Transferability-Guided Cross-Domain Cross-Task Transfer Learning

Yang Tan, *Student Member, IEEE*, Enming Zhang, Yang Li, *Member, IEEE*, Shao-Lun Huang, *Member, IEEE*, and Xiao-Ping Zhang, *Fellow, IEEE*

***Abstract*— We propose two novel transferability metrics fast optimal transport-based conditional entropy (F-OTCE) and joint correspondence OTCE (JC-OTCE) to evaluate how much the source model (task) can benefit the learning of the target task and to learn more generalizable representations for cross-domain cross-task transfer learning. Unlike the original OTCE metric that requires evaluating the empirical transferability on auxiliary tasks, our metrics are auxiliary-free such that they can be computed much more efficiently. Specifically, F-OTCE estimates transferability by first solving an optimal transport (OT) problem between source and target distributions and then uses the optimal coupling to compute the negative conditional entropy (NCE) between the source and target labels. It can also serve as an objective function to enhance downstream transfer learning tasks including model finetuning and domain generalization (DG). Meanwhile, JC-OTCE improves the transferability accuracy of F-OTCE by including label distances in the OT problem, though it incurs additional computation costs. Extensive experiments demonstrate that F-OTCE and JC-OTCE outperform state-of-the-art auxiliary-free metrics by 21.1% and 25.8%, respectively, in correlation coefficient with the ground-truth transfer accuracy. By eliminating the training cost of auxiliary tasks, the two metrics reduce the total computation time of the previous method from 43 min to 9.32 and 10.78 s, respectively, for a pair of tasks. When applied in the model finetuning and DG tasks, F-OTCE shows significant improvements in the transfer accuracy in few-shot classification experiments, with up to 4.41% and 2.34% accuracy gains, respectively.**

***Index Terms*— Cross-domain, cross-task, few-shot learning, source selection, task relatedness, transfer learning, transferability estimation.**

I. Introduction

TRANSFER learning is an effective learning paradigm to enhance the performance on target tasks via leveraging prior knowledge from the related source tasks (or source models), especially when there are only a few labeled data for supervision [1], [2], [3], [4]. However, the success of transfer learning is not always guaranteed. If the source and target tasks are unrelated or if the transferred representation does not carry sufficient information about the target task, transfer learning will not obtain a notable gain on the target task performance and may even experience negative transfer, i.e., the performance becomes worse than that of training from scratch on the target task [5]. Therefore, understanding when and what to transfer between tasks is crucial to the success of transfer learning.

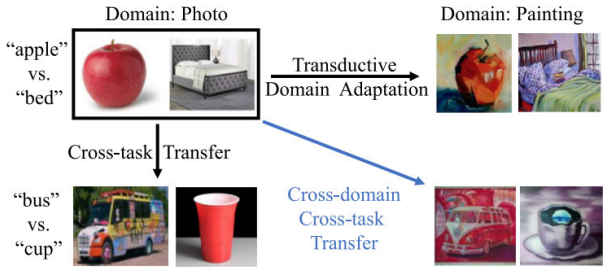

Fig. 1. Illustration of three different transfer learning settings, i.e., transductive domain adaptation [6], cross-task transfer [7], and cross-domain cross-task transfer we investigating.

The "when to transfer" problem was traditionally studied theoretically through the derivation of generalization bounds of transfer learning across tasks [8], [9] and across domains (also known as the domain adaptation problem) [10], [11], [12], [13], [14], [15]. Such studies bound the target task generalization error by a function that depends on certain divergence between the source and target domain or the complexity of the hypothesis class for the source and target tasks. In practice, however, these bounds are difficult to compute from data, and they tend to rely on strict assumptions that cannot be verified. In recent years, the notion of task transferability was proposed to address the "when to transfer" problem in the context of deep transfer learning [7], [16], [17], [18], [19], [20], [21], [22], [23]. The transferability problem aims to quantitatively evaluate how much the source task or source model could benefit the learning of the target task. It can be used to directly select the most "transferable" source model from a model zoo for a target task, rather than exhaustively trying each source model on the target data. In addition, transferability can help prioritize different tasks for joint training [16] and multisource feature fusion [22].

As empirical transferability studies [16], [17], [18], [19], [24] incur heavy computational burdens in retraining the transfer learning model on the target training data, a new

Manuscript received 30 June 2022; revised 31 March 2023 and 8 November 2023; accepted 15 January 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62371270, in part by the Tsinghua Shenzhen International Graduate School (SIGS) Scientific Research Start-Up Fund under Grant QD2021012C, and in part by the Shenzhen Key Laboratory of Ubiquitous Data Enabling under Grant ZDSYS20220527171406015. *(Corresponding author: Yang Li.)*

The authors are with the Shenzhen Key Laboratory of Ubiquitous Data Enabling, Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China (e-mail: yangli@sz.tsinghua.edu.cn).

Digital Object Identifier 10.1109/TNNLS.2024.3358094

trend of transferability research aims to efficiently estimate the transfer performance a priori with little or no training of the transfer model. Several efficient transferability metrics are proposed, including negative conditional entropy (NCE) [20], H-score [7], log expected empirical prediction (LEEP) [21], and LogME [23]. Despite being evidently more efficient in computing from practical data than empirical methods, they are also prone to strict data assumptions [7], [20] and insufficient performance [21], [23], while task complexities are similar. Moreover, the aforementioned metrics are solely used for determining when to transfer between a pair of source and target tasks, but they do not contribute to solving the "what to transfer" problem, i.e., how to obtain more generalizable feature representations across domains and tasks.

Recently, a novel transferability metric optimal transport-based CE (OTCE) [22] is proposed to effectively estimate the transferability under the challenging cross-domain cross-task transfer setting, as shown in Fig. 1. Unlike the transferability metrics mentioned earlier, OTCE adopts a more analytical disentanglement approach. It explicitly assesses the domain difference (measured by the Wasserstein distance) and the task difference (determined by the CE) between tasks, and then integrates them via a linear model to quantify transferability. This technically sound design yields substantial accuracy improvement over the aforementioned metrics. Nevertheless, a major limitation of OTCE is its dependency on auxiliary tasks with known transfer performance to determine the intrinsic parameters of the linear model. On one hand, the availability of sufficient labeled data for creating auxiliary tasks is not always guaranteed. On the other hand, assessing the transfer performance of such auxiliary tasks necessitates retraining the source model, incurring additional computational costs. As a result, the reliance on auxiliary tasks makes OTCE relatively inefficient and less applicable in general practical scenarios.

