathbf{x} \sim \mathbb{P}^t$ is moved to a source class region on the latent space (e.g., the region of class y^s). This sample would be classified to class y^s by a source classifier (i.e., the one that mimics $p^{\#,s}\left(y \mid T^1(\mathbf{x})\right)$). Assume that $\mathbf{x}$ has the ground-truth label y^t, minimizing the label mismatch term in Eq. (8) suggests $y^s = y^t$. In other words, T^1 should transport $\mathbf{x}$ to the proper class region to reduce the label mismatch. Moreover, in unsupervised DA, since target labels are lacking and the neural network generator T^1 can be sufficiently powerful to map a target class region to a wrong source one on the latent space (cf. Figure 2), it is almost impossible to tackle perfectly the label mismatch.

Aligned with (Zhao et al., 2019), the label mismatch term in (7) can be lower-bounded by a divergence between the marginal label distributions of the source and target domains as shown in Corollary 6.

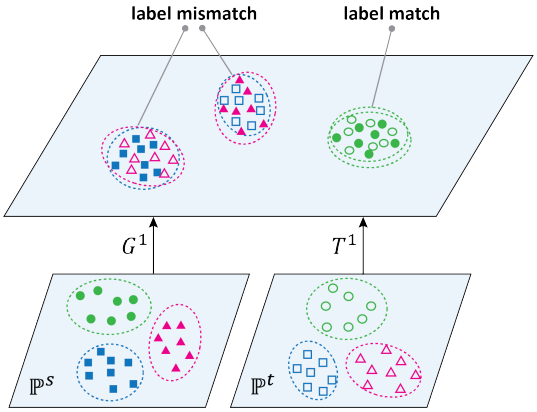

Figure 2. Label match and mismatch on the latent space.

Corollary 6. *Under the ideal scenario, the label mismatch term in (7) has a lower-bound*

$$\left\|\left[p^s(y=i) - p^t(y=i)\right]_{i=1}^C\right\|_1.$$

Under the light of Corollary 6, we find that when pushing $G^1_{\#}\mathbb{P}^s$ to $T^1_{\#}\mathbb{P}^t$ by minimizing $W_{c,p}\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$, the label mismatch term in (8) tends to become higher than the L1 distance between the marginal label distributions of source and target domains. Therefore, if the marginal label distributions of source and target domains (i.e., $p^s(y)$ and $p^t(y)$) are significantly divergent, learning domain invariant representations on a latent space can cause more label shift. To strengthen this observation, we develop a theorem to directly offer an upper bound for the L1 distance between the label marginal distributions. To this end, we define a new metric $\tilde{c}$ w.r.t. the family $\mathcal{H}^a$ of the classifier $\mathcal{A}$ in the OP (5) as:

$$\tilde{c}(\mathbf{z}_1, \mathbf{z}_2) = \sup_{\mathcal{A} \in \mathcal{H}^a} \left\|\mathcal{A}(\mathbf{z}_1) - \mathcal{A}(\mathbf{z}_2)\right\|_1,$$

where $\mathbf{z}_1$ and $\mathbf{z}_2$ lie on the latent space. The following lemma states under which conditions, $\tilde{c}$ is a proper metric on the latent space.

Lemma 7. *For any $\mathbf{z}_1$ and $\mathbf{z}_2$, if $\mathcal{A}(\mathbf{z}_1) = \mathcal{A}(\mathbf{z}_2), \forall \mathcal{A} \in \mathcal{H}^a$ leads to $\mathbf{z}_1 = \mathbf{z}_2$, $\tilde{c}$ is a proper metric.*

It turns out that the necessary (also sufficient) condition in Lemma 7 is realistic and not hard to be satisfied (e.g., the family $\mathcal{H}^a$ contains any bijection). We now can define a WS distance $W_{\tilde{c},p}$ that involves in the following theorem.

Theorem 8. *If $\tilde{c}$ is a proper metric and $p \geq 1$, the quantity $\left\|\left[p^s(y=i) - p^t(y=i)\right]_{i=1}^C\right\|_1$ has the upper-bounds:*

i) $R^s_1(h^s) + R^t_1(h^t) + W_{\tilde{c},p}\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$ *if* $h^s := \mathcal{A}\left(G^1(\mathbf{x})\right)$ *and* $h^t := \mathcal{A}\left(T^1(\mathbf{x})\right)$.

ii) $R^s_1(h^s) + R^t_1(h^t) + W_{\tilde{c},p}\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right) +$

$W_{\tilde{c},p}\left(L_{\#}\mathbb{P}^t, T^1_{\#}\mathbb{P}^t\right)$ *where* $L := T \circ G^1$, *and* h^s *and* h^t *are defined in (6).*

Here $R^s_1(h^s) := \int \|p^s(\cdot \mid \mathbf{x}) - h^s(\mathbf{x})\|_1 p^s(\mathbf{x})\, d\mathbf{x}$ *and* $R^t_1(h^t) := \int \|p^t(\cdot \mid \mathbf{x}) - h^t(\mathbf{x})\|_1 p^t(\mathbf{x})\, d\mathbf{x}$ *are the general losses of* h^s *and* h^t *w.r.t.* $\|\cdot\|_1$.

Remark 9. Theorem 8 reveals that if the marginal label distributions are significantly different between the source and target domains, forcing $W_{\tilde{c},p}\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$ to be smaller increases $R^s_1(h^s) + R^t_1(h^t)$, which directly hurts the predictive performance of the target classifier h^t. The reason is that $R^s_1(h^s)$ would be small since it is trained on labeled source domain. Similar significant theoretical result was discovered in (Zhao et al., 2019) (see Theorem 4.9 in that paper). However, our theory is developed in a more general context of multi-class classification and uses the WS distance rather than the JS distance (Endres & Schindelin, 2006) as in (Zhao et al., 2019). In addition, the advantages of WS distance over JS distance including its numerical stability and continuity have been thoughtfully discussed in Arjovsky et al. (2017). Finally, our Theorem 8 can be generalized to any metric on the simplex Δ_C (e.g., a Wasserstein distance).

3. Label Matching Domain Adaptation

As pointed by our theory and ablation study (see our supplementary material), reducing label mismatch in the joint space when bridging $D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$ (cf. Eq. (4)) between the source and target domains in this space is a key factor to improve the predictive performance of deep unsupervised domain adaptation. Existing approaches (Ganin & Lempitsky, 2015; Tzeng et al., 2015; Long et al., 2015; French et al., 2018) use a binary discriminator to guide target samples for moving to source samples in the joint space.

However, a binary discriminator is only able to distinguish the entire source domain from the target domain, hence cannot elegantly guide target samples moving to the most suitable class in the source domain. Our idea is to increase the resolution of discriminators by utilizing a multi-class discriminator d that can simultaneously (i) distinguish source and target domains and (ii) emphasize the class regions in the source domain.

With the assistance of a multi-class discriminator d, we hope to guide target samples to a suitable class in the source domain. In addition, in conjunction with the multi-class discriminator d, we propose minimizing an optimal transport inspired cost which leverages the class information provided by the multi-class discriminator d for guiding target samples more accurately. We name the proposed method as LAbel Matching Domain Adaptation (LAMDA).

To minimize the discrepancy $D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$, we employ the adversarial learning principle (Goodfellow et al., 2014) with the support of the multi-class discriminator d. Moreover, to simultaneously discriminate the source and target samples and distinguish the classes of the source domain, we use a multi-class discriminator d with $C+1$ probability outputs (C is the number of classes) in which for $\mathbf{x} \sim \mathbb{P}^s$ and $1 \leq i \leq C$, the i-th probability output specifies the probability of that example generated from the i-th class mixture of the source domain, i.e., $d_i\left(G^1(\mathbf{x})\right) = \mathbb{P}(y = i \mid \mathbf{x})$ and for $\mathbf{x} \sim \mathbb{P}^t$, the $C+1$ probability output specifies the probability of that example generated from the target distribution, i.e., $d_{C+1}\left(T^1(\mathbf{x})\right) = \mathbb{P}(y = C+1 \mid \mathbf{x})$.

Training method. Since the discriminator can discriminate the source and target samples and distinguish the classes of the source domain, we solve the following OP for d:

$$\max_d \Bigg(\mathcal{L}_d := \sum_{i=1}^{C} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^s \wedge y=i}\left[\log d_i\left(G^1(\mathbf{x})\right)\right] + \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[\log d_{C+1}\left(T^1(\mathbf{x})\right)\right] + \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[\log\left(1 - d_{C+1}\left(G^1(\mathbf{x})\right)\right)\right] \Bigg). \tag{9}$$

To train the generators G^1, T^1, we update them as follows:

i) We move $G^1(\mathbf{x})$ for $\mathbf{x} \sim \mathbb{P}^s$ to the region of high values for $d_{C+1}(\cdot)$ (i.e., the region of target samples) by minimizing

$$I\left(G^1\right) := \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[\log\left(1 - d_{C+1}\left(G^1(\mathbf{x})\right)\right)\right].$$

ii) We move $T^1(\mathbf{x})$ for $\mathbf{x} \sim \mathbb{P}^t$ to one of class regions in the source domain accordingly. Recalling that $d_i(\mathbf{x})$ represents the likelihood of $\mathbf{x}$ w.r.t. the i-th source class region, we employ $-\log \mathbb{P}(y = i \mid \mathbf{x}) = -\log d_i\left(T^1(\mathbf{x})\right)$ as the cost incurred if we move $T^1(\mathbf{x})$ to $\mathcal{D}^s_i = \{(x, y) \in \mathcal{D}^s \mid y = i\}$.

To specify the probabilities that transports $\mathbf{x} \sim \mathbb{P}^t$ to the source class regions, we use a transportation probability network $S(\mathbf{x})$ for which $S_i(\mathbf{x})$ points out probability to transport $\mathbf{x}$ to $\mathcal{D}^s_i$. Therefore, the total transport cost incurred is

$$TC\left(T^1\right) := \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[-\sum_{i=1}^{C} S_i(\mathbf{x}) \log d_i\left(T^1(\mathbf{x})\right)\right]. \tag{10}$$

In addition, we push $T^1(\mathbf{x})$ for $\mathbf{x} \sim \mathbb{P}^t$ to the the region of low values for $d_{C+1}(\cdot)$ (i.e., the region of source samples) by minimizing

$$J\left(T^1\right) := \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[\log d_{C+1}\left(T^1(\mathbf{x})\right)\right].$$

Moreover, we need to minimize the loss on the source domain

$$\mathcal{L}_{\mathcal{A}} := \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^s}\left[\ell\left(y, \mathcal{A}\left(G^1(\mathbf{x})\right)\right)\right],$$

and the reconstruction term defined as

$$R\left(T^2, G^1\right) := \mathbb{E}_{\mathbb{P}^s}\left[\left\|T^2\left(G^1\left(\mathbf{x}\right)\right) - \mathbf{x}\right\|_2^2\right].$$

Putting all above losses together, the OP to update G^1, T^1, T^2, and $\mathcal{A}$ has the following form:

$$\min_{G^1,T^1,T^2,\mathcal{A}} \mathcal{L}_g, \tag{11}$$

where we have defined

$$\mathcal{L}_g := I\left(G^1\right) + J\left(T^1\right) + \alpha TC\left(T^1\right) + \beta R\left(T^2, G^1\right) + \mathcal{L}_{\mathcal{A}}.$$

The min-max OP of our LAMDA has the following form:

$$\max_{d} \min_{G^1,T^1,T^2,\mathcal{A}} \Bigg(I\left(G^1\right) + J\left(T^1\right) + K\left(d, T^1\right) + \beta R\left(T^2, G^1\right) + \mathcal{L}_{\mathcal{A}} \Bigg), \tag{12}$$

where the term $K\left(d, T^1\right)$ is the first term of $\mathcal{L}_d$ as in (9) for the outer max and $\alpha TC\left(T^1\right)$ as in (10) for the inner min.

It is worth noting that although the min-max problem in (12) is not mathematically rigorous, we still present it to increase the comprehensibility of our LAMDA. In addition, to reduce the model complexity, we share S and $\mathcal{A}$ because the source classification $\mathcal{A}$ can also characterize the source class regions. Finally, the pseudocode for training LAMDA is presented in Algorithm 1.

Algorithm 1 Pseudocode for training LAMDA.

Input: Source $\mathcal{D}^s = \{(\mathbf{x}_k^s, y_k^s)\}_{k=1}^{N_s}$, target $\mathcal{D}^t = \{\mathbf{x}_l^t\}_{l=1}^{N_t}$.
Output: Generator G^{1*}, classifier $\mathcal{A}^*$.

1: **for** number of training iterations **do**
2: Sample minibatch of source $\{(\mathbf{x}_k^s, y_k^s)\}_{k=1}^m$ and target $\{\mathbf{x}_l^t\}_{l=1}^m$.
3: Update d according to Eq. (9).
4: Update G^1, T^1, T^2 and $\mathcal{A}$ according to Eq. (11).
5: **end for**

4. Experiment

4.1. Ablation Study

We start with the ablation study of the effect of the terms in LAMDA especially the reconstruction term $\beta R\left(T^2, G^1\right)$. At the outset, we notice that akin to other DDA works, we share two generators G^1 and T^1 (i.e., $G^1 = T^1 = G$).

4.1.1. The Effect of Reconstruction Term

We conduct the experiments on the three pairs of *Office-31* as shown in Figure 3 (left). The experiments on the *Office-31* use ResNet-50 (He et al., 2016) as a backbone to extract the features. The representations of ResNet-50 are fed to the latent space using a dense layer and on the top of this dense layer, we have another dense layer to connect the latent and the output layers. We employ the reconstruction term to reconstruct the output representations of ResNet-50 from the latent representations (i.e., the output of ResNet-50 $\rightarrow$ latent representation $\rightarrow$ output of ResNet-50). Note that we do not fine-tune the base ResNet-50. We vary β in $\{0, 0.05, 0.1, 0.2, 0.5, 1.0, 5.0\}$ and observe the target test accuracies. As shown in Figure 3, the reconstruction term slightly affects the final performance. Therefore, in our experiments on real-world datasets, we set $\beta = 0$ to reduce the training cost.

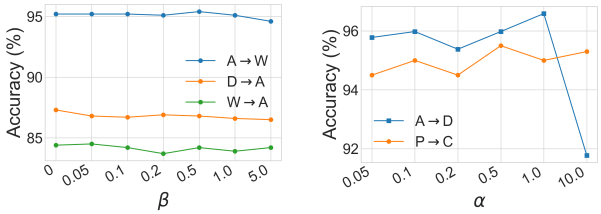

Figure 3. The effect of the reconstruction term (left) and the total transport cost (right).

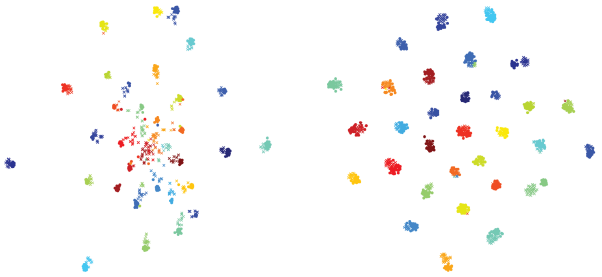

Figure 4. The t-SNE visualization of the transfer task **A**$\rightarrow$**D** with label and domain information. Each color denotes a class while the circle and cross markers represent the source and target data.

4.1.2. The Effect of the Total Transport Cost

We vary the trade-off parameter α of the total transport cost to inspect its effect on the final performance as shown in Figure 3 (right). We empirically find that the appropriate range for α is $[0.1, 0.5]$. In our experiments on the real-world datasets, we set $\alpha = 0.5$.

