Domain Adaptation with Optimal Transport for Extended Variable Space

Toshimitsu Aritake
The Institute of Statistical Mathematics
Tachikawa, Tokyo, Japan
aritake@ism.ac.jp

Hideitsu Hino
The Institute of Statistical Mathematics
Tachikawa, Tokyo, Japan
RIKEN AIP, Chuo-ku, Tokyo, Japan
hino@ism.ac.jp

***Abstract*—Domain adaptation aims to transfer knowledge of labeled instances obtained from a source domain to a target domain to fill the gap between the domains. Most domain adaptation methods assume that the source and target domains have the same dimensionality. Methods that are applicable when the number of features for each sample is different in each domain have rarely been studied, especially when no label information is given for the test data obtained from the target domain. In this paper, it is assumed that common features exist in both domains and that extra (new additional) features are observed in the target domain; hence, the dimensionality of the target domain is higher than that of the source domain. To leverage the homogeneity of the common features, the adaptation between the source and target domains is formulated as an optimal transport (OT) problem. In addition, a learning bound in the target domain for the proposed OT-based method is derived. The experiments with simulated and real-world data show that our proposed algorithm is able to obtain better model for the target domain by considering the extra features given for the target domain.**

***Index Terms*—Heterogeneous Domain Adaptation, Optimal Transport, Transfer Learning**

I. Introduction

The goal of supervised learning is to build a model f that maps the features $\boldsymbol{x}$ to its corresponding label y from a given training dataset $\mathcal{D}_S$ to estimate the label of the unlabeled test dataset $\mathcal{D}_T$. Let the distributions of the training data be $\mathcal{P}_S(\boldsymbol{x}, y)$, and the test data be $\mathcal{P}_T(\boldsymbol{x}, y)$, respectively. However, when $\mathcal{P}_S(\boldsymbol{x}, y) \neq \mathcal{P}_T(\boldsymbol{x}, y)$, the difference leads loss of accuracy of the trained model on the test data. It is still possible to train a model that accurately predicts the label of the test data by considering the difference in the distributions. Domain adaptation techniques are used to consider the difference in the distributions by transferring information from the source domain to the target domain [1], [2]. Henceforth, we refer to the domains of the training and test data as the source and target domains, respectively. In general, domain adaptation aims to match the joint distributions of $(\boldsymbol{x}, y)$ in the source and target domains.

Most domain adaptation methods assume spaces of the same dimensionality as the source and target domains. This type of domain adaptation is called *homogeneous* domain adaptation, and these methods cannot be applied when the number of the features is different for each domain, such as when images of different sizes are observed in each domain. Domain adaptation for spaces of different dimensionalities is called *heterogeneous* domain adaptation. In the literature, only a few methods have been proposed for unsupervised heterogeneous domain adaptation.

In this paper, we consider an unsupervised heterogeneous domain adaptation problem where both the source and target domains have common features and extra (new additional) features are observed in the target domain. Here, we assume that data is tabular data and that it is known whether each feature is a common feature or an extra feature. Also, since each common feature represents the same attribute in the source and target domains, the homogeneity of common features should be considered for domain adaptation. For example, consider the case of measuring the movements of a person with a set of accelerometers. The types of activity are assigned as a label for each observed movement, and these data are used for training. Then, assume that the activities of another person are estimated from the measurements of movements obtained using the same set of accelerometers and additional gyroscopes. In this case, the features obtained by the accelerometers become common features, and the features obtained by the gyroscopes become extra features. The observation of such extra features potentially produce better target data distributions that improves the estimation accuracy of target labels. However, the extra features cannot directly be used for estimation because the extra features are not observed for training data. The goal of this paper is to provide a method that enable the use of extra features for better estimation in the target domain.

However, general heterogeneous domain adaptation methods do not assume the homogeneity of features, and a special case of heterogeneous domain adaptation called hybrid domain adaptation has been studied to address this issue, where it is assumed that the source and target domains have common features, and domain-specific features are also given for each domain. To preserve the homogeneity of the common features, hybrid domain adaptation learns the models used to predict domain-specific features from common features. Then, the learned models are used to estimate the unobserved domain-specific features.

The problem considered in this paper can be seen as a variant of a hybrid domain adaptation problem, in which the domain-specific features are only given for the target domain. In the same manner as for hybrid domain adaptation, the

978-1-7281-8671-9/22/$31.00 ©2022 IEEE

unobserved extra features in the source domain are predicted using common features. Unlike hybrid domain adaptation, our proposed method estimates the unobserved features using optimal transport (OT). Recently, OT has been used for domain adaptation to match the distributions in the source and target domains. However, when the number of features is different between the source and target domains, it is difficult to define an appropriate transport cost for OT.

To solve our problem using OT, it is natural to consider *two-way* OT. Namely, extra features in the source domain are estimated by solving the OT problem from the target domain to the source domain, then the label information in the source domain is transferred to the target domain by solving another OT problem. For these OT problems, we use pseudo-labels as proxies of unobserved true target labels and consider a problem similar to joint distribution optimal transport (JDOT). Namely, in the former OT, the distance between the common features and the mismatch between the source label and the target pseudo-label are used for the transport cost so that the joint distributions of the features and labels are better matched in the source and target domains. Then, in the latter OT, the distance between the extra features is additionally considered. We show that this *two-way* OT is equivalent to *one-way* OT under the assumption that the conditional distribution of an extra feature given a common feature and a label is identical before and after OT. Figure 1 shows the above concept.

We summarize the contributions of this paper:

- We propose an algorithm based on OT for a domain adaptation problem where the domain shift between the source and target domains is caused by the observation of extra (new additional) features and the distribution shift of common features.
- We provide an interpretation of the proposed algorithm that the proposed one-way OT-based algorithm is equivalent to two-way OT.

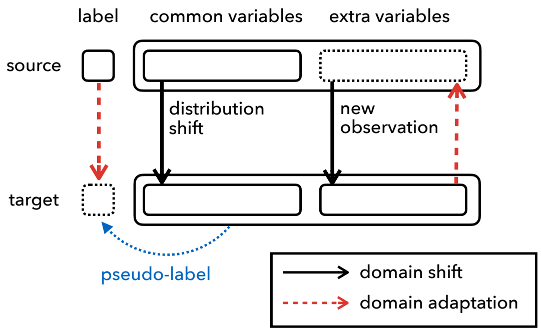


Fig. 1: Conceptual illustration of the proposed domain adaptation method for the observation of extra features. The two-way OT between the source and target domains is considered to estimate the extra features in the source domain and the labels in the target domain. Practically, this two-way OT is solved as one-way OT from the source domain to the target domain.

- We derive a learning bound of the trained model by the proposed method in the target domain. The derived upper bound is based on the Rademacher complexity and the Wasserstein distance between the true and estimated target distributions, and the upper bound also gives an intuitive understanding of the proposed algorithm.

The rest of this paper is organized as follows. In Section II, the related work of domain adaptation and OT is summarized. In Section III, we present the practical algorithm based on one-way OT. Then, we show the equivalence of the proposed method and two-way OT in Section IV. Also, the learning bound of the proposed method is presented. In Section V, we report the results of experiments on synthetic and real-world datasets. Then, we summarize the paper and discuss the limitations of the proposed method and future work in Section VI.