In this article, we aim to broaden the applicability of the OTCE framework and investigate the potential uses of transferability in downstream transfer learning tasks. We propose two auxiliary-free transferability metrics, namely, fast OTCE (F-OTCE) and joint correspondence OTCE (JC-OTCE), which eliminate the need for auxiliary tasks and substantially enhance the efficiency without compromising accuracy. For classification problems, the F-OTCE metric solves the optimal transport (OT) problem [25], [26] to estimate a probabilistic coupling between the unpaired samples from the source and target datasets. Then, the optimal coupling enables us to derive the negative CE between the source and target task labels for representing transferability, which measures the label uncertainty of a target sample given the labels of corresponding source samples. While the F-OTCE metric does not explicitly evaluate the domain difference, the estimated probabilistic coupling between the source and target data implicitly captures the domain difference to some extent in this unified framework.

Then, we propose the JC-OTCE metric to further improve the accuracy of the F-OTCE metric in diverse transfer configurations. Our motivation is that F-OTCE only considers the joint probability distribution of input samples when determining data correspondences between the source and target domains. However, this approach is limited because the definition (label annotations) of the source task can also affect model generalization. To address this limitation, we incorporate label distance into the ground cost of the OT problem, allowing for the computation of correspondences in both sample and label spaces. By including additional label information, JC-OTCE produces improved data correspondences that partially compensate for the lack of explicit domain difference consideration. JC-OTCE achieves comparable transferability accuracy to the original OTCE metric but requires additional computation compared with F-OTCE, which remains preferable for efficiency purposes.

Moreover, we investigate the application of our transferability metric in two downstream transfer learning tasks including *model finetuning* and *DG*, offering a solution to the "what to transfer" problem. Specifically, to enhance the model finetuning performance, we propose an OTCE-based finetune algorithm that optimizes the pretrained source model to learn more transferable feature representation via maximizing the F-OTCE score between the source and target tasks. The optimized model is then finetuned on target training data using the classification loss function.

We also demonstrate that incorporating the F-OTCE metric into a novel DG method universal representation learning (URL) [27] can further improve its generalizability on unseen domains. Our motivation is to view distilling knowledge from domain-specific models to the universal model as maximizing the transferability between them. Therefore, we replace the knowledge distillation function in URL with our F-OTCE score, resulting in significant accuracy improvements in few-shot classification tasks on unseen domains.

This work is an extension of our previous conference paper [22], and the additional contributions are summarized as follows.

1) *Expanding the Applicability of OTCE Framework*: Our proposed F-OTCE and JC-OTCE metrics eliminate the need for auxiliary tasks and achieve comparable transferability accuracy to OTCE. They also outperform previous auxiliary-free transferability metrics in terms of accuracy while maintaining comparable efficiency.
2) *Investigating the Potential Uses of Transferability:* We illustrate the effectiveness of using F-OTCE as an optimization objective in improving the performance of downstream tasks, such as model finetuning and DG. We consider F-OTCE to be a general tool that can be easily integrated into various algorithms for transfer learning and other related applications.

In our experiments using several multidomain classification datasets, we show that our proposed two metrics significantly outperform existing auxiliary-free metrics with 25.8% correlation gain on average, while cutting more than 99% of the computation time in the original OTCE. We also show that, when served as a loss function, F-OTCE leads to notable classification accuracy gains on the model finetuning and DG tasks, with up to 4.41% and 2.34%. The rest of this article is

organized as follows. Section II introduces the formulation of transferability. Section III provides a preliminary analysis of OTCE. Section IV presents our two auxiliary-free transferability metrics. Section V illustrates our proposed transferability-guided model finetuning and DG algorithms. Section VI provides all the experimental results and analyses. Finally, we draw the conclusion in Section VII.

II. Transferability Formulation

Here, we introduce the formal definition of transferability. Suppose we have source data $D_s = \{(x_s^i, y_s^i)\}_{i=1}^m \sim P_s(x, y)$ and target data $D_t = \{(x_t^i, y_t^i)\}_{i=1}^n \sim P_t(x, y)$, where x represents the input instance and y represents the label. We have x_s^i and x_t^i from the input space $\mathcal{X}$, and y_s^i from the source label space $\mathcal{Y}_s$, and y_t^i from the target label space $\mathcal{Y}_t$. Meanwhile, $P(x_s) \neq P(x_t)$ and $\mathcal{Y}_s \neq \mathcal{Y}_t$ indicate different domains and tasks, respectively. In addition, we are given a source model (θ_s, h_s) pretrained on source data D_s, in which $\theta_s : \mathcal{X} \to \mathbb{R}^d$ represents a feature extractor producing d-dimensional features and $h_s : \mathbb{R}^d \to \mathcal{P}(\mathcal{Y}_s)$ is the head classifier predicting the final probability distribution of labels, where $\mathcal{P}(\mathcal{Y}_s)$ is the space of all probability distributions over $\mathcal{Y}_s$. Note that the notation θ and h can also represent model parameters.

In this article, we mainly investigate the transferability estimation problem with two representative transfer paradigms for neural networks [5], i.e., *Retrain head* and *Finetune*. The *Retrain head* method keeps the parameters of the source feature extractor θ_s frozen and retrains a new head classifier h_t. However, the *Finetune* method updates the source feature extractor and the head classifier simultaneously to obtain new (θ_t, h_t). Compared with *Retrain head*, *Finetune* trade-offs transfer efficiency for better transfer accuracy, and it requires more target data to avoid overfitting [22].

To obtain the empirical transferability, we need to retrain the source model via *Retrain head* or *Finetune* on target data and then evaluate the expected log-likelihood on its testing set. Formally, the empirical transferability is defined as follows.

Definition 1: The empirical transferability from the source task S to the target task T is measured by the expected log-likelihood of the retrained (θ_s, h_t) or (θ_t, h_t) on the testing set of target task

$$\mathrm{Trf}(S \to T) = \begin{cases} \mathbb{E}\big[\log\ P(y_t|x_t; \theta_s, h_t)\big] & \text{(Retrain head)} \\ \mathbb{E}\big[\log\ P(y_t|x_t; \theta_t, h_t)\big] & \text{(Finetune)} \end{cases} \tag{1}$$

which indicates how good the transfer performance is on the target task. In practice, we usually take the testing accuracy as an approximation of the log-likelihood [20], [22].