4.1.3. The Effect of the Multi-class Discriminator

We conduct an ablation study on the Office-Home dataset with the ResNet-50 features in which we relax the *multi-class discriminator* by a *binary discriminator*. As shown in Table 2, the experimental results show that our LAMDA with the multi-class discriminator and the total transport cost term (i.e., $TC\left(T^1\right)$ in Eq. (10)) outperforms its binary discriminator relaxation.

Table 1. Classification accuracy (%) on Office-Home dataset using ResNet-50 features.

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
DANN (Ganin & Lempitsky, 2015)	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
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
CDAN (Long et al., 2018)	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
CDAN+TransNorm (Wang et al., 2019)	50.2	71.4	77.4	59.3	72.7	73.1	61.0	53.1	79.5	71.9	59.0	82.9	67.6
TPN (Pan et al., 2019)	51.2	71.2	76.0	65.1	72.9	72.8	55.4	48.9	76.5	70.9	53.4	80.4	66.2
MDD (Zhang et al., 2019a)	54.9	73.7	77.8	60.0	71.4	71.8	61.2	53.6	78.1	72.5	**60.2**	82.3	68.1
MDD+Implicit Alignment (Jiang et al., 2020)	56.2	77.9	79.2	64.4	73.1	74.4	64.2	54.2	79.9	71.2	58.1	83.1	69.5
DeepJDOT (Damodaran et al., 2018)	48.2	69.2	74.5	58.5	69.1	71.1	56.3	46.0	76.5	68.0	52.7	80.9	64.3
SHOT (Liang et al., 2020)	57.1	78.1	81.5	**68.0**	78.2	78.1	**67.4**	54.9	82.2	**73.3**	58.8	**84.3**	71.8
ETD (Li et al., 2020)	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
RWOT (Xu et al., 2020)	55.2	72.5	78.0	63.5	72.5	75.1	60.2	48.5	78.9	69.8	54.8	82.5	67.6
LAMDA	**57.2**	**78.4**	**82.6**	66.1	**80.2**	**81.2**	65.6	**55.1**	**82.8**	71.6	59.2	83.9	**72.0**

Table 2. Performance comparison between two settings on the Office-Home dataset using the ResNet-50 features.

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**
Binary discriminator	51.4	78.2	76.3	**66.1**	74.3	78.9	64.5	47.0	82.3	69.9	53.1	82.6	68.7
Multi-class discriminator	**57.2**	**78.4**	**82.6**	**66.1**	**80.2**	**81.2**	**65.6**	**55.1**	**82.8**	**71.6**	**59.2**	**83.9**	**72.0**

4.1.4. Feature visualization

We visualize the features of ResNet-50 and our method on the transfer task **A**→**D** (*Office-31*) by *t*-SNE (van der Maaten & Hinton, 2008) in Figure 4. The sub-figure (a) show that ResNet-50 classifies quite well on the source domain (**A**) but poorly on the target domain (**D**), while the representation in sub-figure (b) is generated by our method with better alignment. LAMDA achieves exactly 31 clusters corresponding to 31 classes of *Office-31*, which represents the ability of reducing not only the data shift but also the label shift between two domains.

4.2. Our LAMDA Versus the Baselines

4.2.1. Experimental datasets

We conduct the experiments to compare our LAMDA against the state-of-the-art baselines on the *digit*, *traffic sign*, *natural scene*, *Office-Home*, *Office-31*, and *ImageCLEF-DA* datasets. In addition to the baselines in general DDA, we also compare our LAMDA to the ones developed based on the OT theory including DeepJDOT (Damodaran et al., 2018), SWD (Lee et al., 2019), DASPOT (Xie et al., 2019), ETD (Li et al., 2020) and RWOT (Xu et al., 2020). Moreover, we resize the resolution of each sample in *digits*, *traffic sign*, and *natural image* datasets to 32×32, and normalize the value of each pixel to the range of $[-1, 1]$. For *object recognition* datasets, we use features have 2048 dimensions extracted from ResNet-50 (He et al., 2016) pretrained on ImageNet.

Digit datasets.
MNIST. To adapt from MNIST to MNIST-M or SVHN, the MNIST images are replicated from single greyscale channel to obtain digit images which has three channels.

MNIST-M. Following Ganin & Lempitsky (2015), we generate the MNIST-M images by replacing the black background of MNIST images by the color ones.

SVHN. The dataset consists of images obtained by detecting house numbers from Google Street View images. This dataset is a benchmark for recognizing digits and numbers in real-world images.

DIGITS. There are roughly 500,000 images are generated using various data augmentation schemes, i.e., varying the text, positioning, orientation, background, stroke color, and the amount of blur.

Traffic sign datasets.
SIGNS. A synthetic dataset for traffic sign recognition. Images are collected from Wikipedia and then applied various types of transformations to generate 100,000 images for training and test.

GTSRB. Road sign images are extracted from videos recorded on different road types in Germany.

Natural scene datasets.
CIFAR. This dataset includes 50,000 training images and 10,000 test images. However, to adapt with STL dataset, we base on French et al. (2018) to remove one non-overlapping class ("frog").

STL. Similar to CIFAR-10, we remove class named "monkey" to obtain a 9-class classification problem.

Object recognition datasets.
Office-Home. This dataset consists of roughly 15,500 images in a total of 65 object classes belonging to 4 different domains: Artistic (**Ar**), Clip Art (**Cl**), Product (**Pr**) and Real-world (**Rw**). Due to the shortage of data, this dataset is much more challenging for the domain adaptation task.

Office-31. This is a popular dataset for domain adaptation that contains 3 domains Amazon (**A**), Webcam (**W**), and DSLR (**D**). There are 31 common classes and 4,110 images in total.

ImageCLEF-DA. This dataset contains three domains:

Caltech-256 (**C**), ImageNet ILSVRC 2012 (**I**) and Pascal VOC 2012 (**P**). We follow the work in Li et al. (2020) to evaluate 6 adaptation tasks.

Table 3. Classification accuracy (%) on digits and natural image datasets.

Source	MNIST	USPS	MNIST	SVHN	MNIST	CIFAR	STL
Target	USPS	MNIST	MNIST-M	MNIST	SVHN	STL	CIFAR
MMD (Long et al., 2015)	-	-	76.9	71.1	-	-	-
DANN (Ganin & Lempitsky, 2015)	-	-	81.5	71.1	35.7	-	-
DRCN (Ghifary et al., 2016)	-	-	-	82.0	40.1	66.4	-
DSN (Bousmalis et al., 2016)	-	-	83.2	82.7	-	-	-
ATT (Saito et al., 2017)	-	-	94.2	86.2	52.8	-	-
Π-model (French et al., 2018)	-	-	-	92.0	71.4	76.3	64.2
CyCADA (Hoffman et al., 2018)	95.6	96.5	-	90.4	-	-	-
MSTN (Xie et al., 2018)	92.9	97.6	-	91.7	-	-	-
CDAN (Long et al., 2018)	95.6	98.0	-	89.2	-	-	-
MCD (Saito et al., 2018)	94.2	94.1	-	96.2	-	-	-
GTA (Sankaranarayanan et al., 2018)	90.8	95.3	-	92.4	-	-	-
DEV (You et al., 2019)	92.5	96.9	-	93.2	-	-	-
LDVA (Zhu et al., 2019)	98.8	96.8	-	95.2	-	-	-
DeepJDOT(Damodaran et al., 2018)	95.7	96.4	92.4	96.7	-	-	-
DASPOT (Xie et al., 2019)	97.5	96.5	94.9	96.2	-	-	-
SWD (Lee et al., 2019)	98.1	97.1	90.9	98.9	-	-	-
rRevGrad+CAT (Deng et al., 2019)	94.0	96.0	-	98.8	-	-	-
SHOT (Liang et al., 2020)	98.0	**98.4**	-	98.9	-	-	-
RWOT (Xu et al., 2020)	98.5	97.5	-	98.8	-	-	-
LAMDA	**99.5**	98.3	**98.4**	**99.5**	**82.1**	**78.0**	**71.6**

4.2.2. Hyper-parameter setting

We apply Adam Optimizer ($\beta_1 = 0.5$,$\beta_2 = 0.999$) with the learning rate 0.001 *digits*, *traffic sign* and *natural scene* datasets, whereas 0.0001 is the learning rate for *object recognition* datasets. All experiements was trained for 20000 iterations on *Office-31*, *Office-Home*, and *ImageCLEF-DA* and 80000 for the other datasets. The batch size for each dataset is set to 128. We set $\beta = 0$, $\alpha = 0.5$ as described in the ablation study, and γ is searched in $\{0.1, 0.5\}$. We implement our LAMDA in Python (version 3.5) using Tensorflow (version 1.9.0) (Abadi et al., 2016) and run our experiments on a computer with a CPU named Intel Xeon Processor E5-1660 which has 8 cores at 3.0 GHz and 128 GB of RAM, and a GPU called NVIDIA GeForce GTX Titan X with 12 GB memory. For *Office-Home*, *Office-31*, and *ImageCLEF-DA*, we use ResNet-50 as a feature extractor (He et al., 2016). Finally, the further network architecture detail can be found in the supplementary material.

4.2.3. Experimental results

As consistently shown in Tables 1, 3, 4, and 5, our LAMDA outperforms the baselines on the average performances and achieves good performances on the individual pairs. In particular, for the Digit datasets, although the transfer task MNIST→SVHN is extremely challenging in which the source dataset includes grayscale handwritten digits whereas the target dataset is created by real-world digits, our LAMDA is still capable of matching the gap between source and target domains and outperforms the second-best method by a significant margin (10.7%). Evidently, the fact our LAMDA achieves superior performances comparing to the baselines demonstrates that it can efficiently reduce the label mismatch on the latent space.

Table 4. Classification accuracy (%) on Office-31 dataset using either ResNet-50 features or ResNet-50 based deep models.

Method	A→W	A→D	D→W	W→D	D→A	W→A	**Avg**
ResNet-50 (He et al., 2016)	70.0	65.5	96.1	99.3	62.8	60.5	75.7
DeepCORAL (Sun & Saenko, 2016)	83.0	71.5	97.9	98.0	63.7	64.5	79.8
DANN (Ganin et al., 2016)	81.5	74.3	97.1	99.6	65.5	63.2	80.2
RTN (Long et al., 2016)	84.5	77.5	96.8	99.4	66.2	64.8	81.6
ADDA (Tzeng et al., 2017)	86.2	78.8	96.8	99.1	69.5	68.5	83.2
iCAN (Zhang et al., 2018)	92.5	90.1	98.8	**100.0**	72.1	69.9	87.2
CDAN (Long et al., 2018)	94.1	92.9	98.6	**100.0**	71.0	69.3	87.7
GTA (Sankaranarayanan et al., 2018)	89.5	87.7	97.9	99.8	72.8	71.4	86.5
DEV (You et al., 2019)	93.2	92.8	98.4	**100.0**	70.9	71.2	87.8
TPN (Pan et al., 2019)	91.2	89.9	97.7	99.5	70.5	73.5	87.1
MDD (Zhang et al., 2019a)	94.5	93.5	98.4	**100.0**	74.6	72.2	88.9
MDD+Implicit Alignment (Jiang et al., 2020)	90.3	92.1	98.7	99.8	75.3	74.9	88.8
SPL (Wang & Breckon, 2020)	92.7	93.0	98.7	99.8	76.4	76.8	89.6
DeepJDOT (Damodaran et al., 2018)	88.9	88.2	98.5	99.6	72.1	70.1	86.2
SHOT (Liang et al., 2020)	90.1	94.0	98.4	99.9	74.7	74.3	88.6
ETD (Li et al., 2020)	92.1	88.0	**100.0**	**100.0**	71.0	67.8	86.2
RWOT (Xu et al., 2020)	95.1	94.5	99.5	**100.0**	77.5	77.9	90.8
LAMDA	**95.2**	**96.0**	98.5	**100.0**	**87.3**	**84.4**	**93.0**

Table 5. Classification accuracy (%) on ImageCLEF-DA dataset using ResNet-50 features.

Method	I→P	P→I	I→C	C→I	C→P	P→C	**Avg**
ResNet-50 (He et al., 2016)	74.8	83.9	91.5	78.0	65.5	91.2	80.7
DeepCORAL (Sun & Saenko, 2016)	75.1	85.5	92.0	85.5	69.0	91.7	83.1
RTN (Long et al., 2016)	75.6	86.8	95.3	86.9	72.7	92.2	84.9
DANN (Ganin et al., 2016)	75.0	86.0	96.2	87.0	74.3	91.5	85.0
ADDA (Tzeng et al., 2017)	75.5	88.2	96.5	89.1	75.1	92.0	86.0
iCAN (Zhang et al., 2018)	79.5	89.7	94.7	89.9	78.5	92.0	87.4
CDAN (Long et al., 2018)	77.7	90.7	97.7	91.3	74.2	94.3	87.7
CDAN+TransNorm (Wang et al., 2019)	78.3	90.8	96.7	92.3	78.0	94.8	88.5
TPN (Pan et al., 2019)	78.2	92.1	96.1	90.8	76.2	95.1	88.1
CADA-P (Kurmi et al., 2019)	78.0	90.5	96.7	92.0	77.2	95.5	88.3
SymNets (Zhang et al., 2019b)	80.2	93.6	97.0	93.4	78.7	96.4	89.9
DeepJDOT (Damodaran et al., 2018)	77.5	90.5	95.0	88.3	74.9	94.2	86.7
ETD (Li et al., 2020)	81.0	91.7	**97.9**	93.3	79.5	95.0	89.7
RWOT (Xu et al., 2020)	**81.3**	92.9	**97.9**	92.7	79.1	**96.5**	90.0
LAMDA	80.7	**95.0**	96.7	**95.0**	**80.7**	95.8	**90.6**

5. Conclusion

Deep domain adaptation is a recent powerful learning framework that aims to address the problem of scarcity of qualified labeled data for supervised learning. The key ingredient is to learn domain invariant representations, which obviously can address the data shift issue. However, the label shift issue is significantly challenging to define and tackle. In this paper, we propose a new theory setting that allows us to couple the source and target hypotheses for explicitly defining the label shift. We further develop a theory to show the link between minimizing the WS distance for the data shift and bridging the gap between source and target domains on a latent space. In addition, under the light of the theory developed, we can interpret the label shift on the latent space and point out the drawback of learning domain invariant representations. Finally, grounded on the developed theory, we propose LAMDA which outperforms the baselines on real-world datasets. Last but not least, our theory can be extended to rigorously define label shift in various DA settings, but we leave it to future research.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 214–223. PMLR, 06–11 Aug 2017. URL http://proceedings.mlr.press/v70/arjovsky17a.html.

Ben-David, S. and Urner, R. On the hardness of domain adaptation and the utility of unlabeled target samples. In *Proceedings of the 23rd International Conference on Algorithmic Learning Theory*, pp. 139–153, 2012. ISBN 9783642341052.

Ben-David, S. and Urner, R. Domain adaptation—can quantity compensate for quality? *Annals of Mathematics and Artificial Intelligence*, 70(3):185–202, March 2014. ISSN 1012-2443.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Mach. Learn.*, 79(1-2):151–175, May 2010. ISSN 0885-6125.

Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., and Erhan, D. Domain separation networks. In *Advances in neural information processing systems*, pp. 343–351, 2016.

Cortes, C., Mohri, M., and Medina, A. M. Adaptation based on generalized discrepancy. *The Journal of Machine Learning Research*, 20(1):1–30, 2019.