II. Related Work

A. Unsupervised Domain Adaptation

In general, domain adaptation aims to match the joint distributions of the features and the label $(\boldsymbol{x}, y)$ in the source and target domains. Especially, when no labeled data of the target domain are available, the domain adaptation problem is called *unsupervised* domain adaptation, which we consider in this paper.

1) Homogeneous Domain Adaptation: Most domain adaptation methods assume spaces of the same dimensionality as the source and target domains. This type of domain adaptation problem is called *homogeneous* domain adaptation.

In unsupervised homogeneous domain adaptation, the distributions of the source and target domains are matched on the basis of the assumption made for the change in distribution. There are a number of unsupervised domain adaptation methods, which are categorized into several groups. Recall that $\mathcal{P}_S(\boldsymbol{x}, y), \mathcal{P}_T(\boldsymbol{x}, y)$ be the source and target joint distribution of features and label. Then, the covariate shift [3] assumes that $\mathcal{P}_S(y|\boldsymbol{x}) = \mathcal{P}_T(y|\boldsymbol{x})$ and $\mathcal{P}_S(\boldsymbol{x}) \neq \mathcal{P}_T(\boldsymbol{x})$. Therefore, it aims to match the distributions $\mathcal{P}_S(\boldsymbol{x})$ and $\mathcal{P}_T(\boldsymbol{x})$ to match the joint distribution, for example, by importance reweighting [4]. Similarly, the conditional shift or the concept shift [5] assumes either $\mathcal{P}_S(y|\boldsymbol{x}) \neq \mathcal{P}_T(y|\boldsymbol{x})$ or $\mathcal{P}_S(\boldsymbol{x}|y) \neq \mathcal{P}_T(\boldsymbol{x}|y)$, and the target shift (also known as the prior shift) [6], [7] assumes $\mathcal{P}_S(y) \neq \mathcal{P}_T(y)$. Furthermore, recent works have considered to learn domain invariant features for each of these assumptions using deep neural networks including generative adversarial models [8]–[11]. A method to learn models from weakly labeled source samples is also proposed for domain adaptation with cheaper labeling cost [12]. For other domain adaptation methods, see [13], [14] and references therein. These homogeneous domain adaptation methods rely on the assumption that the source and target domains have the same dimensionality; therefore, these methods are not directly applicable when the source and target domains have different dimensionalities.

2) Heterogeneous Domain Adaptation: On the other hand, when the source and target domains have different dimensionalities, the domain adaptation problem is called *heterogeneous* domain adaptation. In the literature, several methods have been proposed to solve heterogeneous domain adaptation problems. However, most heterogeneous domain adaptation methods [15]–[19] require at least partly labeled instances from the target domain, and only a few unsupervised heterogeneous domain adaptation methods have been proposed [20]–[22]. The common strategy for unsupervised heterogeneous domain adaptation is to embed features from the source and target domains to a space of the same dimensionality and consider a homogeneous domain adaptation problem therein. For example, spectral embedding [15], linear embedding [22], and kernel canonical correlation analysis [20] are used for embedding. However, since the features are mixed by embedding, these methods cannot consider the homogeneity of features when the source and target domains have common features.

A special case of heterogeneous domain adaptation called hybrid domain adaptation is studied in [21], [23], [23], where it is assumed that the source and target domains have common features, and domain-specific features are also given for each domain. To consider the homogeneity of the common features, hybrid domain adaptation use the models to predict domain-specific features from common features. The models learned on one domain are used to estimate the unobserved domain-specific features on the other domain. However, it is not always possible to accurately estimate the domain-specific features from the common features. For example, it is difficult to estimate domain-specific features using simple regression models when the distribution of domain-specific features given the common features follow multi-modal distributions.

B. Optimal Transport for Domain Adaptation

Recent works apply OT techniques to match the source and target distributions for domain adaptation [24]–[26]. Optimal transport is a well-established mathematical theory [27] that has been successfully applied to various machine learning tasks involving the transport of a probability distribution. Optimal transport for homogeneous domain adaptation makes the assumption $\mathcal{P}_S(y|\boldsymbol{x}) = \mathcal{P}_T(y|\mathcal{T}(\boldsymbol{x}))$ on the conditional distribution, where $\mathcal{T}$ represents the transport mapping. However, this assumption does not hold in general; therefore, group regularized OT [26] and JDOT [25], which leverages pseudo-labels estimated using the model, are proposed to alleviate this problem. In addition, some works have considered OT problems for heterogeneous feature spaces by defining the transport cost between spaces of different dimensionalities [28], [29]. Although these methods are applicable for heterogeneous domain adaptation, the cost functions defined in these methods do not consider the homogeneity of features.

III. Problem Formulation

A. Optimal Transport in Domain Adaptation

We begin with basics on OT. Let μ_1 and μ_2 be the probability measure on a space Ω. Given a cost function $c : \Omega \times \Omega \to \mathbb{R}_+$, the OT problem is formulated as a problem of seeking a coupling $\pi \in \Pi(\mu_1, \mu_2)$ between μ_1 and μ_2 that minimizes the total transport cost:

$$\inf_{\pi \in \Pi(\mu_1,\mu_2)} \int_{\Omega\times\Omega} c(\boldsymbol{x}, \boldsymbol{x}') d\pi(\mu_1, \mu_2), \tag{1}$$

where $\boldsymbol{x} \sim \mu_1$ and $\boldsymbol{x}' \sim \mu_2$. Here, $\Pi(\mu_1, \mu_2)$ is the set of couplings, that is, joint probability distributions with marginals μ_1 and μ_2. In general, a distance function between the samples is used as a cost function c.

In domain adaptation, OT is used to match the source distribution $\mathcal{P}_S$ and the target distribution $\mathcal{P}_T$. In particular, the OT problem for unsupervised domain adaptation is formulated as OT between two marginal distributions $\mathcal{P}_S(\boldsymbol{x})$ and $\mathcal{P}_T(\boldsymbol{x})$ under the assumption $\mathcal{P}_S(y|\boldsymbol{x}) = \mathcal{P}_T(y|\mathcal{T}(\boldsymbol{x}))$, where $\mathcal{T}$ represents an optimal transport map. However, this assumption does not always hold; hence, JDOT [25] considers the distance between features as well as discrepancy of the labels as transport cost so that $\mathcal{P}_S(\boldsymbol{x}, y)$ and $\mathcal{P}_T(\boldsymbol{x}, y)$ are better matched. Namely, the cost function $c(\boldsymbol{x}_1, y_1; \boldsymbol{x}_2, y_2) = \alpha d(\boldsymbol{x}_1, \boldsymbol{x}_2) + \mathcal{L}(y_1, y_2)$, where $\mathcal{L}$ is the discrepancy between labels y_1 and y_2, is used for OT. Since the target label is not observed in unsupervised domain adaptation, the label estimated as $\hat{y} = f(\boldsymbol{x})$ is used as a proxy of the target label. The technique of using the output of the model as the proxy of the true label is known as pseudo-labeling, and a number of methods that use pseudo-labeling have been proposed for various learning problems including unsupervised domain adaptation [25], [30]–[36]. Furthermore, JDOT also learns a model f that estimates the pseudo-label by minimizing the cost in Eq. (1).