Although empirical transferability can be the golden standard of describing how easy it is to transfer the knowledge learned from a source task to a target task, it is computationally expensive to obtain. The efficient transferability metric is a function of the source and target data that approximates the empirical transferability, i.e., the ground truth of the transfer performance on target tasks. It is, therefore, imperative to find efficient transferability metrics that can accurately estimate empirical transferability.

III. Preliminary Analysis of OTCE

OTCE is an analytical transferability metric proposed for the cross-domain cross-task transfer learning setting. As illustrated in Fig. 2 (upper part), OTCE quantifies transferability as a linear model of the domain difference W_D (measured by Wasserstein distance) and task difference W_T (determined by CE), which is denoted as follows:

$$\mathrm{OTCE} = \lambda_1 W_D + \lambda_2 W_T + b. \tag{2}$$

However, a major limitation of OTCE is its dependency on auxiliary tasks with known transfer accuracy to determine the intrinsic parameters of the linear model. In practice, we are not always able to access sufficient labeled data from the target domain for constructing auxiliary tasks. Meanwhile, obtaining the transfer performance of auxiliary tasks needs retraining the source model, which incurs additional computational costs. As a result, the reliance on auxiliary tasks makes OTCE relatively inefficient and less applicable in general scenarios.

The statistic of the learned parameters λ_1, λ_2, and b (as shown in Fig. 3) reveals that $(|\lambda_2|)/(|\lambda_1|)$ among different transfer configurations varied irregularly, suggesting that the importance of domain difference and task difference varies for different cross-domain transfer learning settings. It is, therefore, incapable of using the predefined coefficients for computing OTCE scores. In addition, we note that the task difference W_T plays a more important role $((|\lambda_2|)/(|\lambda_1|) > 1)$ in evaluating transferability. Therefore, our proposed auxiliary-free transferability metrics mainly utilize the task difference for describing transferability.

IV. Auxiliary-Free Transferability Metrics

Our proposed auxiliary-free transferability metrics F-OTCE and JC-OTCE can be viewed as the efficient versions of the auxiliary-based OTCE metric, which only consider the negative CE to describe transferability, as depicted in Fig. 2. Although we do not explicitly evaluate the domain difference, the estimated probabilistic coupling between the source and target data implicitly captures the domain difference to some extent in this unified framework.

Specifically, F-OTCE achieves higher efficiency, while JC-OTCE performs better in terms of accuracy across diverse scenarios. The main difference between the two metrics is that the ground cost of JC-OTCE considers both sample distance and label distance when calculating the optimal coupling between the source and target data, which approximates computing ground cost in the joint space $\mathcal{X} \times \mathcal{Y}$, resulting in more precise data correspondences.

A. F-OTCE Metric

Formally, we first use the source feature extractor θ_s to embed the source and target input instances as latent features, denoted by $\hat{x}_s^i = \theta_s(x_s^i)$ and $\hat{x}_t^i = \theta_s(x_t^i)$, respectively. Then, the computational process contains two steps as described as follows.

Step 1: Compute Optimal Coupling: First, for the F-OTCE metric, we define the ground cost between samples as follows:

$$c_1(\hat{x}_s^i, \hat{x}_t^j) \triangleq \|\hat{x}_s^i - \hat{x}_t^j\|_2^2 \tag{3}$$

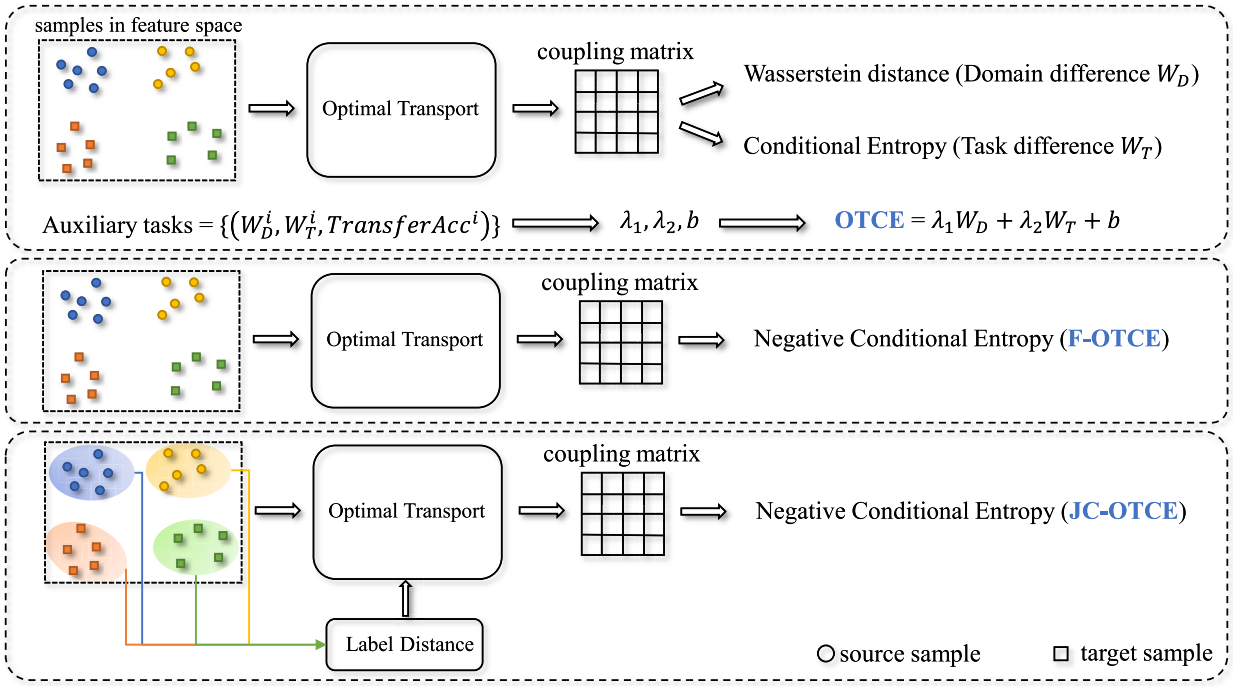

Fig. 2. Illustration of the auxiliary-based OTCE metric [22] (top), our proposed F-OTCE (middle), and JC-OTCE (bottom) metrics which do not require auxiliary tasks with known transfer accuracy to learn the weighting coefficients. For OTCE (top), W_D and W_T represent the domain difference and task difference between two tasks, respectively. To estimate the coefficients λ_1, λ_2, and b of the linear model, we need to sample at least three auxiliary tasks from the target dataset and calculate W_D^i, W_T^i, and transfer accuracy $TransferAcc^i$ between the source task and each auxiliary task as training data.