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems*, pp. 3730–3739, 2017.

Damodaran, B. B., Kellenberger, B., Flamary, R., Tuia, D., and Courty, N. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In Ferrari, V., Hebert, M., Sminchisescu, C., and Weiss, Y. (eds.), *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*, volume 11208 of *Lecture Notes in Computer Science*, pp. 467–483, 2018.

Deng, Z., Luo, Y., and Zhu, J. Cluster alignment with a teacher for unsupervised domain adaptation, 2019.

Endres, D. M. and Schindelin, J. E. A new metric for probability distributions. *IEEE Trans. Inf. Theor.*, 49(7): 1858–1860, 2006. ISSN 0018-9448.

French, G., Mackiewicz, M., and Fisher, M. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018.

Ganin, Y. and Lempitsky, V. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pp. 1180–1189, 2015.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, jan 2016. ISSN 1532-4435.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. A pac-bayesian approach for domain adaptation with specialization to linear classifiers. In *Proceedings of the 30th International Conference on International Conference on Machine Learning*, ICML'13, 2013.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. A new pac-bayesian perspective on domain adaptation. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pp. 859–868, 2016.

Ghifary, M., Kleijn, W. B., Zhang, M., Balduzzi, D., and Li, W. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*, pp. 597–613. Springer, 2016.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Gretton, A., Borgwardt, K. M., Rasch, M., Schölkopf, B., and Smola, A. J. A kernel method for the two-sample-problem. In *Advances in neural information processing systems*, pp. 513–520, 2007.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., and Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning*, pp. 1989–1998, 2018.

Jiang, X., Lao, Q., Matwin, S., and Havaei, M. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4816–4827. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/jiang20d.html.

Johansson, F. D., Sontag, D., and Ranganath, R. Support and invertibility in domain-invariant representations. In *Proceedings of Machine Learning Research*, volume 89, pp. 527–536, 2019.

Kurmi, V. K., Kumar, S., and Namboodiri, V. P. Attending to discriminative certainty for domain adaptation. *CoRR*, abs/1906.03502, 2019.

Lee, C., Batra, T., Baig, M. H., and Ulbricht, D. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 10285–10295. Computer Vision Foundation / IEEE, 2019.

Li, M., Zhai, Y., Luo, Y., Ge, P., and Ren, C. Enhanced transport distance for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Liang, J., Hu, D., and Feng, J. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation, 2020.

Long, M., Cao, Y., Wang, J., and Jordan, M. Learning transferable features with deep adaptation networks. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 97–105, Lille, France, 2015.

Long, M., Zhu, H., Wang, J., and Jordan, M. I. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems 29*, pp. 136–144. 2016.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. Conditional adversarial domain adaptation. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 1640–1650. Curran Associates, Inc., 2018.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation with multiple sources. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L. (eds.), *Advances in Neural Information Processing Systems 21*, pp. 1041–1048. 2009.

Nguyen, T., Le, T., Zhao, H., Tran, H. Q., Nguyen, T., and Phung, D. Most: Multi-source domain adaptation via optimal transport for student-teacher learning. In *UAI*, 2021a.

Nguyen, T., Le, T., Zhao, H., Tran, H. Q., Nguyen, T., and Phung, D. Tidot: A teacher imitation learning approach for domain adaptation with optimal transport. In *IJCAI*, 2021b.

Nguyen, V., Le, T., Le, T., Nguyen, K., Vel, O. D., Montague, P., Qu, L., and Phung, D. Deep domain adaptation for vulnerable code function identification. In *IJCNN*, 2019.

Nguyen, V., Le, T., De Vel, O., Montague, P., Grundy, J., and Phung, D. Dual-component deep domain adaptation: A new approach for cross project software vulnerability detection. In *PAKDD*, 2020.

Nowozin, S., Cseke, B., and Tomioka, R. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pp. 271–279, 2016.

Pan, Y., Yao, T., Li, Y., Wang, Y., Ngo, C., and Mei, T. Transferrable prototypical networks for unsupervised domain adaptation. In *CVPR*, pp. 2234–2242, 2019.

Redko, I., Habrard, A., and Sebban, M. Theoretical analysis of domain adaptation with optimal transport. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 737–753, 2017.

Saito, K., Ushiku, Y., and Harada, T. Asymmetric tri-training for unsupervised domain adaptation. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 2988–2997. JMLR. org, 2017.

Saito, K., Watanabe, K., Ushiku, Y., and Harada, T. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Sankaranarayanan, S., Balaji, Y., Castillo, C. D., and Chellappa, R. Generate to adapt: Aligning domains using generative adversarial networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8503–8512, 2018.

Santambrogio, F. Optimal transport for applied mathematicians. *Birkäuser, NY*, pp. 99–102, 2015.

Shu, R., Bui, H., Narui, H., and Ermon, S. A DIRT-t approach to unsupervised domain adaptation. In *International Conference on Learning Representations*, 2018.

Sun, B. and Saenko, K. Deep coral: Correlation alignment for deep domain adaptation. In Hua, G. and Jégou, H. (eds.), *Computer Vision – ECCV 2016 Workshops*, pp. 443–450, Cham, 2016. Springer International Publishing. ISBN 978-3-319-49409-8.

Tolstikhin, I. O., Bousquet, O., Gelly, S., and Schölkopf, B. Wasserstein auto-encoders. *CoRR*, abs/1711.01558, 2018.

Tzeng, E., Hoffman, J., Darrell, T., and Saenko, K. Simultaneous deep transfer across domains and tasks. *CoRR*, 2015.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2962–2971, 2017.

van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

Vapnik, V. N. *The Nature of Statistical Learning Theory*. Springer, second edition, November 1999. ISBN 0387987800.

Villani, C. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2008. ISBN 9783540710509.

Wang, Q. and Breckon, T. P. Unsupervised domain adaptation via structured prediction based selective pseudo-labeling. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, pp. 6243–6250, 2020.

Wang, X., Jin, Y., Long, M., Wang, J., and Jordan, M. I. Transferable normalization: Towards improving transferability of deep neural networks. In *Advances in Neural Information Processing Systems*, volume 32, pp. 1953–1963, 2019.

Xie, S., Zheng, Z., Chen, L., and Chen, C. Learning semantic representations for unsupervised domain adaptation. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5423–5432. PMLR, 10–15 Jul 2018.

Xie, Y., Chen, M., Jiang, H., Zhao, T., and Zha, H. On scalable and efficient computation of large scale optimal transport. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6882–6892, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

Xu, R., Liu, P., Wang, L., Chen, C., and Wang, J. Reliable weighted optimal transport for unsupervised domain adaptation. In *CVPR 2020*, June 2020.

You, K., Wang, X., Long, M., and Jordan, M. Towards accurate model selection in deep unsupervised domain adaptation. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7124–7133. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/you19a.html.

Zhang, W., Ouyang, W., Li, W., and Xu, D. Collaborative and adversarial network for unsupervised domain adaptation. In *CVPR*, pp. 3801–3809, 2018.

Zhang, Y., Liu, Y., Long, M., and Jordan, M. I. Bridging theory and algorithm for domain adaptation. *CoRR*, abs/1904.05801, 2019a.

Zhang, Y., Tang, H., Jia, K., and Tan, M. Domain-symmetric networks for adversarial domain adaptation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5026–5035, 2019b.

Zhao, H., Des Combes, R. T., Zhang, K., and Gordon, G. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pp. 7523–7532, 2019.

Zhu, P., Wang, H., and Saligrama, V. Learning classifiers for target domain with limited or no labels. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7643–7653. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/zhu19d.html.

Supplementary Material for "LAMDA: Label Matching Deep Domain Adaptation"

In this supplementary material, we provide complete detail for all proofs presented in our main paper together with the related background so that it can be as self-contained as possible. In the following part, we present the experiment on a synthetic dataset to verify our theory, followed by the experimental settings and datasets for our LAMDA.

1 Related Background

In this section, we present the related background for our paper. We depart with the introduction of pushforward measure followed by the definition of optimal transport and the introduction of a standard machine learning setting.

1.1 Pushforward Measure

Given two probability spaces $(\mathcal{X}, \mathcal{F}, \mu)$ and $(\mathcal{Y}, \mathcal{G})$ where $\mathcal{X}, \mathcal{Y}$ are two sample spaces, $\mathcal{F}, \mathcal{G}$ are two σ-algebras over $\mathcal{X}, \mathcal{Y}$ respectively, and μ is a probability measure, a map $T : \mathcal{X} \to \mathcal{Y}$ is said to be $(\mathcal{Y}, \mathcal{G})$- $(\mathcal{X}, \mathcal{F})$ measurable if for every $A \in \mathcal{G}$, the inverse $T^{-1}(A) \in \mathcal{F}$. The $(\mathcal{Y}, \mathcal{G})$- $(\mathcal{X}, \mathcal{F})$ measurable map T when applied to $(\mathcal{X}, \mathcal{F}, \mu)$ induces a distribution ν over $(\mathcal{Y}, \mathcal{G})$ which is defined as:

$$\nu(A) = \mu\left(T^{-1}(A)\right), \forall A \in \mathcal{G}$$

We also say that the map T transport the probability measure μ to ν and denote as $\nu = T_{\#}\mu$. Furthermore, if μ and ν are two given atomless probability measures over $(\mathcal{X}, \mathcal{F})$ and $(\mathcal{Y}, \mathcal{G})$, there exists a bijection $T : \mathcal{X} \to \mathcal{Y}$ that transports μ to ν. This is known as measurable isomorphism and formally stated in [32] (Chapter 1, Page 19).

Theorem 1. *Given two probability spaces $(\mathcal{X}, \mathcal{F}, \mu)$ and $(\mathcal{Y}, \mathcal{G}, \nu)$ with two atomless probability μ, ν over two Polish spaces $\mathcal{X}$, $\mathcal{Y}$ (i.e., separably complete metric spaces), there exist a bijection $T : \mathcal{X} \to \mathcal{Y}$ that transports μ to ν, i.e., $T_{\#}\mu = \nu$.*

1.2 Optimal Transport

Given two probability measures $(\mathcal{X}, \mu)$ and $(\mathcal{Y}, \nu)$ and a cost function $c(\mathbf{x}, \mathbf{x}')$, under the conditions stated in Theorems 1.32 and 1.33 [26], two following definitions of Wasserstein (WS) distance are equivalent:

$$W_{c,p}(\mu, \nu) = \inf_{T_{\#}\mu = \nu} \mathbb{E}_{\mathbf{x}\sim\mu}\left[c(\mathbf{x}, T(\mathbf{x}))^p\right]^{1/p}$$

$$W_{c,p}(\mu, \nu) = \inf_{\pi \in \Gamma(\mu,\nu)} \mathbb{E}_{(\mathbf{x},\mathbf{x}')\sim\pi}\left[c(\mathbf{x}, \mathbf{x}')^p\right]^{1/p}$$

where $p > 0$ and $\Gamma(\mu, \nu)$ specifies the set of joint distributions over $\mathcal{X} \times \mathcal{Y}$ which admits μ and ν as marginals. The first definition is known as Monge problem (MP), while the second one is known as Kantorovich problem (KP).

We now restate the sufficient conditions for which (MP) and (KP) are equivalent (cf. Theorems 1.32 and 1.33 [26]).

Theorem 2. *If $\mathcal{X}$ and $\mathcal{Y}$ are compact, Polish metric spaces, μ and ν are atomless, and c is a lower semi-continuous function, (KP) is equivalent to (MP) in the sense that two infima are equal.*

In what follows, we assume that the relevant conditions are satisfied and use (KP) and (MP) interchangeably depending on the contexts. More specifically, we use (MP) in Theorem (9), while using (KP) in the rest.

1.3 Machine Learning Setting and General Loss

According to [31], a standard machine learning system consists of three components: the generator, the supervisor, and the hypothesis class.

Generator The generator is the mechanism to generate data examples $\mathbf{x} \in \mathbb{R}^d$ and is mathematically formulated by an existed but unknown distribution $p(\mathbf{x})$.

Supervisor The supervisor is the mechanism to assign labels y (e.g., $y \in \{1, 2, \ldots, C\}$ for the classification problem and $y \in \mathbb{R}$ for the regression problem) to a data example $\mathbf{x}$ and is mathematically formulated as a conditional distribution $p(y \mid \mathbf{x})$.

Hypothesis class This specifies the hypothesis set $\mathcal{H} = \{h_{\boldsymbol{\theta}} \mid \boldsymbol{\theta} \in \Theta\}$ parameterized by $\boldsymbol{\theta}$ which is used to predict label for the data examples $\mathbf{x}$.

Given a loss function $l(x, y; \boldsymbol{\theta}) = \ell(y, h_{\boldsymbol{\theta}}(\mathbf{x}))$ where $\ell : \Delta_C \to \mathbb{R}$ (Δ_C is the C-simplex) and $\ell(y, y')$ specifies the loss suffered if predicting the data example $\mathbf{x}$ with the label y' while its true label is y, the general loss of the hypothesis $h_{\boldsymbol{\theta}}$ is defined as the expected loss caused by $h_{\boldsymbol{\theta}}$:

$$R(\boldsymbol{\theta}) = \mathbb{E}_{p(\mathbf{x},y)}\left[\ell(y,\, h_{\boldsymbol{\theta}}(\mathbf{x}))\right] = \int \ell(y,\, h_{\boldsymbol{\theta}}(\mathbf{x}))\, p(\mathbf{x}, y)\, d\mathbf{x}dy$$

The optimal parameter $\boldsymbol{\theta}^* \in \Theta$ is sought by minimizing the general loss as:

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}}\, R(\boldsymbol{\theta})$$

2 Theoretical Results

2.1 Gap between target and source domains

In this section, we investigate the variance $\Delta R(h^s, h^t)$ between the expected loss in target domain $R^t(h^t)$ and the expected loss in source domain $R^s(h^s)$ where $h^t = h^s \circ T$. We embark on with the following simple yet key proposition indicating the connection between $R^t(h^t)$ and $R^{\#}(h^s)$.

Proposition 3. *As long as $h^t = h^s \circ T$, we have $R^t(h^t) = R^{\#}(h^s)$.*

Proof. The proof of the proposition is directly from the definitions of h^t, h^s, and expected losses. In particular, we find that

$$R^{\#}(h^s) = \int \ell(y, h^s(\mathbf{x}))\, p^{\#}(y \mid \mathbf{x})\, p^{\#}(\mathbf{x})\, d\mathbf{x}dy = \mathbb{E}_{\mathbb{P}^{\#}}\left[\int \ell(y, h^s(\mathbf{x}))\, p^{\#}(y \mid \mathbf{x})\, dy\right].$$

Recall that, T transports the target distribution $\mathbb{P}^t$ to the source distribution $\mathbb{P}^\#$, we achieve that

$$\begin{aligned} R^\# \left(h^s\right) &= \mathbb{E}_{\mathbb{P}^t}\left[\int \ell\left(y, h^s\left(T\left(\mathbf{x}\right)\right)\right) p^\#\left(y \mid T\left(\mathbf{x}\right)\right) dy\right] \\ &= \mathbb{E}_{\mathbb{P}^t}\left[\int \ell\left(y, h^t\left(\mathbf{x}\right)\right) p^t\left(y \mid \mathbf{x}\right) dy\right] = R^t\left(h^t\right), \end{aligned}$$

where the second equality is due to the connection $h^t = h^s \circ T$. As a consequence, we reach the conclusion of the proposition. □

Theorem 4. ***(Theorem 1 in the main paper)*** *Given Assumption (A.1), then for any hypothesis $h^s \in \mathcal{H}^s$, the following inequality holds:*

$$\Delta R\left(h^s, h^t\right) \leq M\left(\mathbf{W}_{c_{0/1}}\left(\mathbb{P}^s, \mathbb{P}^\#\right) + \mathbb{E}_{\mathbb{P}^t}\left[\left\|\Delta p\left(\cdot \mid \mathbf{x}\right)\right\|_1\right]\right),$$

where $\Delta p\left(\cdot \mid \mathbf{x}\right) := \left\|\left[p^t\left(y = i \mid \mathbf{x}\right) - p^s\left(y = i \mid T\left(\mathbf{x}\right)\right)\right]_{i=1}^C\right\|_1$, and $W_{c_{0/1}}\left(\cdot, \cdot\right)$ is the Wasserstein distance with respect to the cost function $c_{0/1}\left(\mathbf{x}, \mathbf{x}'\right) = \mathbf{1}_{\mathbf{x} \neq \mathbf{x}'}$, returning 1 if $\mathbf{x} \neq \mathbf{x}'$ and 0 otherwise.