B. Domain Adaptation with Optimal Transport for Extended Variable Space

We consider an unsupervised domain adaptation problem, where both the source and target domains have common features and extra features are observed in the target domain. In this case, it is difficult to define the cost between the source and target features because they are different dimensional vectors. Here, instead of directly defining the cost between the source and target variables, we consider to use the distance between the common variables and the discrepancy between the source label and target pseudo-label as JDOT. We remark that the extra variables are considered by the pseudo-label.

Let $\Omega^c \times \mathcal{C}$ be the source domain, which is a direct product of the space of the common features, Ω^c, and label space $\mathcal{C}$. Also, we define $\Omega^t \equiv \Omega^c \times \Omega^e$, where Ω^e is the space of the extra features, and let $\Omega^t \times \mathcal{C}$ be the target domain. Namely, the spaces of the common features are identical in the source and target domains, and the extra features are only observed in the target domain. Note that even though the spaces of the common features are identical, the distribution on Ω^c can be different between the source and target domains. Here, we denote the probability distributions of the source and target domains as $\mathcal{P}_S(\boldsymbol{x}^c, y)$ and $\mathcal{P}_T(\boldsymbol{x}^c, \boldsymbol{x}^e, y)$, respectively, or $\mathcal{P}_S$ and $\mathcal{P}_T$ for short. Then, we define the training set $\mathcal{D}_S = \{(\boldsymbol{x}^c_{si}, y_{si})\}_{i=1}^{N_s}$

that consists of samples $(\boldsymbol{x}^c_{si}, y_{si}) \sim \mathcal{P}_S$. Similarly, we define the test set by the partial observation $\mathcal{D}_T = \{(\boldsymbol{x}^c_{ti}, \boldsymbol{x}^e_{t,i})\}_{i=1}^{N_t}$ of a sample $(\boldsymbol{x}^c_{ti}, \boldsymbol{x}^e_{ti}, y_{ti}) \sim \mathcal{P}_T$ $(i = 1, \ldots, N_t)$, where the true label y_{ti} is not observed. Since the label y of the target distribution is not observed, we define the estimated target probability distribution as $\mathcal{P}_T^f(\boldsymbol{x}^c, \boldsymbol{x}^e, \hat{y})$, where the label y is replaced by the pseudo-label $\hat{y} = f(\boldsymbol{x}^c, \boldsymbol{x}^e)$. Note here that $\mathcal{P}_T^f(\boldsymbol{x}^c, \boldsymbol{x}^e) = \mathcal{P}_T(\boldsymbol{x}^c, \boldsymbol{x}^e)$ holds for marginal distributions. Similarly, we define the estimated test set $\mathcal{D}_T^f = \{(\boldsymbol{x}^c_{ti}, \boldsymbol{x}^e_{ti}, \hat{y}_i)\}_{i=1}^{N_t}$, where $\hat{y}_i = f(\boldsymbol{x}^c_{ti}, \boldsymbol{x}^e_{ti})$.

To transfer the label information from the source domain to the target domain, we consider the following problem:

$$\begin{aligned}\pi^*, \hat{f} = \inf_{\pi\in\Pi(\mathcal{P}_S,\mathcal{P}_T^f), f\in\mathcal{F}} &\int_{(\Omega_s\times\mathcal{C})\times(\Omega_t\times\mathcal{C})} \\ &\mathcal{E}_\alpha(\boldsymbol{x}^c_s, y_s; \boldsymbol{x}^c_t, \boldsymbol{x}^e_t, y_t) d\pi(\boldsymbol{x}^c_s, y_s; \boldsymbol{x}^c_t, \boldsymbol{x}^e_t, y_t), \quad (2)\end{aligned}$$

where $\Pi(\mathcal{P}_S, \mathcal{P}_T^f)$ is the set of transportation plans between the probability densities $\mathcal{P}_S$ and $\mathcal{P}_T^f$ and $\mathcal{F}$ is a set of models. Here, we use the following cost function for the transport:

$$\mathcal{E}_\alpha(\boldsymbol{x}^c_s, y_s; \boldsymbol{x}^c_t, \boldsymbol{x}^e_t, y_t) \equiv \alpha d(\boldsymbol{x}^c_s, \boldsymbol{x}^c_t) + \mathcal{L}(y_s, y_t), \quad (3)$$

which is the sum of the distance between the common features $d(\boldsymbol{x}^c_s, \boldsymbol{x}^c_t)$ and the discrepancy of the label $\mathcal{L}(y_s, y_t)$. Note here that the extra feature $\boldsymbol{x}^e_t$ is only used to estimate the pseudo-label $\hat{y} = f(\boldsymbol{x}^c, \boldsymbol{x}^e)$. Although the choice of the metric d is arbitrary, here we assume that d is a square distance $d(\boldsymbol{x}^c_s, \boldsymbol{x}^c_t) = \|\boldsymbol{x}^c_s - \boldsymbol{x}^c_t\|_2^2$ for simplicity. Here, $\alpha \in \mathbb{R}_+$ is a hyperparameter that determines the relative importance of $d(\boldsymbol{x}^c_s, \boldsymbol{x}^c_t)$ to $\mathcal{L}(y_s, y_t)$. By solving the above optimization problem, the source labels are transferred to the target domain, and model $f \in \mathcal{F}$ is trained to map the pair of common and extra features to their corresponding transferred labels.

In practice, a finite number of samples obtained from the source and target distributions can be used to solve the OT problem. Therefore, instead of solving the OT problem problem between the source and target distributions, we consider the discrete OT problem between the empirical distributions of the training and test data. The optimization problem Eq. (2) is rewritten as

$$\hat{\pi}^*, \hat{f}_s = \mathop{\arg\min}_{\hat{\pi}\in\hat{\Pi}(\mathcal{D}_S,\mathcal{D}_T^f), f\in\mathcal{F}} \sum_{i=1}^{N_s}\sum_{j=1}^{N_t} \hat{\pi}_{ij}\mathcal{E}_\alpha(\boldsymbol{x}^c_{si}, y_{si}; \boldsymbol{x}^c_{tj}, \hat{y}_{tj}),$$

where $\hat{y}_{tj} = f(\boldsymbol{x}^c_{tj}, \boldsymbol{x}^e_{tj})$ and $\hat{\Pi}(\mathcal{D}_S, \mathcal{D}_T^f)$ is a set of discrete OT plans from the dataset $\mathcal{D}_S$ to the dataset $\mathcal{D}_T^f$, and is defined as

$$\hat{\Pi}(\mathcal{D}_S, \mathcal{D}_T^f) \equiv \left\{\hat{\pi} \in \mathbb{R}_+^{N_s\times N_t} \middle| \textstyle\sum_i \hat{\pi}_{ij} = \frac{1}{N_t}, \sum_j \hat{\pi}_{ij} = \frac{1}{N_s}\right\}.$$