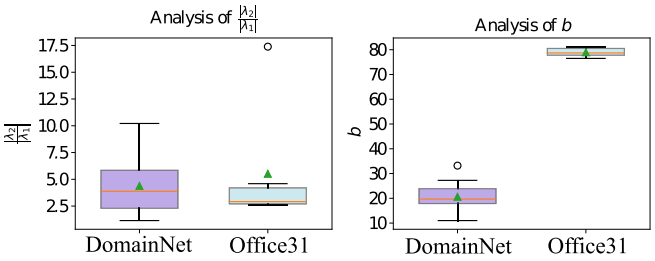

Fig. 3. Statistic of the learned weighting coefficients λ_1 and λ_2 and the bias term b of OTCE under diverse transfer configurations.

so the OT problem with the entropic regularization [28] can be defined as follows:

$$OT(X_s, X_t) \triangleq \min_{\pi \in \mathcal{P}(X_s, X_t)} \sum_{i,j=1}^{m,n} c_1(\hat{x}_s^i, \hat{x}_t^j)\pi_{ij} - \lambda H(\pi) \quad (4)$$

where π is the coupling matrix of size $m \times n$, and $H(\pi) = -\sum_{i=1}^{m}\sum_{j=1}^{n} \pi_{ij} \log \pi_{ij}$ is the entropic regularizer with $\lambda = 0.1$. The OT problem above can be solved efficiently by the Sinkhorn algorithm [28] to produce an optimal coupling matrix π^*.

From a probabilistic point of view, the coupling matrix π^* is a nonparametric estimation of the joint probability distribution of the source and target latent features $P(X_s, X_t)$. We model the relationship between the source and the target data according to the following simple Markov random field: $Y_s - X_s - X_t - Y_t$, where label random variables Y_s and Y_t are only dependent on X_s and X_t, respectively, i.e., $P(Y_s, Y_t|X_s, X_t) = P(Y_s|X_s)P(Y_t|X_t)$. Furthermore, we can derive the empirical joint probability distribution of the source and target labels

$$P(Y_s, Y_t) = \mathbb{E}_{X_s, X_t}[P(Y_s|X_s)P(Y_t|X_t)]. \quad (5)$$

This joint probability distribution can reveal the transfer performance since the goodness of class-to-class matching intuitively reveals the hardness of transfer.

Step 2: Compute Negative Conditional Entropy: We are inspired by Tran et al. [20] who use CE $H(Y_t|Y_s)$ to describe class-to-class matching quality over the same input instances. They have shown that the empirical transferability is lower bounded by the negative CE

$$\widetilde{\text{Trf}}(S \to T) \geq l_S(\theta_s, h_s) - H(Y_t|Y_s) \quad (6)$$

where the training log-likelihood $\widetilde{\text{Trf}}(S \to T) = l_T(\theta_s, h_t) = (1/n)\sum_{i=1}^{n} \log P(y_t^i|x_t^i; \theta_s, h_t)$ is an approximation of the empirical transferability when the retrained model is not overfitted. And, $l_S(\theta_s, h_s)$ is a constant, so the empirical transferability can be attributed to the CE.

We consider it as a reasonable metric to evaluate the transferability under the cross-domain cross-task transfer setting once we learn the soft correspondence π^* between source and target features via OT. We can also compute the empirical joint probability distribution of the source and target labels, and the marginal probability distribution of the source label, denoted as follows:

$$\hat{P}(y_s, y_t) = \sum_{i,j: y_s^i = y_s, y_t^j = y_t} \pi_{ij}^* \quad (7)$$

$$\hat{P}(y_s) = \sum_{y_t \in \mathcal{Y}_t} \hat{P}(y_s, y_t). \quad (8)$$

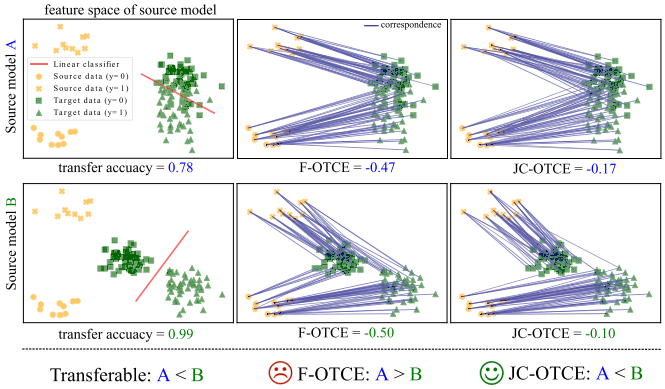

Fig. 4. Toy example shows that the F-OTCE metric fails to distinguish the more transferable source model, while the JC-OTCE predicts correctly by involving the label distance in computing the correspondences.

Then, we can compute the negative CE as the F-OTCE score

$$\text{F-OTCE} = -H_{\pi^*}(Y_t|Y_s) = \sum_{y_t\in\mathcal{Y}_t}\sum_{y_s\in\mathcal{Y}_s}\hat{P}(y_s,y_t)\log\frac{\hat{P}(y_s,y_t)}{\hat{P}(y_s)}. \quad (9)$$

Compared with the auxiliary-based OTCE, we directly use the negative CE to characterize transferability, which avoids the cumbersome parameter fitting process on auxiliary tasks, resulting in a drastic efficiency improvement.