Proof. Invoking the result from Proposition 3 and the basic triangle inequality, we obtain that

$$\begin{aligned} \Delta R\left(h^s, h^t\right) =& \left|R^t\left(h^t\right) - R^s\left(h^s\right)\right| = \left|R^\#\left(h^s\right) - R^s\left(h^s\right)\right| \\ =& \left|R^\#\left(h^s\right) - R^{\#,s}\left(h^s\right) + R^{\#,s}\left(h^s\right) - R^s\left(h^s\right)\right| \\ \leq& \left|R^\#\left(h^s\right) - R^{\#,s}\left(h^s\right)\right| + \left|R^{\#,s}\left(h^s\right) - R^s\left(h^s\right)\right|. \end{aligned}$$

To achieve the conclusion of the theorem, it is sufficient to upper bound the two terms $\left|R^\#\left(h^s\right) - R^{\#,s}\left(h^s\right)\right|$ and $\left|R^{\#,s}\left(h^s\right) - R^s\left(h^s\right)\right|$. For the first term, according the definition of expected losses, we find that

$$\begin{aligned} \left|R^\#\left(h^s\right) - R^{\#,s}\left(h^s\right)\right| &= \left|\int \ell\left(h^s\left(\mathbf{x}\right), y\right)\left(p^\#\left(y \mid \mathbf{x}\right) - p^s\left(y \mid \mathbf{x}\right)\right) p^\#\left(\mathbf{x}\right) d\mathbf{x} dy\right| \\ &= \left|\sum_{y=1}^C \int \ell\left(h^s\left(\mathbf{x}\right), y\right)\left(p^\#\left(y \mid \mathbf{x}\right) - p^s\left(y \mid \mathbf{x}\right)\right) p^\#\left(\mathbf{x}\right) d\mathbf{x}\right| \\ &\leq \sum_{y=1}^C \int \ell\left(h^s\left(\mathbf{x}\right), y\right)\left|p^\#\left(y \mid \mathbf{x}\right) - p^s\left(y \mid \mathbf{x}\right)\right| p^\#\left(\mathbf{x}\right) d\mathbf{x} \\ &\leq M \sum_{y=1}^C \int \left|p^\#\left(y \mid \mathbf{x}\right) - p^s\left(y \mid \mathbf{x}\right)\right| p^\#\left(\mathbf{x}\right) d\mathbf{x} \\ &= M \int \sum_{y=1}^C \left|p^\#\left(y \mid \mathbf{x}\right) - p^s\left(y \mid \mathbf{x}\right)\right| p^\#\left(\mathbf{x}\right) d\mathbf{x} \\ &\leq M \mathbb{E}_{\mathbb{P}^\#}\left[\left\|\left[p^\#\left(y \mid \mathbf{x}\right) - p^s\left(y \mid \mathbf{x}\right)\right]_{y=1}^C\right\|_1\right] \\ &\overset{(1)}{=} M \mathbb{E}_{\mathbb{P}^t}\left[\left\|\left[p^\#\left(y \mid T\left(\mathbf{x}\right)\right) - p^s\left(y \mid T\left(\mathbf{x}\right)\right)\right]_{y=1}^C\right\|_1\right] \\ &= M \mathbb{E}_{\mathbb{P}^t}\left[\left\|\left[p^t\left(y \mid \mathbf{x}\right) - p^s\left(y \mid T\left(\mathbf{x}\right)\right)\right]_{y=1}^C\right\|_1\right], \end{aligned} \tag{1}$$

where (1) is from the fact that $T_\# \mathbb{P}^t = \mathbb{P}^\#$.

For the second term, similar argument as the above argument leads to

$$\begin{aligned}
\left|R^{\#,s}\left(h^s\right)-R^s\left(h^s\right)\right| &= \left|\sum_{y=1}^{C}\int \ell\left(h^s\left(\mathbf{x}\right),y\right)p^s\left(y\mid\mathbf{x}\right)\left[p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right]d\mathbf{x}\right| \\
&\leq \sum_{y=1}^{C}\int \ell\left(h^s\left(\mathbf{x}\right),y\right)p^s\left(y\mid\mathbf{x}\right)\left|p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right|d\mathbf{x} \\
&\leq M\sum_{y=1}^{C}\int p^s\left(y\mid\mathbf{x}\right)\left|p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right|d\mathbf{x} \\
&= M\int\sum_{y=1}^{C}p^s\left(y\mid\mathbf{x}\right)\left|p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right|d\mathbf{x} \\
&= M\int\left|p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right|d\mathbf{x} = MW_{c_{0/1}}\left(\mathbb{P}^s,\mathbb{P}^{\#}\right), \qquad (2)
\end{aligned}$$

where the final equality is from the fact that cost matrix $c_{0/1}$ is given by $c_{0/1}\left(\mathbf{x},\mathbf{x}'\right)=\mathbf{1}_{\mathbf{x}\neq\mathbf{x}'}$, which returns 1 if $\mathbf{x}\neq\mathbf{x}'$ and 0 otherwise (for the second equality, please refer to [12], Page 7 and the coupling characterization of total variance distance).

Combining the results from (1) and (2), we arrive at the bound that

$$\begin{aligned}
\Delta R\left(h^s,h^t\right) &\leq M\left(W_{c_{0/1}}\left(\mathbb{P}^s,\mathbb{P}^{\#}\right)+\mathbb{E}_{\mathbb{P}^t}\left[\left\|p^t\left(y\mid\mathbf{x}\right)-p^s\left(y\mid T\left(\mathbf{x}\right)\right)\right\|_1\right]\right) \\
&= M\left(W_{c_{0/1}}\left(\mathbb{P}^s,\mathbb{P}^{\#}\right)+\mathbb{E}_{\mathbb{P}^t}\left[\left\|\Delta p\left(y\mid\mathbf{x}\right)\right\|_1\right]\right).
\end{aligned}$$

As a consequence, we reach the conclusion of the theorem. □

Remark 5. If the following assumptions hold:

(i) The transformation mapping $T\left(\mathbf{x}\right)=\mathbf{x}$, i.e., we use the same hypothesis set for both the source and target domains,

(ii) The loss $\ell\left(y,h\left(\mathbf{x}\right)\right)=\frac{1}{2}\left|y-h\left(\mathbf{x}\right)\right|$ where we restrict to consider hypothesis $h:\mathcal{X}\to\{-1,1\}$,

then we recover the theoretical result obtained in [2].

Remark 6. When $W_{c_{0/1}}\left(\mathbb{P}^s,\mathbb{P}^{\#}\right)=0$, i.e., $T_{\#}\mathbb{P}^t=\mathbb{P}^s$, and there is a harmony between two supervisors of source and target domain, i.e., $p^t\left(y\mid\mathbf{x}\right)=p^s\left(y\mid T\left(\mathbf{x}\right)\right)$), Theorem 4 suggests that we can perfectly do transfer learning without loss of performance. This fact is summarized in the following corollary.

Corollary 7. *Assume that $T_{\#}\mathbb{P}^t=\mathbb{P}^s$ and the source and target supervisor distributions are harmonic in the sense that $p^s\left(y\mid T\left(\mathbf{x}\right)\right)=p^t\left(y\mid\mathbf{x}\right)$ for $\mathbf{x}\sim\mathbb{P}_t$. Then, we can do a perfect transfer learning between the source and target domains.*

Proof. For any $h^s\in\mathcal{H}^s$, denote $h^t=h^s\circ T$, we have

$$\begin{aligned}
R^s\left(h^s\right) &= \mathbb{E}_{\mathbb{P}^s}\left[\int \ell\left(y,h^s\left(\mathbf{x}\right)\right)p^s\left(y\mid\mathbf{x}\right)dy\right] \\
&= \mathbb{E}_{\mathbb{P}^t}\left[\int \ell\left(y,h^s\left(T\left(\mathbf{x}\right)\right)\right)p^s\left(y\mid T\left(\mathbf{x}\right)\right)dy\right] \\
&= \mathbb{E}_{\mathbb{P}^t}\left[\int \ell\left(y,h^t\left(\mathbf{x}\right)\right)p^t\left(y\mid\mathbf{x}\right)dy\right] = R^t\left(h^t\right),
\end{aligned}$$

where the second equality is from the fact T transport $\mathbb{P}^t$ to $\mathbb{P}^s$. □

Furthermore, given a decreasing function $\phi : \mathbb{R} \to [0,1]$, a hypothesis h^s is said to be ϕ-Lipschitz transferable [6] w.r.t. a joint distribution $\gamma \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^{\#}\right)$, the metric c, and the norm $\|\cdot\|$ if for all $\lambda > 0$, we have

$$\mathbb{P}_{(\mathbf{x}_s, \mathbf{x}_{\#}) \sim \gamma}\left[\|h^s(\mathbf{x}_s) - h^s(\mathbf{x}_{\#})\| > \lambda c(\mathbf{x}_s, \mathbf{x}_{\#})\right] \leq \phi(\lambda).$$

Theorem 8. ***(Theorem 3 in the main paper)*** *Assume that Assumptions (A.1) and (A2) hold, the hypothesis h^s satisfies ϕ-Lipschitz transferable w.r.t the optimal joint distribution (transport plan) $\gamma^* \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^{\#}\right)$, c and $\|\cdot\|$, the following inequality holds for all $\lambda > 0$:*

$$\begin{aligned}\Delta R\left(h^s, h^t\right) \leq M\left(\mathbb{E}_{\mathbb{P}^t}\left[\|\Delta p(\cdot \mid \mathbf{x})\|_1\right] + 2\phi(\lambda)\right) \\ + kC\lambda \mathrm{W}_{c,p}\left(\mathbb{P}^s, \mathbb{P}^{\#}\right).\end{aligned}$$

Proof. We have

$$\begin{aligned}\Delta R\left(h^s, h^t\right) =& \left|R^t\left(h^t\right) - R^s\left(h^s\right)\right| = \left|R^{\#}\left(h^s\right) - R^s\left(h^s\right)\right| \\ =& \left|R^{\#}\left(h^s\right) - R^{\#,s}\left(h^s\right) + R^{\#,s}\left(h^s\right) - R^s\left(h^s\right)\right| \\ \leq& \left|R^{\#}\left(h^s\right) - R^{\#,s}\left(h^s\right)\right| + \left|R^{\#,s}\left(h^s\right) - R^s\left(h^s\right)\right|.\end{aligned}$$

We know that the first term can be bounded as

$$\left|R^{\#}\left(h^s\right) - R^{\#,s}\left(h^s\right)\right| \leq M\mathbb{E}_{\mathbb{P}^t}\left[\|\Delta p(\cdot \mid \mathbf{x})\|_1\right].$$

We manipulate the second term as

$$\begin{aligned}\left|R^{\#,s}\left(h^s\right) - R^s\left(h^s\right)\right| &= \left|\int \ell\left(h^s(\mathbf{x}), y\right) p^s(y \mid \mathbf{x}) \left[p^{\#}(\mathbf{x}) - p^s(\mathbf{x})\right] d\mathbf{x} dy\right| \\ &= \left|\sum_{y=1}^{C} \int \ell\left(h^s(\mathbf{x}), y\right) p^s(y \mid \mathbf{x}) \left(p^{\#}(\mathbf{x}) - p^s(\mathbf{x})\right) d\mathbf{x}\right| \\ &\leq \sum_{y=1}^{C} \left|\int \ell\left(h^s(\mathbf{x}), y\right) p^s(y \mid \mathbf{x}) p^{\#}(\mathbf{x}) d\mathbf{x} - \int \ell\left(h^s(\mathbf{x}), y\right) p^s(y \mid \mathbf{x}) p^s(\mathbf{x}) d\mathbf{x}\right| \\ &= \sum_{y=1}^{C} \left|\int \ell\left(h^s\left(\mathbf{x}^{\#}\right), y\right) p^s\left(y \mid \mathbf{x}^{\#}\right) d\mathbb{P}^{\#}\left(\mathbf{x}^{\#}\right) - \int \ell\left(h^s(\mathbf{x}^s), y\right) p^s(y \mid \mathbf{x}^s) d\mathbb{P}^s(\mathbf{x}^s)\right| \\ &= \sum_{y=1}^{C} \left|\int \left[\ell\left(h^s\left(\mathbf{x}^{\#}\right), y\right) p^s\left(y \mid \mathbf{x}^{\#}\right) - \ell\left(h^s(\mathbf{x}^s), y\right) p^s(y \mid \mathbf{x}^s)\right] d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right)\right| \\ &\leq \sum_{y=1}^{C} \left(\left|\int_A \left[\ell\left(h^s\left(\mathbf{x}^{\#}\right), y\right) p^s\left(y \mid \mathbf{x}^{\#}\right) - \ell\left(h^s(\mathbf{x}^s), y\right) p^s(y \mid \mathbf{x}^s)\right] d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right)\right| \right. \\ &\left. + \left|\int_{A^c} \left[\ell\left(h^s\left(\mathbf{x}^{\#}\right), y\right) p^s\left(y \mid \mathbf{x}^{\#}\right) - \ell\left(h^s(\mathbf{x}^s), y\right) p^s(y \mid \mathbf{x}^s)\right] d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right)\right|\right),\end{aligned}$$

where we denote $A = \left\{\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) : \left\|h\left(\mathbf{x}^{\#}\right) - h\left(\mathbf{x}^s\right)\right\| \leq \lambda c\left(\mathbf{x}^{\#}, \mathbf{x}^s\right)\right\}$, hence $\gamma^*(A^c) \leq \phi(\lambda)$.