This optimization problem is non-convex and computationally intractable; therefore, alternating optimization is used to solve the problem in the same manner as in conventional methods that use pseudo-labeling. At the nth iteration, the optimization problem with respect to $\pi \in \hat{\Pi}(\mathcal{D}_S, \mathcal{D}_T^f)$ with fixed $\hat{f}_s^{(n)} \in \mathcal{F}$ becomes a discrete OT problem, where the transport cost $\mathcal{E}_\alpha$ is calculated using the pseudo-label estimated as $\hat{y}_{tj}^{(n)} = \hat{f}_s^{(n)}(\boldsymbol{x}^c_{tj}, \boldsymbol{x}^e_{tj})$. When we calculate the transportation cost as $E_{ij}^{(n)} = \mathcal{E}_\alpha(\boldsymbol{x}^c_{si}, y_{si}; \boldsymbol{x}^c_{tj}, \hat{y}_{tj}^{(n)})$, the optimization problem becomes linear programming with equality constraints as,

$$\begin{aligned}\hat{\pi}_n^* = &\mathop{\arg\min}_{\hat{\pi}\in\hat{\Pi}(\mathcal{D}_S,\mathcal{D}_T^f)} \sum_{i=1}^{N_s}\sum_{j=1}^{N_t} \hat{\pi}_{ij}E_{ij}^{(n)} \\ &\text{subject to } \textstyle\sum_i \hat{\pi}_{ij} = \frac{1}{N_t}, \sum_j \hat{\pi}_{ij} = \frac{1}{N_s}. \quad (4)\end{aligned}$$

This problem can efficiently solved using entropy regularization and Sinkhorn-Knopp algorithm [37]. The model $\hat{f}_s^{(1)}$ is not obtained for the first iteration; hence, the cost $\mathcal{E}^0(\boldsymbol{x}^c_s; \boldsymbol{x}^c_t, \boldsymbol{x}^e_t) \equiv d(\boldsymbol{x}^c_{si}, \boldsymbol{x}^c_{tj})$ is used instead of $\mathcal{E}_\alpha$ only for the first iteration.

Then, the optimization problem with respect to f with a fixed nth OT plan $\hat{\pi}_n^*$ is solved to train model f, namely,

$$\hat{f}_s^{(n+1)} = \mathop{\arg\min}_{f\in\mathcal{F}} \sum_{i=1}^{N_s}\sum_{j=1}^{N_t} (\hat{\pi}_n^*)_{ij}\mathcal{L}(y_{si}, f(\boldsymbol{x}^c_{tj}, \boldsymbol{x}^e_{tj})). \quad (5)$$

In the above problem, there are cases in which different labels are transferred onto one sample. In these cases, we regard the transferred labels as soft labels that take values in $[-1, 1]$. Then, the transferred labels can be calculated by barycentric mapping using $\hat{\pi}_n^*$. When the distance between the common features is the squared distance, we obtain

$$\tilde{\boldsymbol{y}}_t = \mathrm{diag}(\boldsymbol{1}^\top\hat{\pi}_n^*)^{-1}(\hat{\pi}_n^*)^\top\boldsymbol{y}_s.$$

In particular, when the task is classification, the assigned labels can be seen as soft class labels; however, to train model f, it would be easier to use hard labels, which are estimated as

$$\bar{y}_{tj} = \begin{cases}\mathrm{sign}(\tilde{y}_{tj}) & \mathrm{sign}(\tilde{y}_{tj}) \neq 0 \\ \tau & \mathrm{sign}(\tilde{y}_{tj}) = 0\end{cases}, \quad (6)$$

where $P(\tau = +1) = P(\tau = -1) = 1/2$. Then, the training of the model Eq. (5) becomes the following simple training process in the target domain:

$$\hat{f}_s^{(n)} = \mathop{\arg\min}_{f\in\mathcal{F}} \sum_{j=1}^{N_t} \mathcal{L}(\bar{y}_{tj}, f(\boldsymbol{x}^c_{tj}, \boldsymbol{x}^e_{tj})). \quad (7)$$

The above algorithm using the hard labels is summarized as Algorithm 1.

IV. Theoretical Justification And Analysis of Proposed Algorithm

In this section, the proposed method is analyzed mainly from two perspectives. First, we give an interpretation of our proposed method. Briefly, the main optimization problem Eq. (2) of the proposed method is identical to the two-way OT between the source and target domains under an assumption that the conditional distributions of $\boldsymbol{x}^e$ given $\boldsymbol{x}^c$ and y in the source and target domains are identical.

Then, a learning bound of the model f on the target domain is derived. The conventional analyses of domain adaptation methods based on OT [25], [38] give learning bounds that mainly focuses on the Wasserstein distance between the source

Algorithm 1 Domain adaptation for extended variable space

```
Input: datasets $\mathcal{D}_S, \mathcal{D}_T$, model set $\mathcal{F}$
Output: Optimal transport plan $\hat{\pi}_N^*$, trained model $\hat{f}_s^{(N)}$
  $n \leftarrow 1$
  while $n <$ max iteration do
    if $n = 1$ then
      $\hat{\pi}_1^* \leftarrow \mathop{\arg\min}_{\pi \in \hat{\Pi}(\mathcal{D}_S, \mathcal{D}_T)} \sum_{i,j} \pi_{ij} \mathcal{E}^0(\boldsymbol{x}_{si}^c, \boldsymbol{x}_{tj}^c)$
    else
      $\hat{\pi}_n^* \leftarrow$ Optimization in Eq. (4).
    end if
    $\tilde{\boldsymbol{y}}_t \leftarrow \mathrm{diag}(\mathbf{1}^\top \hat{\pi}_n^*)^{-1} (\hat{\pi}_n^*)^\top \boldsymbol{y}_s$
    estimate hard labels $\bar{y}_{tj}$ $(j = 1, \dots N_t)$ by Eq. (6)
    $\hat{f}_s^{(n+1)} \leftarrow \arg\min_{f \in \mathcal{F}} \sum_{j=1}^{N_t} \mathcal{L}(\bar{y}_{tj}, f(\boldsymbol{x}_{tj}^c, \boldsymbol{x}_{tj}^e))$
    $n \leftarrow n + 1$
  end while
```

and target distributions. Although it is possible to extend this upper bound for our algorithm, they become loose when the Wasserstein distance between the source and target distributions becomes large even if it is possible to correctly transfer source labels to the target domain. Moreover, the upper bound does not consider how the model f is trained in the target domain. On the other hand, we give an upper bound that focuses on the training of the model f in this paper, and the target error is upper bounded by the Rademacher complexity and the Wasserstein distance between the estimated and true target distributions. That is, the upper bound becomes tight when the transferred source distribution is close to the true target distribution, and the model can accurately predict the transferred label. This interpretation gives an intuitive understanding of the required conditions for the successful domain adaptation.

A. Theoretical Justification of Proposed Algorithm

The main problem stated in Eq. (2) considers the transportation from the source domain to the target domain. Let us start with an ideal case that the common feature $\boldsymbol{x}^c$, the extra feature $\boldsymbol{x}^e$, and the label y are observed in both the source and target domains. Here, let $\boldsymbol{x}_s^e$ be the extra features in the source domain, which are not observed in practice. In this ideal case, the domain adaptation becomes a homogeneous domain adaptation problem, which is relatively easy to solve. The cost function for the transportation is defined as

$$\begin{aligned} &\mathcal{E}_\alpha^*(\boldsymbol{x}_s^c, \boldsymbol{x}_s^e, y_s; \boldsymbol{x}_t^c, \boldsymbol{x}_t^e, y_t) \\ &\quad = \alpha d((\boldsymbol{x}_s^c, \boldsymbol{x}_s^e), (\boldsymbol{x}_t^c, \boldsymbol{x}_t^e)) + \mathcal{L}(y_s, y_t). \end{aligned} \tag{8}$$