B. JC-OTCE Metric

F-OTCE is an efficient transferability metric in practical scenarios, but its accuracy can be further improved. Take a toy example as shown in Fig. 4 for illustration, where the F-OTCE metric fails to distinguish the more transferable source model. This observation suggests that computing data correspondences solely based on sample distance (in space $\mathcal{X}$) may not always accurately capture the class-to-class matching quality (or the label uncertainty of the target task) as expected. Therefore, to further improve the accuracy of F-OTCE, we propose the JC-OTCE metric which involves the additional label distance in computing the joint correspondences between data in the joint space $\mathcal{X}\times\mathcal{Y}$.

Formally, we first define the data instances of the source and target tasks as $z_s = (\hat{x}_s, y_s)$ and $z_t = (\hat{x}_t, y_t)$, respectively, where $z_s \in \mathcal{Z}_s = \mathcal{X}\times\mathcal{Y}_s$ and $z_t \in \mathcal{Z}_t = \mathcal{X}\times\mathcal{Y}_t$. And, we define $\alpha_y \triangleq P(X|Y=y)$, which can be estimated from a collection of finite samples with label y. Inspired by recent work [29], we compute the label distance as the Wasserstein distance $\text{Wass}(\alpha_{y_s}, \alpha_{y_t})$. Then, the ground cost for JC-OTCE can be defined as follows:

$$c_2(z_s^i, z_t^j) \triangleq \gamma\|\hat{x}_s^i - \hat{x}_t^j\|_2^2 + (1-\gamma)\text{Wass}(\alpha_{y_s^i}, \alpha_{y_t^j}) \quad (10)$$

where $\gamma \in [0,1]$ is a weighting coefficient to combine the sample distance and the label distance, and here, we let $\gamma = 0.5$. More discussion about γ is described in Section VI-C. Similarly, the OT problem for Z_s and Z_t is defined as follows:

$$OT(Z_s, Z_t) \triangleq \min_{\pi\in\mathcal{P}(Z_s,Z_t)}\sum_{i,j=1}^{m,n}c_2(z_s^i, z_t^j)\pi_{ij} - \lambda H(\pi). \quad (11)$$

TABLE I
DIFFERENCES BETWEEN MODEL FINETUNING AND DG

	Model finetuning	Domain generalization
Source data	Single	Multiple
Target task	Known	Unknown
Goal	Achieving higher accuracy on the target task	Learning generalizable feature representations

By solving this OT problem, we also obtain the optimal coupling matrix π^*. Then, following the **Step 2** described in Section IV-A [see (7) and (8)], the JC-OTCE score is computed as the negative CE as well

$$\text{JC-OTCE} = -H_{\pi^*}(Y_t|Y_s) = \sum_{y_t\in\mathcal{Y}_t}\sum_{y_s\in\mathcal{Y}_s}\hat{P}(y_s,y_t)\log\frac{\hat{P}(y_s,y_t)}{\hat{P}(y_s)}. \quad (12)$$

V. TRANSFERABILITY-GUIDED TRANSFER LEARNING

In this section, we present two examples of utilizing our transferability metric to boost the performance of downstream transfer learning tasks, including model finetuning and DG. The differences between these two transfer learning tasks are described in Table I.

To facilitate the training process, we adopt the F-OTCE metric as the optimization objective since using the JC-OTCE metric needs solving multiple OT problems to compute pairwise label distances, which incurs significant computational costs. In addition, due to graphics processing unit (GPU) memory constraints, we typically perform mini-batch training which only loads a subset of the dataset in the current training iteration, while computing label distance requires loading the entire dataset.

A. OTCE-Based Model Finetuning

The vanilla finetune algorithm follows the "pretraining + finetuning" pipeline that is commonly used in transfer learning. However, this scheme does not consider the relatedness between the source and target tasks. To address this issue, our proposed OTCE-based finetune algorithm introduces an intermediate step into the conventional pipeline, i.e., maximize the transferability of transferring from the source task to the target task, resulting in a "pretraining + adaptation (maximizing transferability) + finetuning" framework. The moderate optimization during the adaptation step utilizes the task relationship characterized by our F-OTCE score to enable the source feature representation to become more transferable to the target task. This facilitates easier learning of the head classifier during the finetuning step and ultimately leads to higher transfer accuracy.

Suppose we have obtained the pretrained model on the source task, the OTCE-based finetune algorithm is a two-step framework, as depicted in Fig. 5 and Algorithm 1. First, we optimize the source feature extractor $\hat{\theta}_s$ by minimizing the CE within one epoch. Formally,

$$\hat{\theta}_s^* = \arg\min_{\hat{\theta}_s} H_{\pi^*}(Y_t|Y_s)$$

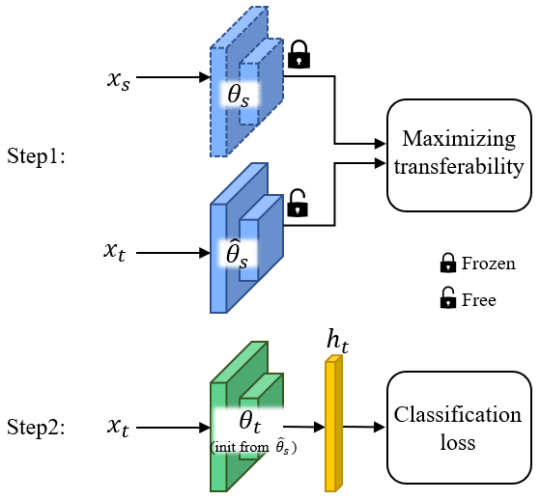

Fig. 5. Pipeline of our OTCE-based finetune method.

$$= -\arg\min_{\hat{\theta}_s} \sum_{y_t \in \mathcal{Y}_t} \sum_{y_s \in \mathcal{Y}_s} \hat{P}(y_s, y_t) \log \frac{\hat{P}(y_s, y_t)}{\hat{P}(y_s)} \tag{13}$$

where π^* is the optimal coupling matrix computed from (4). Joint label distribution $\hat{P}(y_s, y_t)$ and marginal $\hat{P}(y_s)$ are computed from (7) and (8). The computation of solving the OT problem with entropic regularizer [28] (4) is differentiable [30] since the iterations form a sequence of linear operations, so it can be implemented on the PyTorch framework as a specialized layer[1] of the neural network. After that, we initialize the target feature extractor θ_t from the optimized source weights $\hat{\theta}_s^*$, and then retrain the target model (θ_t, h_t) on the target training data using the cross-entropy loss function

$$\theta_t^*, h_t^* = \arg\max_{\theta_t, h_t} \sum_{i=1}^{m} \sum_{l=1}^{k} \mathbf{1}\{y_t^i = l\} \log \frac{\exp(h_t^l(\theta_t(x_t^i)))}{\sum_{j=1}^{k} \exp(h_t^j(\theta_t(x_t^i)))} \tag{14}$$

where m represents the number of target training samples and k is the number of the categories of the target task.