We manipulate the second term as:

$$\begin{aligned}
&\sum_{y=1}^{C}\left|\int_{A^c}\left[\ell\left(h^s\left(\mathbf{x}^{\#}\right),y\right)p^s\left(y\mid\mathbf{x}^{\#}\right)-\ell\left(h^s\left(\mathbf{x}^s\right),y\right)p^s\left(y\mid\mathbf{x}^s\right)\right]d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right)\right| \\
&\leq\sum_{y=1}^{C}\int_{A^c}\left[\ell\left(h^s\left(\mathbf{x}^{\#}\right),y\right)p^s\left(y\mid\mathbf{x}^{\#}\right)+\ell\left(h^s\left(\mathbf{x}^s\right),y\right)p^s\left(y\mid\mathbf{x}^s\right)\right]d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right) \\
&\leq M\sum_{y=1}^{C}\int_{A^c}\left[p\left(y\mid\mathbf{x}^{\#}\right)+p^s\left(y\mid\mathbf{x}^s\right)\right]d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right) \\
&=M\int_{A^c}\left[\sum_{y=1}^{C}p\left(y\mid\mathbf{x}^{\#}\right)+\sum_{y=1}^{C}p^s\left(y\mid\mathbf{x}^s\right)\right]d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right) \\
&=2M\gamma^*\left(A^c\right)\leq 2M\phi\left(\lambda\right).
\end{aligned}$$

We derive the first term as:

$$\begin{aligned}
U=&\sum_{y=1}^{C}\left|\int_{A}\left[\ell\left(h^s\left(\mathbf{x}^{\#}\right),y\right)p^s\left(y\mid\mathbf{x}^{\#}\right)-\ell\left(h^s\left(\mathbf{x}^s\right),y\right)p^s\left(y\mid\mathbf{x}^s\right)\right]d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right)\right| \\
=&\sum_{y=1}^{C}\left|\int_{A}\ell\left(h^s\left(\mathbf{x}^{\#}\right),y\right)p^s\left(y\mid\mathbf{x}^{\#}\right)d\mathbb{P}^{\#}\left(\mathbf{x}^{\#}\right)-\int_{A}\ell\left(h^s\left(\mathbf{x}^s\right),y\right)p^s\left(y\mid\mathbf{x}^s\right)d\mathbb{P}^s\left(\mathbf{x}^s\right)\right| \\
=&\sum_{y=1}^{C}\left|\int_{A}\ell\left(h^s\left(\mathbf{x}\right),y\right)p^s\left(y\mid\mathbf{x}\right)p^{\#}\left(\mathbf{x}\right)d\mathbf{x}-\int_{A}\ell\left(h^s\left(\mathbf{x}\right),y\right)p^s\left(y\mid\mathbf{x}\right)p^s\left(\mathbf{x}\right)d\mathbf{x}\right| \\
\leq&\sum_{y=1}^{C}\int_{A}\ell\left(h^s\left(\mathbf{x}\right),y\right)p^s\left(y\mid\mathbf{x}\right)\left|p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right|d\mathbf{x} \\
\leq&\sum_{y=1}^{C}\int_{A}\ell\left(h^s\left(\mathbf{x}\right),y\right)\left|p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right|d\mathbf{x}
\end{aligned}$$

Denote $A_1=\left\{\mathbf{x}\in A:p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\geq 0\right\}$, we then have

$$\begin{aligned}
U\leq&\sum_{y=1}^{C}\int_{A}\ell\left(h^s\left(\mathbf{x}\right),y\right)\left|p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right|d\mathbf{x} \\
=&\sum_{y=1}^{C}\left[\int_{A_1}\ell\left(h^s\left(\mathbf{x}\right),y\right)\left(p^{\#}\left(\mathbf{x}\right)-p^s\left(\mathbf{x}\right)\right)d\mathbf{x}+\int_{A\backslash A_1}\ell\left(h^s\left(\mathbf{x}\right),y\right)\left(p^s\left(\mathbf{x}\right)-p^{\#}\left(\mathbf{x}\right)\right)d\mathbf{x}\right] \\
=&\sum_{y=1}^{C}\left[\int_{A_1}\left[\ell\left(h^s\left(\mathbf{x}^{\#}\right),y\right)-\ell\left(h^s\left(\mathbf{x}^s\right),y\right)\right]d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right)+\int_{A\backslash A_1}\left[\ell\left(h^s\left(\mathbf{x}^s\right),y\right)-\ell\left(h^s\left(\mathbf{x}^{\#}\right),y\right)\right]d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right)\right] \\
\leq&\sum_{y=1}^{C}\int_{A}\left|\ell\left(h^s\left(\mathbf{x}^{\#}\right),y\right)-\ell\left(h^s\left(\mathbf{x}^s\right),y\right)\right|d\gamma^*\left(\mathbf{x}^{\#},\mathbf{x}^s\right).
\end{aligned}$$

$$
\begin{aligned}
U &\leq \sum_{y=1}^{C} \int_{A} \left|\ell\left(h^s\left(\mathbf{x}^{\#}\right), y\right) - \ell\left(h^s\left(\mathbf{x}^s\right), y\right)\right| d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) \\
&\overset{(1)}{\leq} k \sum_{y=1}^{C} \int_{A} \left\|h^s\left(\mathbf{x}^{\#}\right) - h^s\left(\mathbf{x}^s\right)\right\|_1 d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) \\
&= kC \int_{A} \left\|h^s\left(\mathbf{x}^{\#}\right) - h^s\left(\mathbf{x}^s\right)\right\|_1 d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) \\
&\overset{(2)}{\leq} \lambda kC \int_{A} c\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) \leq \lambda kC \int c\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) d\gamma^*\left(\mathbf{x}^{\#}, \mathbf{x}^s\right) = \lambda kC W_c\left(\mathbb{P}^{\#}, \mathbb{P}^s\right) \\
&\overset{(3)}{\leq} \lambda kC W_{c,p}\left(\mathbb{P}^{\#}, \mathbb{P}^s\right).
\end{aligned}
$$

Here we note that we have (1) due to ℓ is k-Lipschitz w.r.t $\|\cdot\|$, (2) due to the definition of A, and (3) due to $p \geq 1$ hence $W_c\left(\mathbb{P}^{\#}, \mathbb{P}^s\right) \leq W_{c,p}\left(\mathbb{P}^{\#}, \mathbb{P}^s\right)$ (see Section 5.1 in [26]). □

2.2 Data shift via Wasserstein metric

Let $\mathcal{Z}$ be an intermediate space (i.e., the joint space $\mathcal{Z} = \mathbb{R}^m$). We consider the composite mappings: $T\left(\mathbf{x}\right) = T^2\left(T^1\left(\mathbf{x}\right)\right)$ where T^1 is a mapping from the target domain $\mathcal{X}^t$ to the joint space $\mathcal{Z}$ and T^2 maps from the joint space $\mathcal{Z}$ to the source domain $\mathcal{X}^s$ (note that if $\mathcal{Z} = \mathcal{X}^s$ then $T^2 = id$ is the identity function). Based on this structure, we consider the following optimization problem:

$$
\min_{T^1, T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#} \mathbb{P}^t, \mathbb{P}^s\right). \tag{3}
$$

In the following theorem, we demonstrate that the above optimization problem can be equivalently transformed into another form involving the joint space (see Figure 1 for an illustration of that theorem).

Theorem 9. ***(Theorem 4 in the main paper)*** *The optimal objective value of the OP (3) is equal to that of the OP (4), that is*

$$
\begin{aligned}
&\min_{T^1, T^2} \mathrm{W}_{c,p}\left(\left(T^2 \circ T^1\right)_{\#} \mathbb{P}^t, \mathbb{P}^s\right) = \\
&\min_{T^1, T^2} \min_{G^1 : T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}
\end{aligned} \tag{4}
$$

where G^1 is a map from $\mathcal{X}^s$ to $\mathcal{Z}$.

Proof. From the definition of Wasserstein metric, we obtain that

$$
W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#} \mathbb{P}^t, \mathbb{P}^s\right) = \inf_{L : L_{\#}\mathbb{P}^s = T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p}.
$$

Therefore, we can rewrite the optimization problem in the left side of (4) as follows:

$$
\min_{T^1, T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#} \mathbb{P}^t, \mathbb{P}^s\right) = \min_{T^1, T^2} \inf_{L : L_{\#}\mathbb{P}^s = T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p}.
$$

We first prove that

$$
\min_{T^1, T^2} \min_{G^1 : T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} \geq \min_{T^1, T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#} \mathbb{P}^t, \mathbb{P}^s\right).
$$

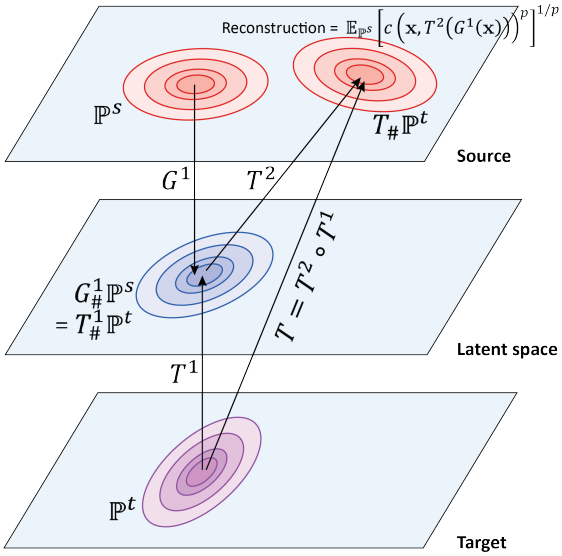

Figure 1: The mapping $T = T^2 \circ T^1$ maps from the target to source domains. We minimize $D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right)$ to close the discrepancy gap of the source and target domains in the joint space.

Given the mappings T^1, T^2, for any mapping G^1 satisfying the equation $T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s$, we let $T' = T^2 \circ G^1$. Then, we arrive at $T'_{\#}\mathbb{P}^s = T_{\#}\mathbb{P}^t$. Hence, we find that

$$\mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} = \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T'\left(\mathbf{x}\right)\right)^p\right]^{1/p} \geq \inf_{L:L_{\#}\mathbb{P}^s = T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p}.$$

The above inequality directly leads to

$$\min_{G^1:T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} \geq \inf_{L:L_{\#}\mathbb{P}^s = T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p}.$$

As a consequence, we achieve the following inequality

$$\begin{aligned}\min_{T^1,T^2} \min_{G^1:T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} &\geq \min_{T^1,T^2} \inf_{L:L_{\#}\mathbb{P}^s = T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p} \\ &= \min_{T^1,T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#}\mathbb{P}^t, \mathbb{P}^s\right).\end{aligned}$$

We now prove that

$$\min_{T^1,T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#}\mathbb{P}^t, \mathbb{P}^s\right) \geq \min_{T^1,T^2} \min_{G^1:T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.$$

Given the mapping T^1, we consider the distribution $\mathbb{Q}$ over the source domain such that there exists a map T^2 for which $T^2_{\#}\left(T^1_{\#}\mathbb{P}^t\right) = \mathbb{Q}$. For any mapping L satisfying the equation $L_{\#}\mathbb{P}^s = \mathbb{Q}$, we can find mappings U, V such that $U_{\#}\mathbb{P}^s = T^1_{\#}\mathbb{P}^t$ and $L = V \circ U$. To

this end, there exists a bijective mapping V satisfying $V_{\#}\left(T^1_{\#}\mathbb{P}^t\right) = \mathbb{Q}$ since these two distributions are atomless (see Theorem 1). Additionally, we can set $U = V^{-1} \circ L$. It is obvious that $U_{\#}\mathbb{P}^s = T^1_{\#}\mathbb{P}^t$ and $L = V \circ U$ from the definitions of U and V. Therefore, we have that

$$\begin{aligned}\mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p} =& \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, V\left(U\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} \\ \geq& \min_{T^2:T^2_{\#}\left(T^1_{\#}\mathbb{P}^t\right)=\mathbb{Q}} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.\end{aligned}$$

Invoking the above equality, we find that

$$\inf_{L:L_{\#}\mathbb{P}^s=\mathbb{Q}} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p} \geq \min_{T^2:T^2_{\#}\left(T^1_{\#}\mathbb{P}^t\right)=\mathbb{Q}} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.$$

With that inequality, we directly achieve the following inequality

$$\begin{aligned}&\min_{\mathbb{Q}} \inf_{L:L_{\#}\mathbb{P}^s=\mathbb{Q}} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p} \\ &\geq \min_{\mathbb{Q}} \min_{T^2:T^2_{\#}\left(T^1_{\#}\mathbb{P}^t\right)=\mathbb{Q}} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.\end{aligned}$$

$$\min_{T^2} \inf_{L:L_{\#}\mathbb{P}^s=T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p} \geq \min_{T^2} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.$$

Note that from the definitions of $\mathbb{Q}$ and T^2, it is obvious that

$$\min_{\mathbb{Q}} \inf_{L:L_{\#}\mathbb{P}^s=\mathbb{Q}} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p} = \min_{T^2} \inf_{L:L_{\#}\mathbb{P}^s=T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p}.$$

$$\begin{aligned}&\min_{T^2} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p} \\ &= \min_{\mathbb{Q}} \min_{T^2:T^2_{\#}\left(T^1_{\#}\mathbb{P}^t\right)=\mathbb{Q}} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.\end{aligned}$$

By varying the mapping T^1 in both sides of the above inequality, we arrive at the following inequality

$$\min_{T^1,T^2} \inf_{L:L_{\#}\mathbb{P}^s=T_{\#}\mathbb{P}^t} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, L\left(\mathbf{x}\right)\right)^p\right]^{1/p} \geq \min_{T^1,T^2} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.$$

Hence, we obtain that

$$\min_{T^1,T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#}\mathbb{P}^t, \mathbb{P}^s\right) \geq \min_{T^1,T^2} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.$$

Finally, we reach the conclusion as:

$$\min_{T^1,T^2} W_{c,p}\left(\left(T^2 \circ T^1\right)_{\#}\mathbb{P}^t, \mathbb{P}^s\right) = \min_{T^1,T^2} \min_{G^1:T^1_{\#}\mathbb{P}^t=G^1_{\#}\mathbb{P}^s} \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p\right]^{1/p}.$$

□

It is interesting to interpret G^1 and T^1 as two generators that map the source and target domains to the common joint space $\mathcal{Z}$ respectively. The constraint $T^1_{\#}\mathbb{P}^t = G^1_{\#}\mathbb{P}^s$ further indicates that the gap between the source and target distributions is closed in the joint space via two generators G^1 and T^1. Furthermore, T^2 maps from the joint space to the source domain and aims to reconstruct G^1. Similar to [29], we do relaxation and arrive at the optimization problem:

$$\min_{T^1,T^2,G^1} \left(\mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s} \left[c\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)^p \right]^{1/p} + \alpha D\left(G^1_{\#}\mathbb{P}^s, T^1_{\#}\mathbb{P}^t\right) \right), \tag{5}$$

where $D\left(\cdot,\cdot\right)$ specifies a divergence between two distributions over the joint space and $\alpha > 0$.

2.3 Label shift via Wasserstein metric

Since G^1 and T^1 are two maps from the source and target domains to the joint space, we can further define two source and target supervisor distributions on the joint space as $p^{\#,s}\left(y \mid G^1\left(\mathbf{x}\right)\right) = p^s\left(y \mid \mathbf{x}\right)$ and $p^{\#,t}\left(y \mid T^1\left(\mathbf{x}\right)\right) = p^t\left(y \mid \mathbf{x}\right)$. With respect to the joint space, the second term of the upper bound in Theorem 4 can be rewritten as in the following corollary.