Since this cost function is symmetric, the transportation between the source and target domains is invertible. However, $\boldsymbol{x}_s^e$ and y_t are not observed in practice, making it necessary to estimate these values. Although the label y_t is substituted by its estimated value $\hat{y}_t = f(\boldsymbol{x}_t^c, \boldsymbol{x}_t^e)$, the model to estimate the extra feature $\boldsymbol{x}_s^e$ is not considered in the proposed Algorithm 1. Here, let us consider the estimation of $\boldsymbol{x}_s^e$. A straightforward method of estimating $\boldsymbol{x}_s^e$ is to transfer the information of $\boldsymbol{x}^e$ in the target domain to the source domain by OT. The cost function for this transportation is defined as

$$\mathcal{E}_\alpha^{ts}(\boldsymbol{x}_t^c, y_t; \boldsymbol{x}_s^c, y_s) = \alpha d(\boldsymbol{x}_t^c, \boldsymbol{x}_s^c) + \mathcal{L}(y_t, y_s). \tag{9}$$

Owing to the lack of $\boldsymbol{x}_s^e$, the extra feature $\boldsymbol{x}^e$ is only considered with the estimated target label $f(\boldsymbol{x}_s^c, \boldsymbol{x}_t^e)$ that is used to substitute the unobserved target label y_t. Here, an additional assumption is made so that the distribution of the extra features is estimated by OT using the above cost function, that is,

$$\mathcal{P}_T(\boldsymbol{x}^e | \boldsymbol{x}^c, y) = \mathcal{P}_S(\boldsymbol{x}^e | \mathcal{T}(\boldsymbol{x}^c, y)), \tag{10}$$

where $\mathcal{T}$ represents the OT of the common features $\boldsymbol{x}^c$ and the label y from the target distribution to the source distribution. This assumption means that the conditional distribution of the extra features $\boldsymbol{x}^e$ given $(\boldsymbol{x}^c, y)$ is identical before and after OT of the common features $\boldsymbol{x}^c$ and the label y. Under this assumption, the target extra features can be transferred to the source domain by OT.

After $\boldsymbol{x}_s^e$ is estimated using the above OT, it is possible to transfer the label information from the source domain to the target domain by OT using the cost function Eq. (8). However, under the assumption Eq. (10), there always exists a target sample that has the same extra features $\boldsymbol{x}^e$ as a source sample at the destination of OT. In other words, when we solve the OT problem for common features and labels, the transport cost of the extra features is always minimized to zero. Therefore, eventually, the estimation of the source extra feature is not required, and solving the one-way OT problem Eq. (2) is equivalent to solving the two-way OT problem.

B. Learning Bound of Trained Model on Target Domain

In this subsection, we show the learning bound of the model f on the target domain. The upper bound derived here is related to the upper bound derived in [25]. Their upper bound focuses on the transportation between the source distribution and the estimated target distribution that is solved for JDOT. However, their upper bound does not take into account the training of model f involved in the practical algorithm. On the other hand, the upper bound derived here focuses on training of the model f; hence, our upper bound becomes tighter with respect to the model. More specifically, the derived upper bound consists of the Rademacher complexity of the model set and the Wasserstein distance between the estimated target distribution and the true target distribution. We remark that although the Wasserstein distance between the estimated target distribution and the true target distribution is contained in the upper bound, the transportation between the source and estimated target distributions is not considered explicitly. Namely, instead of considering the training of the model, our analysis does not consider how to estimate the target distribution explicitly.

To begin with, we consider the probabilistic transfer Lipschitzness introduced in [25].

Definition 1 (Probabilistic Transfer Lipschitzness)**.** Let μ_s and μ_t be the source and target distributions, respectively, and define $\phi(\lambda) : \mathbb{R} \to [0,1]$. A labeling function $f : \Omega \to \mathbb{R}$ and a joint distribution $\pi(\mu_s, \mu_t)$ over the distributions μ_s and μ_t are ϕ-Lipschitz transferable if for all $\lambda > 0$,

$$\Pr_{(\boldsymbol{x}_1,\boldsymbol{x}_2)\sim\pi(\mu_s,\mu_t)}[|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| > \lambda d(\boldsymbol{x}_1, \boldsymbol{x}_2)] \le \phi(\lambda).$$

The definition of probabilistic transfer Lipschitzness implies that if two instances $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are sufficiently close, the probability that these instances have different labels is bounded by $\phi(\lambda)$, where λ is inversely proportional to the closeness of the instances.

To derive an upper bound, let $\mathcal{P}_{\hat{T}}$ be the distribution that estimates the true target distribution $\mathcal{P}_T$. Then, we define the expected loss for a model $f \in \mathcal{F}$ with respect to each distribution $\mathcal{P}_T$, $\mathcal{P}_{\hat{T}}$, as

$$\begin{aligned}\mathrm{err}_T(y, f) &= \mathbb{E}_{(x^c,x^e,y)\sim\mathcal{P}_T}\mathcal{L}(y, f(x^c, x^e)),\\ \mathrm{err}_{\hat{T}}(y, f) &= \mathbb{E}_{(x^c,x^e,y)\sim\mathcal{P}_{\hat{T}}}\mathcal{L}(y, f(x^c, x^e)).\end{aligned}$$

Also, let $\hat{\mathcal{P}}_{\hat{T}}$ be the empirical distribution that consists of samples $\{(\boldsymbol{x}_i^c, \boldsymbol{x}_i^e, y_i)\}_{i=1}^{N_t}$ that follow $\mathcal{P}_{\hat{T}}$. Then, the empirical loss for model f with respect to $\hat{\mathcal{P}}_{\hat{T}}$ is defined as

$$\widehat{\mathrm{err}}_{\hat{T}}(y, f) = \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(y_i, f(\boldsymbol{x}_i^c, \boldsymbol{x}_i^e)).$$

Let f_0 and $\hat{f}_s$ be the models that minimize the expected loss $\mathrm{err}_T(y, f)$ and the empirical loss $\widehat{\mathrm{err}}_{\hat{T}}(y, f)$:

$$f_0 = \inf_{f\in\mathcal{F}} \mathrm{err}_T(y, f), \qquad \hat{f}_s = \min_{f\in\mathcal{F}} \widehat{\mathrm{err}}_{\hat{T}}(y, f).$$

Now, we are ready to present our main result. Assume that the following condition holds.

- The space of the target features, Ω_t, is endowed with a positive definite kernel K, and let $\mathcal{H}$ be its associated reproducing kernel Hilbert space.
- The kernel K is bounded as $\sup_{\boldsymbol{x}\in\Omega_t} K(\boldsymbol{x}, \boldsymbol{x}) \le \Lambda^2$.
- The model set $\mathcal{F}$ is a ball of radius a in $\mathcal{H}$, namely, $\mathcal{F} = \{f \in \mathcal{H} \mid \|f\|_{\mathcal{H}} \le a\}$.
- The loss function $\mathcal{L}$
  - is symmetric, $\mathcal{L}(y_1, y_2) = \mathcal{L}(y_2, y_1)$,
  - satisfies the triangle inequality, $\mathcal{L}(y_1, y_2) + \mathcal{L}(y_2, y_3) \ge \mathcal{L}(y_1, y_3)$,
  - is Lipschitz continuous with constant k, $|\mathcal{L}(y_1, y_2) - \mathcal{L}(y_1, y_3)| \le k|y_2 - y_3|$, and
  - is bounded as $L_0 = \sup_{y\in\mathcal{C}} \mathcal{L}(0, y) < \infty$.
- The optimal model $f_0 \in \mathcal{F}_0$ is upper bounded as, for all $\boldsymbol{x}_1^c, \boldsymbol{x}_1^e, \boldsymbol{x}_2^c, \boldsymbol{x}_2^e$, $|f_0(\boldsymbol{x}_1^c, \boldsymbol{x}_1^e) - f_0(\boldsymbol{x}_2^c, \boldsymbol{x}_2^e)| \le M$.
- The optimal model f_0 and the OT plan π^* from $\mathcal{P}_{\hat{T}}$ to $\mathcal{P}_T$ satisfy the ϕ-probabilistic transfer Lipschitzness.