Note that we do not make it a one-step framework, i.e., simultaneously maximize the transferability and minimize the classification loss. Because optimizing two objectives simultaneously may cause gradient conflicts in mini-batch training, which will deteriorate the final classification performance.

B. OTCE-Based Domain Generalization

In contrast to the model finetuning task, the DG task aims to learn the generalizable feature representation exhibiting domain-irrelevant and task-irrelevant characteristics from multiple training domains. Therefore, the learned model can also achieve high classification accuracy when transferred to unseen tasks from unseen domains. We integrate our F-OTCE metric into a state-of-the-art DG method URL [27], [31] as a loss function to illustrate its effectiveness in boosting the DG algorithm.

More specifically, URL learns a universal model via distilling the common knowledge from multiple pretrained domain-specific models corresponding to each training domain. The universal model is required to achieve high classification accuracy in all training domains as well. Once the universal model is obtained, we can use it to extract feature representations for unseen few-shot classification tasks and make predictions via the nearest neighbor classifier (NCC).

Algorithm 1 OTCE-Based Finetune

Require: source dataset $D_s = \{(x_s^i, y_s^i)\}_{i=1}^m$
target dataset $D_t = \{(x_t^i, y_t^i)\}_{i=1}^n$
source feature extractor θ_s

Initialize $\hat{\theta}_s = \theta_s$
while sampling mini-batches within one epoch **do**
  Generate mini-batch $B_s = \{(\theta_s(x_s^i), y_s^i)\}_i^M$
  Generate mini-batch $B_t = \{(\hat{\theta}_s(x_t^i), y_t^i)\}_i^N$
  Update $\hat{\theta}_s$ via maximizing F-OTCE(B_s, B_t)
end while
Initialize $\theta_t = \hat{\theta}_s$
Randomly initialize h_t
while θ_t, h_t not converge **do**
  Update θ_t, h_t using equation (14)
end while

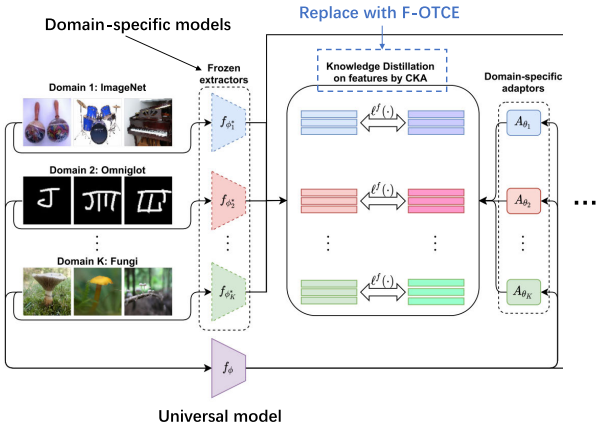

Fig. 6. Partial illustration of the URL framework [27]. We replace the CKA similarity with our F-OTCE metric.

In our opinion, the process of distilling knowledge from domain-specific models can be interpreted as maximizing the transferability between the domain-specific models and the universal model. Therefore, we propose to replace the knowledge distillation objective centered kernel alignment (CKA) [32] similarity used in URL with our F-OTCE metric, as illustrated in Fig. 6. Unlike CKA which solely focuses on minimizing feature differences, F-OTCE considers a wider range of task-specific information to minimize the label uncertainty of the universal model. We follow the default configuration of the URL algorithm. Please refer to [31] and [27] and the official codebase[2] for more details about the URL algorithm.

[1]https://github.com/dfdazac/wassdistance

[2]https://github.com/VICO-UoE/URL

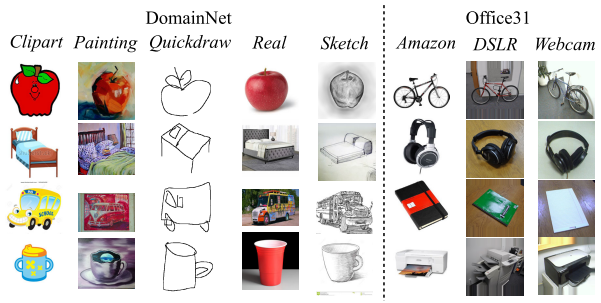

Fig. 7. Examples from the cross-domain datasets DomainNet and Office31, where images from different domains exhibit different image styles or are captured by different devices.

C. Few-Shot Classification Task Definition

We evaluate the effectiveness of our algorithms based on their transfer accuracy on few-shot classification tasks across domains. A few-shot classification task known as *C-way-K-shot* means that the support (training) set $S = \{(x^i, y^i)\}_{i=1}^{k\times C}$ contains k labeled instances from each of the C categories. The query set $Q = \{(x^i, y^i)\}_{i=1}^{q\times C}$ contains q samples per category and serves as the testing set to evaluate the classification accuracy of the model finetuned on the support set.

VI. Experiments

In this section, we begin by conducting quantitative evaluations of our proposed transferability metrics under various cross-domain cross-task transfer settings. We also explore their applications in source model selection and multisource feature fusion, as well as provide further analyses on computational efficiency, memory consumption, and hyperparameters. In addition, we conduct extensive evaluations of our proposed transferability-guided transfer learning methods including the OTCE-based finetune algorithm and the OTCE-based URL algorithm.

A. Evaluation on Transferability Estimation

1) Datasets: Our experiments are conducted on the data from the largest-to-date cross-domain dataset DomainNet [33] and popular Office31 [34] dataset. The DomainNet dataset contains 345-category images in five domains (image styles), i.e., Clipart (C), Painting (P), Quickdraw (Q), Real (R), and Sketch (S), and the Office31 contains 31-category images in three domains including Amazon (A), DSLR (D), and Webcam (W). Data examples are shown in Fig. 7.