Corollary 10. ***(Corollary 5 in the main paper)*** *The second term of the upper bound in Theorem 4 can be rewritten as*

$$\mathbb{E}_{\mathbb{P}^t}\left[\left\|p^{\#,s}\left(y \mid G^1\left(T^2\left(T^1\left(\mathbf{x}\right)\right)\right)\right) - p^{\#,t}\left(y \mid T^1\left(\mathbf{x}\right)\right)\right\|_1\right]. \tag{6}$$

Proof. The proof is trivial from the definitions of $p^{\#,s}\left(y \mid G^1\left(\mathbf{x}\right)\right) = p^s\left(y \mid \mathbf{x}\right)$ and $p^{\#,t}\left(y \mid T^1\left(\mathbf{x}\right)\right) = p^t\left(y \mid \mathbf{x}\right)$. □

Corollary 11. ***(Corollary 6 in the main paper)*** *Under the ideal scenario, the label mismatch term in (6) has a lower-bound*

$$\left\|\left[p^s\left(y=i\right) - p^t\left(y=i\right)\right]_{i=1}^C\right\|_1.$$

Proof. Under the ideal scenario, the label mismatch term becomes

$$\mathbb{E}_{\mathbb{P}^t}\left[\left\|p^{\#,s}\left(y \mid \left(T^1\left(\mathbf{x}\right)\right)\right) - p^{\#,t}\left(y \mid T^1\left(\mathbf{x}\right)\right)\right\|_1\right].$$

We derive as follows:

$$\begin{aligned}
U =& \mathbb{E}_{\mathbb{P}^t}\left[\left\|p^{\#,s}\left(y \mid \left(T^1\left(\mathbf{x}\right)\right)\right) - p^{\#,t}\left(y \mid T^1\left(\mathbf{x}\right)\right)\right\|_1\right] \\
=& \sum_{i=1}^C \int \left|p^{\#,s}\left(y=i \mid \left(T^1\left(\mathbf{x}\right)\right)\right) - p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right)\right| p^t\left(\mathbf{x}\right) d\mathbf{x} \\
\geq& \sum_{i=1}^C \left| \int \left(p^{\#,s}\left(y=i \mid \left(T^1\left(\mathbf{x}\right)\right)\right) - p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right)\right) p^t\left(\mathbf{x}\right) d\mathbf{x} \right| \\
=& \sum_{i=1}^C \left| \int p^{\#,s}\left(y=i \mid \left(T^1\left(\mathbf{x}\right)\right)\right) p^t\left(\mathbf{x}\right) d\mathbf{x} - \int p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right) p^t\left(\mathbf{x}\right) d\mathbf{x} \right| \\
=& \sum_{i=1}^C \left| \int p^{\#,s}\left(y=i \mid \left(T^1\left(\mathbf{x}\right)\right)\right) d\mathbb{P}^t - \int p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right) d\mathbb{P}^t \right| \\
=& \sum_{i=1}^C \left| \int p^{\#,s}\left(y=i \mid \mathbf{z}\right) dT^1_{\#}\mathbb{P}^t - \int p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right) d\mathbb{P}^t \right| \\
=& \sum_{i=1}^C \left| \int p^{\#,s}\left(y=i \mid \mathbf{z}\right) dG^1_{\#}\mathbb{P}^s - \int p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right) d\mathbb{P}^t \right|
\end{aligned}$$

$$
\begin{aligned}
U &\geq \sum_{i=1}^{C} \left| \int p^{\#,s}\left(y=i \mid \mathbf{z}\right) dG^1_{\#}\mathbb{P}^s - \int p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right) d\mathbb{P}^t \right| \\
&= \sum_{i=1}^{C} \left| \int p^{\#,s}\left(y=i \mid G^1\left(\mathbf{x}\right)\right) d\mathbb{P}^s - \int p^{\#,t}\left(y=i \mid T^1\left(\mathbf{x}\right)\right) d\mathbb{P}^t \right| \\
&= \sum_{i=1}^{C} \left| \int p^{s}\left(y=i \mid \mathbf{x}\right) d\mathbb{P}^s - \int p^{t}\left(y=i \mid \mathbf{x}\right) d\mathbb{P}^t \right| \\
&= \sum_{i=1}^{C} \left| \int p^{s}\left(y=i \mid \mathbf{x}\right) p^s\left(\mathbf{x}\right) d\mathbf{x} - \int p^{t}\left(y=i \mid \mathbf{x}\right) p^t\left(\mathbf{x}\right) d\mathbf{x} \right| \\
&= \sum_{i=1}^{C} \left| p^s\left(y=i\right) - p^t\left(y=i\right) \right| \\
&= \left\| \left[p^s\left(y=i\right) - p^t\left(y=i\right) \right]_{i=1}^{C} \right\|_1 .
\end{aligned}
$$

□

It is also worth mentioning that with regard to the latent space and the above equipment for $T = T^2 \circ T^1$, we have the following formulations for the source classifier (i.e., h^s) and target classifier (i.e., h^t) now become:

$$
h^s\left(\mathbf{x}\right) = \mathcal{A}\left(G^1\left(\mathbf{x}\right)\right) \text{ and } h^t\left(\mathbf{x}\right) = \mathcal{A}\left(G^1\left(T\left(\mathbf{x}\right)\right)\right). \tag{7}
$$

We define a new metric $\tilde{c}$ w.r.t. the family $\mathcal{H}^a$ of the classifier $\mathcal{A}$ as:

$$
\tilde{c}\left(\mathbf{z}_1, \mathbf{z}_2\right) = \sup_{\mathcal{A} \in \mathcal{H}^a} \left\| \mathcal{A}\left(\mathbf{z}_1\right) - \mathcal{A}\left(\mathbf{z}_2\right) \right\|_1 ,
$$

where $\mathbf{z}_1$ and $\mathbf{z}_2$ lie on the latent space. The following lemma states under which conditions, $\tilde{c}$ is a proper metric on the latent space.

Lemma 12. ***(Lemma 7 in the main paper)*** *For any $\mathbf{z}_1$ and $\mathbf{z}_2$, if $\mathcal{A}\left(\mathbf{z}_1\right) = \mathcal{A}\left(\mathbf{z}_2\right), \forall \mathcal{A} \in \mathcal{H}^a$ leads to $\mathbf{z}_1 = \mathbf{z}_2$, $\tilde{c}$ is a proper metric.*

Proof. First, $\tilde{c}\left(\mathbf{z}_1, \mathbf{z}_2\right) \geq 0$ and $\tilde{c}\left(\mathbf{z}_1, \mathbf{z}_2\right) = 0$ means $\mathcal{A}\left(\mathbf{z}_1\right) = \mathcal{A}\left(\mathbf{z}_2\right), \forall \mathcal{A} \in \mathcal{H}^a$, which leads to $\mathbf{z}_1 = \mathbf{z}_2$. Second, it is obvious that $\tilde{c}\left(\mathbf{z}_1, \mathbf{z}_2\right) = \tilde{c}\left(\mathbf{z}_2, \mathbf{z}_1\right), \forall \mathbf{z}_1, \mathbf{z}_2$.

Given any $\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3$, we have

$$
\begin{aligned}
\tilde{c}\left(\mathbf{z}_1, \mathbf{z}_3\right) &= \sup_{\mathcal{A} \in \mathcal{H}^a} \left\| \mathcal{A}\left(\mathbf{z}_1\right) - \mathcal{A}\left(\mathbf{z}_3\right) \right\|_1 \leq \sup_{\mathcal{A} \in \mathcal{H}^a} \left(\left\| \mathcal{A}\left(\mathbf{z}_1\right) - \mathcal{A}\left(\mathbf{z}_2\right) \right\|_1 + \left\| \mathcal{A}\left(\mathbf{z}_2\right) - \mathcal{A}\left(\mathbf{z}_3\right) \right\|_1 \right) \\
&\leq \sup_{\mathcal{A} \in \mathcal{H}^a} \left(\left\| \mathcal{A}\left(\mathbf{z}_1\right) - \mathcal{A}\left(\mathbf{z}_2\right) \right\|_1 \right) + \sup_{\mathcal{A} \in \mathcal{H}^a} \left(\left\| \mathcal{A}\left(\mathbf{z}_2\right) - \mathcal{A}\left(\mathbf{z}_3\right) \right\|_1 \right) \\
&= \tilde{c}\left(\mathbf{z}_1, \mathbf{z}_2\right) + \tilde{c}\left(\mathbf{z}_2, \mathbf{z}_3\right).
\end{aligned}
$$

Therefore, $\tilde{c}$ is a proper metric. □

It turns out that the necessary (also sufficient) condition in Lemma 12 is realistic and not hard to be satisfied (e.g., the family $\mathcal{H}^a$ contains any bijection). We now can define a WS distance $W_{\tilde{c},p}$ that involves in Theorem 14 whose proof needs the following lemma.

Lemma 13. *Let p^{h^s} be the density of the distribution $\mathbb{P}^{h^s}$ formed by pushing forward $\mathbb{P}^s$ via h^s and p^{h^t} be the density of the distribution $\mathbb{P}^{h^t}$ formed by pushing forward $\mathbb{P}^t$ via h^t. If $\gamma \in \Gamma\left(\mathbb{P}^{h^s}, \mathbb{P}^{h^t}\right)$, there exists $\gamma' \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^t\right)$ such that $\left(h^s, h^t\right)_{\#} \gamma' = \gamma$.*

Proof. Let denote γ^s as the joint distribution of the samples $\left(\mathbf{x}^s, h^s\left(\mathbf{x}^s\right)\right)$ where $\mathbf{x}^s \sim \mathbb{P}^s$ and γ^t as the joint distribution of the samples $\left(\mathbf{x}^t, h^t\left(\mathbf{x}^t\right)\right)$ where $\mathbf{x}^t \sim \mathbb{P}^t$. It is obvious that γ^s is a joint distribution of $\mathbb{P}^s$ and $\mathbb{P}^{h^s}$ and γ^t is a joint distribution of $\mathbb{P}^t$ and $\mathbb{P}^{h^t}$. According to the gluing lemma (see Lemma 5.5 in [26]), there exists a joint distribution μ such that for any draw $\left(\mathbf{x}^s, \boldsymbol{\tau}^s, \boldsymbol{\tau}^t, \mathbf{x}^t\right) \sim \mu$ then $\left(\mathbf{x}^s, \boldsymbol{\tau}^s\right) \sim \gamma^s$, $\left(\boldsymbol{\tau}^s, \boldsymbol{\tau}^t\right) \sim \gamma$, and $\left(\mathbf{x}^t, \boldsymbol{\tau}^t\right) \sim \gamma^t$.

Let γ' be the distribution of samples $\left(\mathbf{x}^s, \mathbf{x}^t\right)$ (i.e., the projection of μ onto the first and fourth dimensions). This follows that γ' is a joint distribution of $\mathbb{P}^s$ and $\mathbb{P}^t$ (i.e., $\gamma' \in \Gamma\left(\mathbb{P}^s, \mathbb{P}^t\right)$). In addition, since $\left(\mathbf{x}^s, \boldsymbol{\tau}^s\right) \sim \gamma^s$, $\boldsymbol{\tau}^s = h^s\left(\mathbf{x}^s\right)$, since $\left(\mathbf{x}^t, \boldsymbol{\tau}^t\right) \sim \gamma^t$, $\boldsymbol{\tau}^t = h^t\left(\mathbf{x}^t\right)$, and $\left(\boldsymbol{\tau}^s, \boldsymbol{\tau}^t\right) \sim \gamma$. Therefore, we reach $\left(h^s, h^t\right)_{\#} \gamma' = \gamma$.

We note that in the above proof, we employ a general form of the gluing lemma for 4 distributions and spaces. The proof is mainly based on the gluing lemma for 3 distributions and spaces and trivial. □

Theorem 14. ***(Theorem 8 in the main paper)*** *If $\tilde{c}$ is a proper metric and $p \geq 1$, the quantity $\left\|\left[p^s\left(y=i\right) - p^t\left(y=i\right)\right]_{i=1}^C\right\|_1$ has the upper-bounds:*

i) $R_1^s\left(h^s\right) + R_1^t\left(h^t\right) + W_{\tilde{c},p}\left(G_{\#}^1 \mathbb{P}^s, T_{\#}^1 \mathbb{P}^t\right)$ if $h^s := \mathcal{A}\left(G^1\left(\mathbf{x}\right)\right)$ and $h^t := \mathcal{A}\left(T^1\left(\mathbf{x}\right)\right)$.

ii) $R_1^s\left(h^s\right) + R_1^t\left(h^t\right) + W_{\tilde{c},p}\left(G_{\#}^1 \mathbb{P}^s, T_{\#}^1 \mathbb{P}^t\right) + W_{\tilde{c},p}\left(L_{\#} \mathbb{P}^t, T_{\#}^1 \mathbb{P}^t\right)$ where $L := T \circ G^1$, and h^s and h^t are defined in (7).

Here $R_1^s\left(h^s\right) := \int \left\|p^s\left(\cdot \mid \mathbf{x}\right) - h^s\left(\mathbf{x}\right)\right\|_1 p^s\left(\mathbf{x}\right) d\mathbf{x}$ and $R_1^t\left(h^t\right) := \int \left\|p^t\left(\cdot \mid \mathbf{x}\right) - h^t\left(\mathbf{x}\right)\right\|_1 p^t\left(\mathbf{x}\right) d\mathbf{x}$ are the general losses of h^s and h^t w.r.t. $\|\cdot\|_1$.

Proof. i) We derive as follows:

$$\begin{aligned}\left\|\left[p^s\left(y=i\right) - p^t\left(y=i\right)\right]_{i=1}^C\right\|_1 \leq & \left\|\left[p^s\left(y=i\right) - p^{h^s}\left(y=i\right)\right]_{i=1}^C\right\|_1 + \left\|\left[p^{h^s}\left(y=i\right) - p^{h^t}\left(y=i\right)\right]_{i=1}^C\right\|_1 \\ & + \left\|\left[p^{h^t}\left(y=i\right) - p^t\left(y=i\right)\right]_{i=1}^C\right\|_1,\end{aligned}$$

where p^{h^s} is the density of the distribution $\mathbb{P}^{h^s}$ formed by pushing forward $\mathbb{P}^s$ via h^s and p^{h^t} is the density of the distribution $\mathbb{P}^{h^t}$ formed by pushing forward $\mathbb{P}^t$ via h^t.