We remark that $\mathcal{F} \ne \mathcal{F}_0$ in general. Then, our main result is summarized as follows.

Theorem 1. Under the above assumptions, let $\hat{f}_s \in \mathcal{F}$ be the trained model that minimizes the empirical loss $\widehat{\mathrm{err}}_{\hat{T}}$ of the distribution $\mathcal{P}_{\hat{T}}$ that estimates the true target distribution. Then, for all $\lambda > 0$ with $\alpha = k\lambda$, for any $\delta \in (0, 1)$ with probability at least $1 - \delta$,

$$\begin{aligned}\mathrm{err}_T(y, \hat{f}_s) \le \widehat{\mathrm{err}}_{\hat{T}}(\hat{f}_s, y) + \frac{2ka\Lambda}{\sqrt{N_t}} + (L_0 + ka\Lambda)\sqrt{\frac{\ln(1/\delta)}{N_t}}\\ + W(\mathcal{P}_{\hat{T}}, \mathcal{P}_T) + 2\,\mathrm{err}_T(y, f_0) + kM\phi(\lambda),\end{aligned}$$

where $W(\mathcal{P}_{\hat{T}}, \mathcal{P}_T)$ is the Wasserstein distance between the estimated target distribution and the true target distribution and is defined as

$$\begin{aligned}W(\mathcal{P}_{\hat{T}}, \mathcal{P}_T) = \inf_{\pi\in\Pi(\mathcal{P}_{\hat{T}},\mathcal{P}_T)} \int_{(\Omega_t\times\mathcal{C})^2}\\ \mathcal{E}_\alpha^*(\boldsymbol{x}_1^c, \boldsymbol{x}_1^e, y_1; \boldsymbol{x}_2^c, \boldsymbol{x}_2^e, y_2) d\Pi(\boldsymbol{x}_1^c, \boldsymbol{x}_1^e, y_1; \boldsymbol{x}_2^c, \boldsymbol{x}_2^e, y_2).\end{aligned}$$

The sketch of the proof is as follows. First, we consider the difference of the expected loss and its upper bound by triangular inequality $\mathrm{err}_T(y, \hat{f}_s) - \mathrm{err}_T(y, f_0) \le \mathrm{err}_T(\hat{f}_s, f_0) \le \mathrm{err}_{\hat{T}}(y, \hat{f}_s) + \mathrm{err}_{\hat{T}}(y, f_0)$. The first term is upper bounded by the Rademacher complexity based on uniform law of large numbers [39]. Then, the second term is upper bounded by the similar upper bound as the bound shown in [25]. Here, we consider the Wasserstein distance between the estimated target distribution and the true target distribution, because calculation of the distance between the source and target distributions is not possible due to the difference of the dimensionality. The complete proof is omitted due to the limitation of space, and will be presented in a full paper, to be released later. When we assume $\mathcal{P}_{\hat{T}} = \mathcal{P}_T^f$, the upper bound corresponds to Algorithm 1. The above upper bound is divided into three parts. The first part, $\widehat{\mathrm{err}}_{\hat{T}}(\hat{f}_s, y) + \frac{2ka\Lambda}{\sqrt{N_t}} + (L_0 + ka\Lambda)\sqrt{\ln(1/\delta)/N_t}$, is an upper bound based on the Rademacher complexity and estimates $\mathrm{err}_{\hat{T}}(\hat{f}_s, y)$ from a finite number of samples that follow $\mathcal{P}_{\hat{T}}$. In addition, $\hat{f}_s$ is obtained by minimizing $\widehat{\mathrm{err}}_{\hat{T}}$; hence, these terms are minimized in terms of the model $f \in \mathcal{F}$. The second part, $W(\mathcal{P}_{\hat{T}}, \mathcal{P}_T)$, is the discrepancy between the estimated and true target distributions. This distance becomes small if the estimated target distribution is close to the true target distribution irrespective of the distance between the source and target distributions. Namely, this term focuses on the transferability of the source label information to the target domain and is more plausible as a term of an upper bound than the distance between the source and target distribution itself. The last part, $2\,\mathrm{err}_T(y, f_0) + kM\phi(\lambda)$, is determined by the predictability of the target distribution and the probabilistic transfer Lipschitzness of model f_0; hence, these terms are considered constants that depend on the problem. In conclusion, the upper bound becomes tight when the estimated and true target distributions are close, and the model can accurately predict the transferred label.

V. Numerical Experiments

In this section, we present experimental results of domain adaptation problems for the observation of extra features using both synthetic and real data. We used Python Optimal Transport (POT) [40] in the following experiments, and codes are publicly available at https://github.com/t-aritake/DAEVS.

A. Experiments with Synthetic Data

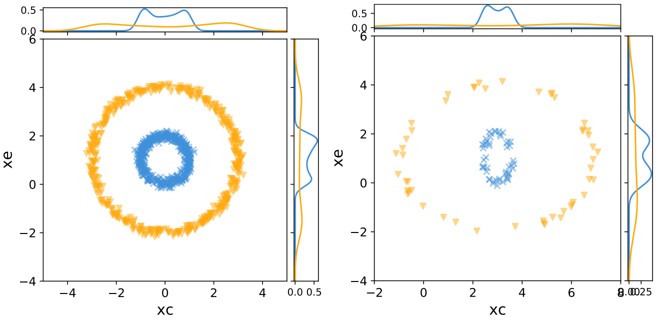
In this subsection, we show experimental results obtained with synthetic data. In this experiment, unsupervised domain adaptation for a binary classification problem is considered. We assume the circular dataset shown in Fig. 2 as the true source and target datasets. Here, both the common and extra features are one-dimensional for the purpose of visualization, and the extra features in the source domain and the labels in the target domain are not used for the classification. We set the number of samples to be $N_s = 1,000$ and $N_t = 100$ for the source and target domains, respectively. The dataset of each domain contains the same number of positive and negative samples.

We compared our proposed method with JDOT [25], which ignores the extra variable x^e, heterogeneous domain adaptation using canonical correlation analysis (CCA) [20], and domain specific feature transfer (DSFT) [21]. Also, JDOT in the ideal situation, where the extra variable is observed in both source and target domains, is used as a benchmark for the optimal performance. We assume that the class set $\mathcal{F}$ is the set of support vector machines (SVMs) with a Gaussian kernel for all methods. We use the training loss of SVMs as $\mathcal{L}$, which is a surrogate loss of 0-1 loss, and set the balancing parameter α of Eq. (3) to $\alpha = 0.1$, which was experimentally determined.