2) Evaluation Criteria: To quantitatively evaluate the effectiveness of transferability metrics, we adopt the commonly used Spearman's rank correlation coefficient (Spearman's ρ coefficient) and the Kendall rank correlation coefficient (Kendall's τ coefficient) [35] to assess the correlation between the transfer accuracy and predicted transferability scores. Specifically, Spearman's ρ coefficient is defined as follows:

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{15}$$

where $d_i = R(\text{Acc}_i) - R(\text{Trf}_i)$ is the difference between the rankings of transfer accuracy Acc_i and transferability score Trf_i for the ith source-target task pair, and n represents the total number of task pairs.

Kendall's τ coefficient in our experiments is defined as follows:

$$\tau = \frac{2}{n(n-1)} \sum_{i<j} \text{sgn}(\text{Acc}_i - \text{Acc}_j)\text{sgn}(\text{Trf}_i - \text{Trf}_j). \tag{16}$$

Kendall's τ coefficient computes the number of concordant pairs minus the number of discordant pairs divided by the number of total pairs. A higher rank correlation indicates a more accurate transferability estimation result.

3) Transfer Settings: In the DomainNet dataset, we successively take each domain as the source domain and use the rest as target domains. For each target domain, we generate 100 target tasks by randomly sampling images in different categories. Then, we transfer the source models (ResNet-18 [36]) pretrained on all source domain data to each target task to obtain the ground-truth transfer accuracy. To investigate the performance of transferability metrics under various transfer configurations, three different transfer settings are considered, i.e., the *standard setting*, the *few-shot setting*, and the *fixed category size* setting.

1) *Standard Setting:* We keep all the training samples of the target task for retraining the source model. Meanwhile, the number of categories of target tasks ranges from 10 to 100. Thus, we totally conduct $5 \times 4 \times 100 = 2000$ cross-domain cross-task transfer tests.
2) *Few-Shot Setting:* As transfer learning is commonly used in scenarios where only a few labeled data are provided, it is worth evaluating the accuracy of transferability metrics on few-shot cases. The only difference with the *standard setting* is that we limit the target tasks to have only ten training samples per category.
3) *Fixed Category Size Setting:* As studied in [22], the intrinsic complexity of the target task, e.g., category size (number of categories), also affects the transfer accuracy. Usually, a larger category size makes the target task more difficult to learn from limited data. As a result, in the previous two settings, the intrinsic complexity of target tasks with different category sizes may overshadow the more subtle variations in the relatedness with the source task. To investigate whether the transferability metrics are capable of capturing those subtle variations, we propose a more challenging *fixed category size* setting, where all target tasks have the same $category\_size = 50$. Other configurations are the same as the *standard setting*.

Moreover, in the Office31 dataset, the DSLR and Webcam domains contain very few samples (∼15 samples per category) and suffer from severe category imbalance. Consequently, we construct two different configurations: *data-imbalanced* and *data-balanced* settings. Both of these two settings are few-shot, but the data-balanced setting permits a maximum of ten samples per category. Here, we only use Amazon as the source domain since the other two domains lack sufficient data to train generalizable source models. It is worth noting that we use the *average per-class accuracy* instead of the *overall*

TABLE II

QUANTITATIVE COMPARISONS EVALUATED BY SPEARMAN'S ρ COEFFICIENT AND KENDALL'S τ COEFFICIENT BETWEEN TRANSFERABILITY METRICS AND TRANSFER ACCURACY UNDER DIFFERENT CROSS-DOMAIN CROSS-TASK TRANSFER SETTINGS FOR IMAGE CLASSIFICATION TASKS. OUR PROPOSED JC-OTCE AND F-OTCE METRICS CONSISTENTLY OUTPERFORM STATE-OF-THE-ART AUXILIARY-FREE METRICS. MEANWHILE, THE JC-OTCE ACHIEVES COMPARABLE PERFORMANCE TO THE AUXILIARY-BASED OTCE