We manipulate the first term as:

$$\begin{aligned}&\left\|\left[p^s\left(y=i\right) - p^{h^s}\left(y=i\right)\right]_{i=1}^C\right\|_1 = \sum_{i=1}^C \left|p^s\left(y=i\right) - p^{h^s}\left(y=i\right)\right| \\ =&\sum_{i=1}^C \left|\int \left(p^s\left(y=i, \mathbf{x}\right) - p^{h^s}\left(y=i, \mathbf{x}\right)\right) p^s\left(\mathbf{x}\right) d\mathbf{x}\right| = \sum_{i=1}^C \left|\int \left(p^s\left(y=i, \mathbf{x}\right) - h_i^s\left(\mathbf{x}\right)\right) p^s\left(\mathbf{x}\right) d\mathbf{x}\right| \\ \leq &\int \sum_{i=1}^C \left|p^s\left(y=i, \mathbf{x}\right) - h_i^s\left(\mathbf{x}\right)\right| p^s\left(\mathbf{x}\right) d\mathbf{x} = \int \sum_{i=1}^C \left\|\left[p^s\left(y=i, \mathbf{x}\right) - h_i^s\left(\mathbf{x}\right)\right]_{i=1}^C\right\|_1 p^s\left(\mathbf{x}\right) d\mathbf{x} = R_1^s\left(h^s\right).\end{aligned}$$

Similarly, we can bound the third term as:

$$\left\|\left[p^{h^t}\left(y=i\right) - p^t\left(y=i\right)\right]_{i=1}^C\right\|_1 \leq R_1^t\left(h^t\right).$$

To handle the second term $\left\|\left[p^{h^s}(y=i)-p^{h^t}(y=i)\right]_{i=1}^{C}\right\|_1$, we first prove that

$$\left\|\left[p^{h^s}(y=i)-p^{h^t}(y=i)\right]_{i=1}^{C}\right\|_1 \leq W_1\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right),$$

where the WS w.r.t. the metric $\|\cdot\|_1$. Indeed, consider a joint distribution $\gamma \in \Gamma\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right)$. According to Lemma 13, there exists $\gamma' \in \Gamma\left(\mathbb{P}^s,\mathbb{P}^t\right)$ such that $\left(h^s,h^t\right)_{\#}\gamma' = \gamma$, we then have

$$\begin{aligned}
&\mathbb{E}_{(\mathbf{y}^s,\mathbf{y}^t)\sim\gamma}\left[\left\|\mathbf{y}^s-\mathbf{y}^t\right\|_1\right] = \mathbb{E}_{(\mathbf{x}^s,\mathbf{x}^t)\sim\gamma'}\left[\left\|h^s\left(\mathbf{x}^s\right)-h^t\left(\mathbf{x}^t\right)\right\|_1\right] \\
&= \int \left\|h^s\left(\mathbf{x}^s\right)-h^t\left(\mathbf{x}^t\right)\right\|_1 d\gamma'\left(\mathbf{x}^s,\mathbf{x}^t\right) = \sum_{i=1}^{C}\int \left|h_i^s\left(\mathbf{x}^s\right)-h_i^t\left(\mathbf{x}^t\right)\right| d\gamma'\left(\mathbf{x}^s,\mathbf{x}^t\right) \\
&\geq \sum_{i=1}^{C}\left|\int\left(h_i^s\left(\mathbf{x}^s\right)-h_i^t\left(\mathbf{x}^t\right)\right)d\gamma'\left(\mathbf{x}^s,\mathbf{x}^t\right)\right| = \sum_{i=1}^{C}\left|\int h_i^s\left(\mathbf{x}^s\right)d\mathbb{P}^s\left(\mathbf{x}^s\right)-\int h_i^t\left(\mathbf{x}^t\right)d\mathbb{P}^t\left(\mathbf{x}^t\right)\right| \\
&= \sum_{i=1}^{C}\left|p^{h^s}(y=i)-p^{h^t}(y=i)\right| = \left\|\left[p^{h^s}(y=i)-p^{h^t}(y=i)\right]_{i=1}^{C}\right\|_1.
\end{aligned}$$

Therefore, we achieve

$$W_1\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right) = \inf_{\gamma\in\Gamma\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right)}\mathbb{E}_{(\mathbf{y}^s,\mathbf{y}^t)\sim\gamma}\left[\left\|\mathbf{y}^s-\mathbf{y}^t\right\|_1\right] \geq \left\|\left[p^{h^s}(y=i)-p^{h^t}(y=i)\right]_{i=1}^{C}\right\|_1.$$

We now need to prove that $W_1\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right) \leq W_{\tilde{c},p}\left(G^1_{\#}\mathbb{P}^s,T^1_{\#}\mathbb{P}^t\right)$ $(p\geq 1)$. Indeed, given any $\gamma' \in \Gamma\left(G^1_{\#}\mathbb{P}^s,T^1_{\#}\mathbb{P}^t\right)$, let denote $\gamma = \mathcal{A}_{\#}\gamma'$, then $\gamma \in \Gamma\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right)$. We then have:

$$\begin{aligned}
&\mathbb{E}_{(\mathbf{y}^s,\mathbf{y}^t)\sim\gamma}\left[\left\|\mathbf{y}^s-\mathbf{y}^t\right\|_1\right] = \mathbb{E}_{(\boldsymbol{\tau}^s,\boldsymbol{\tau}^t)\sim\gamma'}\left[\left\|\mathcal{A}\left(\boldsymbol{\tau}^s\right)-\mathcal{A}\left(\boldsymbol{\tau}^t\right)\right\|_1\right] \\
&\leq \mathbb{E}_{(\boldsymbol{\tau}^s,\boldsymbol{\tau}^t)\sim\gamma'}\left[\tilde{c}\left(\boldsymbol{\tau}^s,\boldsymbol{\tau}^t\right)\right].
\end{aligned}$$

This follows that

$$W_1\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right) \leq \mathbb{E}_{(\mathbf{y}^s,\mathbf{y}^t)\sim\gamma}\left[\left\|\mathbf{y}^s-\mathbf{y}^t\right\|_1\right] \leq \mathbb{E}_{(\boldsymbol{\tau}^s,\boldsymbol{\tau}^t)\sim\gamma'}\left[\tilde{c}\left(\boldsymbol{\tau}^s,\boldsymbol{\tau}^t\right)\right],$$

which further implies

$$W_1\left(\mathbb{P}^{h^s},\mathbb{P}^{h^t}\right) \leq \inf_{\gamma'\in\Gamma\left(G^1_{\#}\mathbb{P}^s,T^1_{\#}\mathbb{P}^t\right)}\mathbb{E}_{(\boldsymbol{\tau}^s,\boldsymbol{\tau}^t)\sim\gamma'}\left[\tilde{c}\left(\boldsymbol{\tau}^s,\boldsymbol{\tau}^t\right)\right] = W_{\tilde{c}}\left(G^1_{\#}\mathbb{P}^s,T^1_{\#}\mathbb{P}^t\right) \leq W_{\tilde{c},p}\left(G^1_{\#}\mathbb{P}^s,T^1_{\#}\mathbb{P}^t\right).$$

ii) Using the same derivation as in (i) in which T^1 is replaced by L, we achieve

$$\begin{aligned}
\left\|\left[p^s(y=i)-p^t(y=i)\right]_{i=1}^{C}\right\|_1 &\leq R_1^s\left(h^s\right)+R_1^t\left(h^t\right)+W_{\tilde{c},p}\left(G^1_{\#}\mathbb{P}^s,L_{\#}\mathbb{P}^t\right) \\
&\leq R_1^s\left(h^s\right)+R_1^t\left(h^t\right)+W_{\tilde{c},p}\left(G^1_{\#}\mathbb{P}^s,T^1_{\#}\mathbb{P}^t\right)+W_{\tilde{c},p}\left(T^1_{\#}\mathbb{P}^t,L_{\#}\mathbb{P}^t\right).
\end{aligned}$$

□

We note that our proof is still applicable if we generalize $\|\cdot\|_1$ to any metric d in Δ_C, which can be decomposed $d\left(\mathbf{y}^s,\mathbf{y}^t\right) = \sum_{i=1}^{C} d_i\left(\mathbf{y}_i^s,\mathbf{y}_i^t\right)$.

3 Experiments

3.1 Ablation Study

3.1.1 Experiment on Synthetic Data

Synthetic Dataset for the Source and Target Domains

We generate two synthetic labeled datasets for the source and target domains. We generate the $10,000$ data examples of the source dataset from the mixture of two Gaussian distributions: $p^s(\mathbf{x}) = \pi_1^s \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_1^s, \Sigma_1^s) + \pi_2^s \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_2^s, \Sigma_2^s)$ where $\pi_1^s = \pi_2^s = \frac{1}{2}$, $\boldsymbol{\mu}_1^s = [1, 1, ..., 1] \in \mathbb{R}^{10}$, $\boldsymbol{\mu}_2^s = [2, 2, ..., 2] \in \mathbb{R}^{10}$ and $\Sigma_1^s = \Sigma_2^s = \mathbb{I}_{10}$. Similarly, we generate the another $10,000$ data examples of the target dataset from the mixture of two Gaussian distributions: $p^t(\mathbf{x}) = \pi_1^t \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_1^t, \Sigma_1^t) + \pi_2^t \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_2^t, \Sigma_2^t)$ where $\pi_1^t = \frac{1}{3}$, $\pi_2^t = \frac{2}{3}$, $\boldsymbol{\mu}_1^t = [4, 4, ..., 4] \in \mathbb{R}^{10}$, $\boldsymbol{\mu}_2^t = [5, 5, ..., 5] \in \mathbb{R}^{10}$ and $\Sigma_1^t = \Sigma_2^t = \mathbb{I}_{10}$. For each data example in the source and target domains, we assign label $y = 0$ if this data example is generated from the first Gauss and $y = 1$ if this data example is generated from the second Gauss using Bayes' s rule.

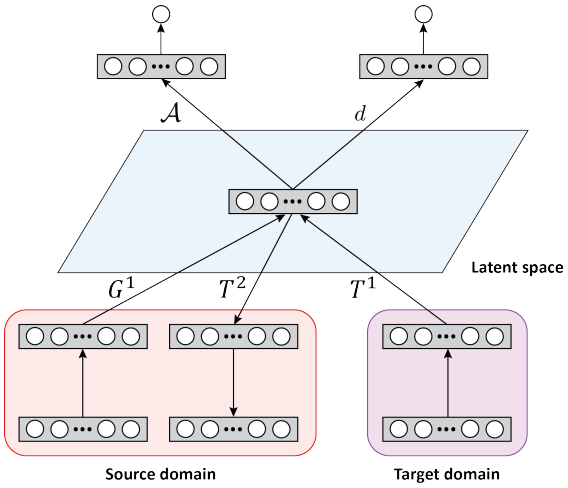

Figure 2: Architecture of networks for deep domain adaptation on the synthetic datasets.

Deep Domain Adaptation on the Synthetic Dataset

Figure 2 shows the architectures of networks used in our experiments on the synthetic datasets. Two generators G^1, T^1 with the same architectures ($10 \to 5\,(\text{ReLu}) \to 5\,(\text{ReLu})$) map the source and target data to the intermediate joint layer. Note that different from other works in deep domain adaptation, we did not tie G^1 and T^1. The network T^2 with the architecture ($10 \to 5\,(\text{ReLu}) \to 5\,(\text{ReLu})$) maps from the intermediate joint layer to the source and target domains respectively. To break the gap between the source and target domains in the joint layer, we employ GAN principle [13, 10] wherein we invoke a discriminator network d ($5 \to 5\,(\text{ReLu}) \to 1\,(\text{sigmoid})$) to discriminate the source and target data examples in the joint space. The classifier network $\mathcal{A}$ ($5 \to 5\,(\text{ReLu}) \to 1\,(\text{sigmoid})$) is employed to classify the labeled source data examples. To approximate the $0/1$ cost function, we use the modified sigmoid function [23]: $c_\gamma(\mathbf{x}, \mathbf{x}') = 2/\left[1 + \exp\left\{-\gamma \left\|\mathbf{x} - \mathbf{x}'\right\|_2\right\}\right] - 1$ with $\gamma = 100$. It can be seen that when $\gamma \to +\infty$, the cost function c_γ approaches the $0/1$ cost function. More specifically, we need to update $G^1, T^1, T^2, \mathcal{A}$, and d as

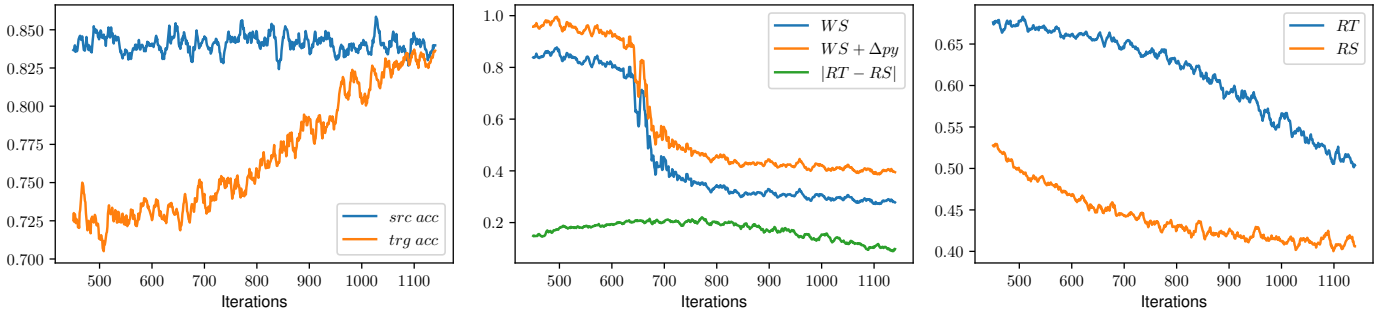

Figure 3: Left: the accuracies on the source and target datasets. Middle: the plots of three terms in Theorem 4. Right: the plot of empirical losses on the source and target datasets.

follows:

$$\left(G^1, T^1, T^2, \mathcal{A}\right) = \underset{G^1, T^1, T^2, \mathcal{A}}{\operatorname{argmin}} \ \mathcal{I}\left(G^1, T^1, T^2, \mathcal{A}\right) \text{ and } d = \underset{d}{\operatorname{argmax}}\, \mathcal{J}\left(d\right),$$

where α is set to 0.1 and we have defined

$$\begin{aligned}
\mathcal{I}\left(G^1, T^1, T^2, \mathcal{A}\right) = & \\
& + \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[c_\gamma\left(\mathbf{x}, T^2\left(G^1\left(\mathbf{x}\right)\right)\right)\right] + \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^s}\left[\ell\left(y, \mathcal{A}\left(G^1\left(\mathbf{x}\right)\right)\right)\right] \\
& + \alpha\left[\mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[\log\left(d\left(G^1\left(\mathbf{x}\right)\right)\right)\right] + \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[\log\left(1 - d\left(T^1\left(\mathbf{x}\right)\right)\right)\right]\right] \\
\mathcal{J}\left(d\right) = & \ \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[\log\left(d\left(G^1\left(\mathbf{x}\right)\right)\right)\right] + \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[\log\left(1 - d\left(T^1\left(\mathbf{x}\right)\right)\right)\right].
\end{aligned}$$

Based on the classifier $\mathcal{A}$ on the joint space, we can identify the corresponding hypotheses on the source and target domains as: $h^s\left(\mathbf{x}\right) = \mathcal{A}\left(G^1\left(\mathbf{x}\right)\right)$ and $h^t\left(\mathbf{x}\right) = \mathcal{A}\left(T^1\left(\mathbf{x}\right)\right)$.

Verification of Our Theory for Unsupervised Domain Adaptation

In this experiment, we assume that none of data example in the target domain has label. We measure three terms, namely $\left|R\left(h^t\right) - R\left(h^s\right)\right|$, $W\left(\mathbb{P}^s, \mathbb{P}^\#\right)$ and $\mathbb{E}_{\mathbb{P}^t}\left[\left\|\Delta p\left(y \mid \mathbf{x}\right)\right\|_1\right]$ ($M = 1$ since we are using the logistic loss) as defined in Theorem 4 across the training progress. Actually, we approximate $R\left(h^t\right), R\left(h^s\right)$ using the corresponding empirical losses. As shown in Figure 3 (middle), the green plot is always above the blue plot and this empirically confirms the inequality in Theorem 4. Furthermore, the fact that three terms consistently decrease across the training progress indicates an improvement when $\mathbb{P}^\#$ is shifting toward $\mathbb{P}^s$. This improvement is also reflected in Figure 3 (left and right) wherein the target accuracy and empirical loss gradually increase and decrease accordingly.

3.1.2 The Effect of Class Alignment in the Joint Space.

In this experiment, we inspect the influence of the harmony of two labeling assignment mechanisms to the predictive performance. In particular, we assume that a portion ($r = 5\%,\ 15\%,\ 25\%,\ 50\%$) of the target domain has label and consider two settings: i) the labels of the target and source domains are totally properly matched in the joint space (i.e., 0 matches 0, 1 matches 1,..., and 9 matches 9) and ii) the labels of the target and source domains are totally improperly matches in the joint space (i.e., 0 matches 1, 1 matches 2,..., and 9 matches 0).