The classification accuracy for the proposed method, JDOT ignoring the extra variable, and the optimal benchmark, which is JDOT with extra variable, is shown in Table I. We generated 10 different random datasets, where each dataset is similar to the two-circle dataset in Fig. 2. Then, we calculated the average prediction accuracy and its variance of the transferred label (transfer) and the estimated target label using the trained model f (model) for our proposed method and JDOT-based method. We evaluated these values because it is possible to build an accurate model from partly incorrectly transferred labels, or conversely, there is possibility to build an inaccurate model from the correctly transferred labels. Also, Fig. 3 shows the decision boundary of a trained model in the target domain obtained by the proposed method. From Table I, we can see that our proposed method outperforms other methods. The target distributions have better separability due to the existence of extra features; therefore, it is difficult to embed the source and target distributions into the same dimensionality properly. Also, the extra distribution $\mathcal{P}_T(x^e|x^s)$ is multimodal; hence, it is difficult to estimate the source extra feature from the common feature by regression.

Also, we can see that our proposed method outperforms JDOT without an extra variable, and the model accuracy is higher than the transfer accuracy in both methods. Since the marginal distribution $\mathcal{P}(x^c)$ of the positive and negative class are highly overlapped, as can be seen in Fig. 2, it is difficult to build a model that accurately predicts the label only from the common feature. On the other hand, our proposed method accurately estimated the true target labels by OT. Furthermore, as shown in Fig. 3, even when some of the labels are not correctly transferred, the trained model is able to estimate the true target labels accurately.

Intuitively, our proposed method recovers the true target distribution $\mathcal{P}_T(x^c, x^e, y)$ from the source marginal distribution $\mathcal{P}_S(x^c, y)$ and the target marginal distribution $\mathcal{P}_T(x^c, x^e)$. In this problem, $\mathcal{P}_T(x^c, y)$ is estimated from $\mathcal{P}_S(x^c, y)$ by OT and $\mathcal{P}_T(x^c, x^e)$ is known. Still, it is not possible to recover $\mathcal{P}_T(x^c, x^e, y)$ since $\mathcal{P}_T(x^e|x^c, y)$ is not known. Then, the OT problem of our proposed method can be seen as a process to estimate $\mathcal{P}_T(x^e|x^c, y)$ through the transportation cost $\mathcal{L}(y_{si}, f(\boldsymbol{x}^c_{tj}, \boldsymbol{x}^e_{tj}))$ based on the model f. Note, however, that estimability of $\mathcal{P}_T(x^e|x^c, y)$ depends on the structure of the true target distribution, especially, complexity of the model and the (probabilistic) Lipschitzness of the true labeling function. A detailed analysis of the required conditions for the success of the estimation is left for our future work.



(a) Source dataset (b) Target dataset

Fig. 2: Example of the true dataset used in the experiment. Triangles show positive samples and crosses show negative samples. The center of the dataset is different in the source and target domains, and the shape of the circle is also slightly different. The x_e of the source samples and the labels of the target samples are not accessible in the experiment.

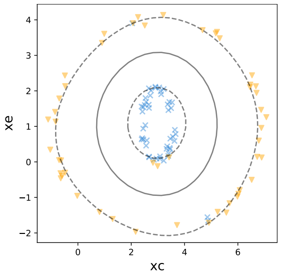


Fig. 3: Transferred labels in the target domain and the obtained decision boundary of the learned model.

TABLE I: Accuracy with synthetic data

	Proposed	JDOT no extra	CCA	DSFT	JDOT ideal
transfer	**92.6** (3.80)	84.6 (3.41)	-	-	96.3 (3.44)
model	**99.4** (1.28)	85.0 (2.10)	73.9 (10.3)	55.7 (2.34)	100 (0.00)

B. Experiments With Real Data

In this subsection, we show experimental results obtained with real data. We used the gas sensor array drift dataset used in [41]. The original data are 16-channel time series obtained by measuring one of six gases at different concentration levels using an array of 16 gas sensors. The dataset consists of 10 batches, where the samples in each batch are obtained for a different month and are affected by different levels of sensor drift; hence, each batch can be used as a dataset of different domains. We used the first four batches and considered a domain adaptation problem between these batches. Also, we consider the binary classification problem to classify only two types of gases, ethanol and ethylene, out of the six types of gases. We used the six transient features extracted from each sensor for classification. We selected eight out of 16 sensors, and transient features extracted from these sensors are used as common features, while the features extracted from the rest of the sensors are used as extra features. Here, the sensors used to extract common features are selected so that the extra features make the classification more accurate. Although, in practice, it is possible that the extra features do not contribute to the accuracy of the classification, here, we considered the reasonable scenario that informative features for the classification are observed as extra features.

Table II shows the prediction accuracy in the target domain for each domain adaptation problem. The row of domains $A \rightarrow B$ shows the experimental results where batch A and batch B are used as the source and target domains, respectively. The Baseline column shows the prediction accuracy on the test data without domain adaptation. Namely, the baseline model is learned using only common features given in the source domain. Similarly, the model accuracies of JDOT, CCA-based heterogeneous domain adaptation, DSFT, the proposed method, and the optimal benchmark are shown in the table. As we can see from the table, some domains do not require domain adaptation, and the baseline model outperforms JDOT and the proposed method. However, for other domains, the prediction accuracy is largely improved by considering domain adaptation by OT. In addition, the proposed method outperforms JDOT in most domains using the informative extra features. Also, in some cases, the proposed method outperforms optimal benchmark. This result suggests that, not all features may contribute to the accurate estimation of the target distribution by OT for the real data.

VI. Conclusion

In this paper, we considered the domain adaptation problem in which common features are observed in both the source and target domains, and extra features are observed only in the target domain. We proposed an unsupervised heterogeneous domain adaptation method based on JDOT to effectively utilize the extra features for better estimation of the target labels. We showed that our proposed method is equivalent to the two-way OT to transfer extra features and labels between the domains under the assumption that the conditional distribution of extra features given common features, and . Also, we derived a learning bound of the model in the target domain on the basis of the Rademacher complexity and the Wasserstein distance between the estimated and true target distributions. The experimental results demonstrate the ability to estimate a distribution close to the true target distribution by the proposed method, hence the better target model is obtained.

TABLE II: Accuracy for real data

domains	Baseline	JDOT	CCA	DSFT	Proposed	JDOT
		no extra				ideal
1→ 2	**83.33**	77.71	66.87	42.97	78.31	83.73
1→ 3	52.28	93.45	43.27	57.78	**96.02**	94.15
1→ 4	64.49	60.75	58.88	**94.39**	87.85	85.98
2→ 1	52.13	79.26	56.91	62.77	**84.04**	85.64
2→ 3	56.84	89.36	25.03	84.56	**89.47**	90.99
2→ 4	63.55	69.16	51.40	41.12	**71.96**	74.77
3→ 1	51.06	92.02	92.55	66.49	**94.15**	95.74
3→ 2	68.67	81.92	67.67	86.94	**88.76**	88.55
3→ 4	94.39	77.57	**96.26**	40.19	81.31	80.37
4→ 1	50.00	52.66	48.93	51.60	**53.72**	80.85
4→ 2	42.97	71.08	32.93	32.93	**74.30**	73.89
4→ 3	**92.98**	82.81	57.31	42.69	82.57	82.57

The accurate estimation of the true target distribution is not always possible, and the conditions for the success of the estimation by the proposed method are not yet fully understood. The analysis of such conditions is important future work. Furthermore, the case where some of the features in the source domain become unobservable (because of, e.g., mechanical breakdown of sensors) should also be discussed as a future extension of the proposed method.