Setting	Source domain	Target domain	Spearman / Kendall correlation coefficient						
			Auxiliary-based	Auxiliary-free					
			OTCE [22]	JC-OTCE	F-OTCE	LEEP [21]	NCE [20]	H-score [7]	LogME [23]
Standard (Retrain head)	C	P,Q,R,S	0.976 / 0.861	0.965 / 0.836	**0.966 / 0.839**	0.932 / 0.779	0.825 / 0.670	0.920 / 0.748	0.867 / 0.667
	P	C,Q,R,S	0.977 / 0.868	**0.966 / 0.837**	0.960 / 0.822	0.906 / 0.743	0.849 / 0.686	0.937 / 0.777	0.929 / 0.761
	Q	C,P,R,S	0.961 / 0.826	0.962 / **0.833**	**0.963** / 0.832	0.953 / 0.810	0.943 / 0.793	0.942 / 0.784	0.912 / 0.744
	R	C,P,Q,S	0.975 / 0.863	**0.965 / 0.836**	0.951 / 0.808	0.910 / 0.747	0.872 / 0.707	0.942 / 0.786	0.855 / 0.670
	S	C,P,Q,R	0.969 / 0.842	0.965 / 0.834	**0.967 / 0.839**	0.965 / 0.834	0.962 / 0.830	0.950 / 0.802	0.908 / 0.733
Standard (Finetune)	C	P,Q,R,S	0.932 / 0.766	**0.900 / 0.713**	0.884 / 0.689	0.814 / 0.618	0.664 / 0.517	0.889 / 0.704	0.890 / 0.695
	P	C,Q,R,S	0.803 / 0.612	0.874 / 0.698	**0.880** / 0.698	0.850 / 0.655	0.797 / 0.613	0.876 / **0.716**	0.848 / 0.664
	Q	C,P,R,S	0.896 / 0.719	**0.906 / 0.732**	0.895 / 0.719	0.880 / 0.696	0.874 / 0.684	0.873 / 0.686	0.891 / 0.699
	R	C,P,Q,S	0.912 / 0.732	**0.905** / 0.725	0.882 / 0.689	0.821 / 0.616	0.770 / 0.571	0.902 / **0.727**	0.876 / 0.681
	S	C,P,Q,R	0.923 / 0.752	**0.932 / 0.767**	0.929 / 0.763	0.927 / 0.766	0.925 / 0.757	0.915 / 0.747	0.894 / 0.706
		Average	0.932 / 0.784	**0.934 / 0.782**	0.928 / 0.770	0.896 / 0.727	0.849 / 0.682	0.915 / 0.748	0.887 / 0.702
Few-shot (Retrain head)	C	P,Q,R,S	0.926 / 0.756	**0.926 / 0.757**	0.909 / 0.729	0.836 / 0.640	0.745 / 0.576	0.762 / 0.567	0.731 / 0.524
	P	C,Q,R,S	0.931 / 0.772	**0.928 / 0.769**	0.886 / 0.701	0.803 / 0.618	0.746 / 0.575	0.811 / 0.608	0.849 / 0.649
	Q	C,P,R,S	0.821 / 0.631	0.856 / 0.673	0.829 / 0.636	0.798 / 0.602	0.782 / 0.584	0.813 / 0.614	**0.866 / 0.682**
	R	C,P,Q,S	0.929 / 0.769	**0.897 / 0.724**	0.853 / 0.666	0.770 / 0.589	0.728 / 0.559	0.845 / 0.652	0.774 / 0.574
	S	C,P,Q,R	0.914 / 0.742	**0.902 / 0.725**	0.895 / 0.710	0.872 / 0.680	0.872 / 0.679	0.838 / 0.645	0.867 / 0.684
		Average	0.905 / 0.734	**0.902 / 0.729**	0.875 / 0.689	0.815 / 0.625	0.775 / 0.595	0.814 / 0.618	0.818 / 0.623
Fixed category size (Retrain head)	C	P,Q,R,S	0.701 / 0.500	**0.695 / 0.498**	0.687 / 0.487	0.685 / 0.486	0.666 / 0.472	-0.438 / -0.290	-0.222 / -0.151
	P	C,Q,R,S	0.670 / 0.485	**0.665 / 0.479**	0.631 / 0.448	0.630 / 0.446	0.612 / 0.430	-0.529 / -0.371	-0.043 / -0.039
	Q	C,P,R,S	0.341 / 0.225	**0.381 / 0.261**	0.316 / 0.211	0.210 / 0.136	0.291 / 0.191	-0.256 / -0.172	0.066 / 0.037
	R	C,P,Q,S	0.637 / 0.455	**0.695 / 0.498**	0.598 / 0.415	0.587 / 0.407	0.586 / 0.406	-0.094 / -0.063	-0.382 / -0.252
	S	C,P,Q,R	0.428 / 0.292	**0.497 / 0.343**	0.436 / 0.299	0.404 / 0.277	0.432 / 0.298	-0.247 / -0.164	0.027 / 0.006
		Average	0.555 / 0.391	**0.587 / 0.416**	0.534 / 0.372	0.503 / 0.350	0.517 / 0.359	-0.313 / -0.212	-0.111 / -0.080
Imbalanced (Retrain head)	A	D	- / -	**0.844 / 0.646**	0.829 / 0.627	0.822 / 0.616	0.801 / 0.589	0.674 / 0.476	0.785 / 0.593
	A	W	- / -	0.847 / 0.651	0.850 / 0.653	**0.862 / 0.665**	0.859 / 0.663	0.657 / 0.489	0.787 / 0.590
Balanced (Retrain head)	A	D	- / -	0.822 / **0.627**	**0.824** / 0.625	0.796 / 0.592	0.783 / 0.572	0.574 / 0.393	0.747 / 0.536
	A	W	- / -	**0.879 / 0.686**	0.871 / 0.673	0.872 / 0.674	0.856 / 0.656	0.669 / 0.477	0.797 / 0.604
		Average	- / -	**0.848 / 0.653**	0.844 / 0.645	0.838 / 0.637	0.825 / 0.620	0.644 / 0.459	0.779 / 0.581

Bold denotes the best result, and underline denotes the 2nd best result.

accuracy for representing the transfer performance under the data-imbalanced setting.

For all the settings above, we adopt a stochastic gradient descent (SGD) optimizer with a learning rate of 0.01 to optimize the cross-entropy loss for 100 epochs during the transfer training phase.

4) Results: Quantitative comparisons with state-of-the-art auxiliary-free transferability metrics including LEEP [21], NCE [20], H-score [7], and LogME [23] are shown in Table II, and visual comparisons are illustrated in Fig. 8. First, we can see that both our JC-OTCE and F-OTCE metrics consistently outperform recent LEEP, NCE, H-score, and LogME metrics on all three transfer settings. In particular, our JC-OTCE metric achieves (7.3%, 14.7%, 4.5%, 11.4%) and (16.6%, 22.5%, 18.0%, 17.0%) average gains on Kendall correlation compared to LEEP, NCE, H-score, and LogME, respectively, under the *standard* setting and the *few-shot* setting. Moreover, the H-score metric and the LogME metric failed under the more challenging *fixed category size* setting, where they showed a negative correlation with the transfer accuracy.

Second, the JC-OTCE metric outperforms the F-OTCE metric with an average 5.4% gain on Kendall correlation, which shows that involving the label distance in computing the data correspondences makes the transferability estimation more accurate. Meanwhile, the JC-OTCE metric performs comparably to the original OTCE metric in accuracy, while the former one is evidently more efficient and has fewer restrictions.

Basically, we can conclude that OTCE $\approx$ JC-OTCE $>$ F-OTCE in accuracy and OTCE $<$ JC-OTCE $<$ F-OTCE in efficiency. These three metrics can be applied flexibly according to the needs of different practical situations.

B. Efficiency Analysis

Given d-dimensional extracted features of m source samples and n target samples, assuming that $|\mathcal{Y}_s|, |\mathcal{Y}_t| < \min(m, n)$, the computational complexity of F-OTCE is $O(mn \max\{d, k\})$, where k is the number of Sinkhorn iterations in the OT computation. Specifically, the worst-case complexity of computing the cost matrix between source and target samples is $O(mnd)$. Solving the OT problem by Sinkhorn algorithm with ϵ accuracy has complexity $O(mnk) = O(2\ mn\|c\|_\infty^2/(\lambda\epsilon))$ [37], where $\|c\|_\infty = \sup_{(z_s,z_t)\in\mathcal{Z}^2} c(z_s, z_t)$ is the maximum cost between source and target sample features and λ is the weighting coefficient of the entropic regularizer. In practice,