To push a specific labeled portion of the target domain to the corresponding label portion of the source domain in the joint space (the label i to i in the first setting and the label i to $(i+1) \bmod 10$ in the second setting for $i = 0, 1, \ldots, 9$), we again make use of

GAN principle and employ additional discriminators to push the corresponding labeled portions together. Note that the parameters of the additional discriminators and the primary discriminator (used to push the target data toward source data in the joint space) are tied up to the penultimate layer.

It can be observed from Table 1 that for the case of proper matching, when increasing the ratio of labeled portion, we increase the chance to match the corresponding labeled portions properly, hence significantly improving the predictive performance. In contrast, for the case of improper matching, when increasing the ratio of labeled portion, we increase the chance to match the corresponding labeled portions improperly, hence significantly reducing the predictive performance.

Table 1: The variation of predictive performance in percentage as increasing the ratio of labeled portion when the labels of the target domain are properly or improperly matched to those in the source domain. Note that we emphasize in bold and italic/bold the best and worse performance.

	Proper match				**Improper match**				**Base**
$\boldsymbol{r}$	5%	15%	25%	50%	5%	15%	25%	50%	0%
MNIST→MNIST-M	86.4	88.8	92.9	**93.2**	75.5	70.2	64.5	***58.4***	81.5
SVHN→MNIST	72.3	74.1	76.2	**77.5**	69.8	60.8	56.8	***56.4***	71.0

3.2 Experimental Setting for our LAMDA

3.2.1 The Objective Function of LAMDA

Note that in our implementation, to reduce the model complexity, we set $G^1 = T^1 = G$, $T^2 = T$, $S = \mathcal{A}$ (S is a transportation probability network which is shared weights with the classifier $\mathcal{A}$). Let us further denote:

$$\mathcal{L}_{\mathcal{A}} \coloneqq \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^s}\left[\ell\left(y, \mathcal{A}\left(G\left(\mathbf{x}\right)\right)\right)\right],$$

$$\begin{aligned}\mathcal{L}_g \coloneqq{}& \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[\log\left(1 - d_{C+1}\left(G(\mathbf{x})\right)\right)\right] + \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[\log d_{C+1}\left(G\left(\mathbf{x}\right)\right)\right] \\ &+ \alpha\mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[-\sum_{i=1}^{C}\mathcal{A}_i\left(\mathbf{x}\right)\log d_i\left(G\left(\mathbf{x}\right)\right)\right] + \beta R\left(T, G\right) + \mathcal{L}_{\mathcal{A}},\end{aligned}$$

where $R\left(T, G\right)$ is the reconstruction term defined as

$$R\left(T, G\right) \coloneqq \mathbb{E}_{\mathbb{P}^s}\left[\left\|T\left(G\left(\mathbf{x}\right)\right) - \mathbf{x}\right\|_2^2\right].$$

$$\begin{aligned}\mathcal{L}_d \coloneqq \sum_{i=1}^{C}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^s\wedge y=i}\left[\log d_i\left(G\left(\mathbf{x}\right)\right)\right] &+ \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^t}\left[\log d_{C+1}\left(G\left(\mathbf{x}\right)\right)\right] \\ &+ \mathbb{E}_{\mathbf{x}\sim\mathbb{P}^s}\left[\log\left(1 - d_{C+1}\left(G\left(\mathbf{x}\right)\right)\right)\right].\end{aligned}$$

To update G, T and $\mathcal{A}$, we solve:

$$\min_{G,T,\mathcal{A}}\mathcal{L}_g.$$

To update d, we solve:

$$\max_{d}\mathcal{L}_d.$$

Table 2: The full experimental results in percent of our LAMDA and the baselines on digits, traffic sign, and natural image datasets.

Source	**MNIST**	**USPS**	**MNIST**	**SVHN**	**MNIST**	**DIGITS**	**SIGNS**	**CIFAR**	**STL**
Target	**USPS**	**MNIST**	**MNIST-M**	**MNIST**	**SVHN**	**SVHN**	**GTSRB**	**STL**	**CIFAR**
MMD [21]	-	-	76.9	71.1	-	88.0	91.1	-	-
DANN [10]	-	-	81.5	71.1	35.7	90.3	88.7	-	-
DRCN [11]	-	-	-	82.0	40.1	-	-	66.4	58.7
DSN [4]	-	-	83.2	82.7	-	91.2	93.1	-	-
kNN-Ad [27]	-	-	86.7	78.8	40.3	-	-	-	-
PixelDA [3]	-	-	98.2	-	-	-	-	-	-
ATT [24]	-	-	94.2	86.2	52.8	92.9	96.2	-	-
Π-model [9]	-	-	-	92.0	71.4	94.2	98.4	76.3	64.2
ADDA [30]	89.4	90.1	-	76.0	-	-	-	-	-
CyCADA [15]	95.6	96.5	-	90.4	-	-	-	-	-
MSTN [33]	92.9	97.6	-	91.7	-	-	-	-	-
CDAN [22]	95.6	98.0	-	89.2	-	-	-	-	-
MCD [25]	94.2	94.1	-	96.2	-	-	94.4	-	-
PFAN [5]	95.0	-	-	93.9	57.6	-	-	-	-
DADA [28]	96.1	96.5	-	95.6	-	-	-	-	-
DeepJDOT[7]	95.7	96.4	92.4	96.7	30.8	84.2	70.0	61.6	49.6
DASPOT [34]	97.5	96.5	94.9	96.2	-	-	-	-	-
GPDA [16]	96.5	96.4	-	98.2	-	-	96.2	-	-
SWD [18]	98.1	97.1	90.9	98.9	49.5	88.7	98.6	65.3	52.1
rRevGrad+CAT [8]	94.0	96.0	-	98.8	-	-	-	-	-
SHOT [20]	98.0	**98.4**	-	98.9	-	-	-	-	-
RWOT [35]	98.5	97.5	-	98.8	-	-	-	-	-
LAMDA	**99.5**	98.3	**98.4**	**99.5**	**82.1**	**95.9**	**99.2**	**78.0**	**71.6**

3.2.2 Experimental Datasets

Digit datasets

MNIST. The dataset is commonly used in domain adaptation literature. To adapt from MNIST to MNIST-M or SVHN, the MNIST images are replicated from single greyscale channel to obtain digit images which has three channels.
MNIST-M. Following by the implementation in [10], we generate the MNIST-M images by replacing the black background of MNIST images by the color ones and obtain the same number of training and test samples as the MNIST dataset.
SVHN. The dataset consists of images obtained by detecting house numbers from Google Street View images. This dataset is a benmark for recognizing digits and numbers in real-world images.
DIGITS. There are roughly 500,000 images are generated using various data augmentation schemes, i.e., varying the text, positioning, orientation, background, stroke color, and the amount of blur.

We compare our LAMDA with renown baselines especially OT-based ones (e.g., SWD [18], DeepJDOT [7], DASPOT [34], ETD [19] and RWOT [35]). As shown in Table 2, LAMDA outperforms other baselines on most of digit datasets. It is noticeable that although the transfer task MNIST→SVHN is extremely challenging in which the source dataset includes grayscale handwritten digits whereas the target dataset is created by real-world digits, our LAMDA is still capable of matching the gap between source and target domains and outperforms the second-best method by a sizeable margin (10.7%).

Traffic sign datasets

SIGNS. A synthetic dataset for traffic sign recognition. Images are collected from Wikipedia and then applied various types of transformations to generate 100,000 images for training and test.
GTSRB. Road sign images are extracted from videos recorded on different road types in Germany. We preprocess the data by croping out the region of interest of each image, and then scale them to a resolution of 32×32.

Natural scene datasets

CIFAR. The CIFAR-10 [17] dataset includes 50,000 training images and 10,000 test images. However, to adapt with STL dataset, we base on [9] to remove one non-overlapping class ("frog"). The numbers of training examples and test examples therefore are reduced to 45,000 and 9,000 respectively.
STL. Similar to CIFAR-10, we remove class named "monkey" to obtain a 9-class classification problem. Also, STL-10 images are down-scaled to a resolution of 96×96 to 32×32.

Object recognition datasets

Office-Home consists of roughly 15,500 images in a total of 65 object classes and belonging to 4 different domains: Artistic (**Ar**), Clip Art (**Cl**), Product (**Pr**) and Real-world (**Rw**).
Office-31 is a popular dataset for domain adaptation that contains 3 domains Amazon (**A**), Webcam (**W**), and DSLR (**D**). There are 31 common classes for all domains and the total number of images is 4,110.
ImageCLEF-DA contains three domains: Caltech-256 (**C**), ImageNet ILSVRC 2012 (**I**) and Pascal VOC 2012 (**P**). There are total 600 images in each domain and 12 common classes. We follow the work in [19] to evaluate 6 adaptation tasks.

We resize the resolution of each sample in *digits*, *traffic sign*, and *natural image* datasets to 32×32, and normalize the value of each pixel to the range of $[-1, 1]$. For *object recognition* datasets, we use features have 2048 dimensions extracted from ResNet-50 [14] pretrained on ImageNet.

Table 3: Small, medium and large network architecture of LAMDA. We use the small network for object recognition datasets, medium network for digits and traffic sign, and the large one for natural scene datasets. The parameter a for Leaky ReLU (lReLU) activation function is set to 0.1.

Architecture	Small Network	Medium Network	Large Network
Input size	2048	$32 \times 32 \times 3$	$32 \times 32 \times 3$
Generator G		instance normalization	instance normalization
	256 dense, ReLU	3×3 conv. 64 lReLU	3×3 conv. 96 lReLU
	dropout, $p = 0.5$	3×3 conv. 64 lReLU	3×3 conv. 96 lReLU
	Gaussian noise, σ = 1	3×3 conv. 64 lReLU	3×3 conv. 96 lReLU
		2×2 max-pool, stride 2	2×2 max-pool, stride 2
		dropout, $p = 0.5$	dropout, $p = 0.5$
		Gaussian noise, σ = 1	Gaussian noise, σ = 1
		3×3 conv. 64 lReLU	3×3 conv. 192 lReLU
		3×3 conv. 64 lReLU	3×3 conv. 192 lReLU
		3×3 conv. 64 lReLU	3×3 conv. 192 lReLU
		2×2 max-pool, stride 2	3×3 max-pool, stride 2
		dropout, $p = 0.5$	dropout, $p = 0.5$
		Gaussian noise, σ = 1	Gaussian noise, σ = 1
Classifier $\mathcal{A}$	C dense, softmax	3×3 conv. 64 lReLU	3×3 conv. 192 lReLU
		3×3 conv. 64 lReLU	3×3 conv. 192 lReLU
		3×3 conv. 64 lReLU	3×3 conv. 192 lReLU
		global average pool	global average pool
		C dense, softmax	C dense, softmax

3.2.3 Network Architectures

We use small and large network architecture for specific datasets, which are described in Table 3 and 4. Noticeably, batch normalization layers are applied on the top of convolutional layers (6 for the generator and 3 for the classifier) to prevent the overfitting. For *Office-31*, *Office-Home*, and *ImageCLEF-DA*, we removed the dense layers of the pretrained models and replaced by two dense layers (i.e., the first layer has 256 neurons and the second one has C neurons where C is the number of classes).

Table 4: The architecture of discriminator d.

Digits, traffic sign and natural scene datasets	Object recognition datasets
3×3 conv. 64 lReLU	$C + 1$ dense, softmax
3×3 conv. 64 lReLU	
3×3 conv. 64 lReLU	
global average pool	
$C + 1$ dense, softmax	

3.2.4 Hyperparameter setting

We apply Adam Optimizer ($\beta_1 = 0.5$,$\beta_2 = 0.999$) with the learning rate 0.001 *digits*, *traffic sign* and *natural scene* datasets, whereas 0.0001 is the learning rate for *object recognition* datasets. All experiements was trained for 20000 iterations on *Office-31*, *Office-*

Home, and *ImageCLEF-DA* and 80000 for the other datasets. The batch size for each dataset is set to 128. We set $\beta = 0$, $\alpha = 0.5$ as described in the ablation study, and γ is searched in $\{0.1, 0.5\}$. We implement our LAMDA in Python (version 3.5) using Tensorflow (version $1.9.0$) [1] and run our experiments on a computer with a CPU named Intel Xeon Processor E5-1660 which has 8 cores at 3.0 GHz and 128 GB of RAM, and a GPU called NVIDIA GeForce GTX Titan X with 12 GB memory.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Mach. Learn.*, 79(1-2):151–175, May 2010.

[3] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3722–3731, 2017.

[4] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In *Advances in neural information processing systems*, pages 343–351, 2016.

[5] C. Chen, W. Xie, W. Huang, Y. Rong, X. Ding, Y. Huang, T. Xu, and J. Huang. Progressive feature alignment for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 627–636. Computer Vision Foundation / IEEE, 2019.

[6] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems*, pages 3730–3739, 2017.

[7] B. B. Damodaran, B. Kellenberger, R. Flamary, D. Tuia, and N. Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*, volume 11208 of *Lecture Notes in Computer Science*, pages 467–483, 2018.

[8] Z. Deng, Y. Luo, and J. Zhu. Cluster alignment with a teacher for unsupervised domain adaptation, 2019.

[9] G. French, M. Mackiewicz, and M. Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018.

[10] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 1180–1189, 2015.

[11] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016.

[12] A. L. Gibbs and F. E. Su. On choosing and bounding probability metrics. *INTERNAT. STATIST. REV.*, pages 419–435, 2002.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] J. Hoffman, E. Tzeng, T. Park, J-Y Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning*, pages 1989–1998, 2018.

[16] M. Kim, P. Sahu, B. Gholami, and V. Pavlovic. Unsupervised visual domain adaptation: A deep max-margin gaussian process approach. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4375–4385, 2019.

[17] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

[18] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 10285–10295. Computer Vision Foundation / IEEE, 2019.

[19] M. Li, Y. Zhai, Y. Luo, P. Ge, and C. Ren. Enhanced transport distance for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[20] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation, 2020.

[21] M. Long, Y. Cao, J. Wang, and M. Jordan. Learning transferable features with deep adaptation networks. In F. Bach and D. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 97–105, Lille, France, 2015.

[22] M. Long, Z. Cao, J. Wang, and M. I. Jordan. Conditional adversarial domain adaptation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 1640–1650. Curran Associates, Inc., 2018.

[23] T. Nguyen and S. Sanner. Algorithms for direct 0–1 loss optimization in binary classification. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1085–1093, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

[24] K. Saito, Y. Ushiku, and T. Harada. Asymmetric tri-training for unsupervised domain adaptation. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2988–2997. JMLR. org, 2017.

[25] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[26] F. Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, pages 99–102, 2015.

[27] O. Sener, H O Song, A. Saxena, and S. Savarese. Learning transferrable representations for unsupervised domain adaptation. In *Advances in Neural Information Processing Systems*, pages 2110–2118, 2016.

[28] Hui Tang and Kui Jia. Discriminative adversarial domain adaptation, 2019.

[29] I. O. Tolstikhin, O. Bousquet, S. Gelly, and B. Schölkopf. Wasserstein auto-encoders. *CoRR*, abs/1711.01558, 2018.

[30] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2962–2971, 2017.

[31] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, second edition, November 1999.

[32] C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2008.

[33] S. Xie, Z. Zheng, L. Chen, and C. Chen. Learning semantic representations for unsupervised domain adaptation. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5423–5432. PMLR, 10–15 Jul 2018.

[34] Y. Xie, M. Chen, H. Jiang, T. Zhao, and H. Zha. On scalable and efficient computation of large scale optimal transport. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6882–6892, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[35] R. Xu, P. Liu, L. Wang, C. Chen, and J. Wang. Reliable weighted optimal transport for unsupervised domain adaptation. In *CVPR 2020*, June 2020.