Acknowledgement

Part of this work is supported by JST CREST JPMJCR1761, JPMJCR2015, Mirai Project JPMJMI21G2, and JSPS JP20K06922.

References

[1] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine Learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[2] I. Redko, E. Morvant, A. Habrard, M. Sebban, and Y. Bennani, *Advances in domain adaptation theory*. ISTE Press, Elsevier, 2019.

[3] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *Journal of Statistical Planning and Inference*, vol. 90, no. 2, pp. 227–244, Oct. 2000.

[4] M. Sugiyama, S. Nakajima, H. Kashima, P. v. Bünau, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," in *Proceedings of the 20th International Conference on Neural Information Processing Systems*, 2007, pp. 1433–1440.

[5] G. Widmer and M. Kubat, "Learning in the presence of concept drift and hidden contexts," *Mach. Learn.*, vol. 23, no. 1, pp. 69–101, Apr. 1996.

[6] G. I. Webb and K. M. Ting, "On the Application of ROC Analysis to Predict Classification Performance Under Varying Class Distributions," *Machine Learning*, vol. 58, no. 1, pp. 25–32, 2005.

[7] R. Alaiz-Rodríguez and N. Japkowicz, "Assessing the impact of changing environments on classifier performance," in *Advances in Artificial Intelligence*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 13–24.

[8] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, jul 2017, pp. 2962–2971.

[9] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, S. A. McIlraith and K. Q. Weinberger, Eds. AAAI Press, 2018, pp. 4058–4065.

[10] F. Zhou, C. Shui, S. Yang, B. Huang, B. Wang, and B. Chaib-draa, "Discriminative active learning for domain adaptation," *Knowledge-Based Systems*, vol. 222, p. 106986, 2021.

[11] H. Zhao, R. T. D. Combes, K. Zhang, and G. Gordon, "On learning invariant representations for domain adaptation," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 7523–7532.

[12] Y. Zhang, F. Liu, Z. Fang, B. Yuan, G. Zhang, and J. Lu, "Learning from a complementary-label source domain: Theory and algorithms," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2021.

[13] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera, "A unifying view on dataset shift in classification," *Pattern Recognition*, vol. 45, no. 1, pp. 521–530, 2012.

[14] W. M. Kouw and M. Loog, "A review of domain adaptation without target labels," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 3, p. 766—785, March 2021.

[15] X. Shi, Q. Liu, W. Fan, and P. S. Yu, "Transfer across Completely Different Feature Spaces via Spectral Embedding," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 4, pp. 906–918, 2013.

[16] C. Wang and S. Mahadevan, "Heterogeneous Domain Adaptation Using Manifold Alignment," in *IJCAI'11: Proceedings of the Twenty-Second international joint conference on Artificial Intelligence*. AAAI Press, 2011, pp. 1541–1546.

[17] W. Li, S. Member, L. Duan, D. Xu, S. Member, and I. W. Tsang, "Learning with Augmented Features for Supervised and Semi-Supervised Heterogeneous Domain Adaptation," *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 36, no. 6, 2014.

[18] M. Xiao and Y. Guo, "Feature Space Independent Semi-Supervised Domain Adaptation via Kernel Matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 1, pp. 54–66, 2015.

[19] Z. Fang, J. Lu, F. Liu, and G. Zhang, "Semi-supervised heterogeneous domain adaptation: Theory and algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2022.

[20] Y.-R. Yeh, C.-H. Huang, and Y.-C. F. Wang, "Heterogeneous domain adaptation and classification by exploiting the correlation subspace," *IEEE Transactions on Image Processing*, vol. 23, no. 5, pp. 2009–2018, 2014.

[21] P. Wei, Y. Ke, and C. K. Goh, "A General Domain Specific Feature Transfer Framework for Hybrid Domain Adaptation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 8, pp. 1440–1451, 2019.

[22] J. Zhou, S. Pan, I. Tsang, and Y. Yan, "Hybrid Heterogeneous Transfer Learning through Deep Learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 28, no. 1, 2014.

[23] A. G. Prabono, B. N. Yahya, and S.-L. Lee, "Hybrid domain adaptation for sensor-based human activity recognition in a heterogeneous setup with feature commonalities," *Pattern Analysis and Applications*, vol. 24, no. 4, pp. 1501–1511, 2021.

[24] T. Kerdoncuff, R. Emonet, and M. Sebban, "Metric learning in optimal transport for domain adaptation," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 7 2020, pp. 2162–2168, main track.

[25] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, "Joint distribution optimal transportation for domain adaptation," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, vol. 30, 2017, pp. 3733–3742.

[26] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy, "Optimal transport for domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 9, pp. 1853–1865, 2017.

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

[28] G. Peyré, M. Cuturi, and J. Solomon, "Gromov-wasserstein averaging of kernel and distance matrices," in *Proceedings of The 33rd International Conference on Machine Learning*, vol. 48, New York, USA, 20–22 Jun 2016, pp. 2664–2672.

[29] V. Titouan, I. Redko, R. Flamary, and N. Courty, "CO-Optimal Transport," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., 2020, pp. 17 559–17 570.

[30] D.-H. Lee *et al.*, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on challenges in representation learning, ICML*, vol. 3, no. 2, 2013, p. 896.

[31] B. B. Damodaran, B. Kellenberger, R. Flamary, D. Tuia, and N. Courty, "DeepJDOT: Deep joint distribution optimal transport for unsupervised domain adaptation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[32] Y. Zou, Z. Yu, B. V. Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[33] I. Shin, S. Woo, F. Pan, and I. S. Kweon, "Two-phase pseudo label densification for self-training based domain adaptation," in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIII*, ser. Lecture Notes in Computer Science, vol. 12358. Springer, 2020, pp. 532–548.

[34] S. Xie, Z. Zheng, L. Chen, and C. Chen, "Learning semantic representations for unsupervised domain adaptation," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, 2018, pp. 5419–5428.

[35] W. Zhang, W. Ouyang, W. Li, and D. Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018, pp. 3801–3809.

[36] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 2019, pp. 4893–4902.

[37] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in Neural Information Processing Systems*, pp. 1–9, 2013.

[38] I. Redko, A. Habrard, and M. Sebban, "Theoretical Analysis of Domain Adaptation with Optimal Transport," in *ECML PKDD 2017*, Skopje, Macedonia, Sep. 2017.

[39] V. N. Vapnik, *Statistical Learning Theory*, ser. A Wiley-Interscience publication. Wiley, 1998.

[40] R. Flamary, N. Courty, A. Gramfort, M. Z. Alaya, A. Boisbunon, S. Chambon, L. Chapel, A. Corenflos, K. Fatras, N. Fournier, L. Gautheron, N. T. Gayraud, H. Janati, A. Rakotomamonjy, I. Redko, A. Rolet, A. Schutz, V. Seguy, D. J. Sutherland, R. Tavenard, A. Tong, and T. Vayer, "POT: Python optimal transport," *Journal of Machine Learning Research*, vol. 22, no. 78, pp. 1–8, 2021.

[41] A. Vergara, S. Vembu, T. Ayhan, M. A. Ryan, M. L. Homer, and R. Huerta, "Chemical gas sensor drift compensation using classifier ensembles," *Sensors and Actuators B: Chemical*, vol. 166-167, pp. 320–329, 2012.