On Balancing Bias and Variance in Unsupervised Multi-Source-Free Domain Adaptation

Maohao Shen [1] **Yuheng Bu** [2] **Gregory Wornell** [1]

Abstract

Due to privacy, storage, and other constraints, there is a growing need for unsupervised domain adaptation techniques in machine learning that do not require access to the data used to train a collection of source models. Existing methods for multi-source-free domain adaptation (MSFDA) typically train a target model using pseudo-labeled data produced by the source models, which focus on improving the pseudo-labeling techniques or proposing new training objectives. Instead, we aim to analyze the fundamental limits of MSFDA. In particular, we develop an information-theoretic bound on the generalization error of the resulting target model, which illustrates an inherent bias-variance trade-off. We then provide insights on how to balance this trade-off from three perspectives, including domain aggregation, selective pseudo-labeling, and joint feature alignment, which leads to the design of novel algorithms. Experiments on multiple datasets validate our theoretical analysis and demonstrate the state-of-art performance of the proposed algorithm, especially on some of the most challenging datasets, including Office-Home and DomainNet.

1. Introduction

Machine learning models trained in a standard supervised manner suffer from the problem of domain shift (Quiñonero-Candela et al., 2008), i.e., directly applying the model trained on the source domain to a distinct target domain usually results in poor generalization performance. Unsupervised Domain Adaptation (UDA) techniques have been proposed to mitigate this issue by transferring the knowledge learned from a labeled source domain to an unlabeled target domain. One prevailing UDA strategy to resolve the domain shift issue is domain alignment, i.e., learning domain-invariant features either by minimizing the discrepancy between the source and target data (Long et al., 2015; 2018; Peng et al., 2019) or through adversarial training (Ganin & Lempitsky, 2015; Tzeng et al., 2017). However, traditional UDA methods require access to labeled source data and only apply to the single source domain adaptation, which cannot fulfill the emerging challenges in real-world applications.

In practice, the source data might not be available due to various reasons: privacy preservation, i.e., the data that contains sensitive information, such as health and financial status, is unsuitable to be made public; and storage limitations, i.e., large-scale datasets, such as high-resolution videos, require substantial storage space. Due to these practical concerns, the source-free domain adaptation (SFDA) problem has attracted increasing attentions (Yang et al., 2020; Kim et al., 2020; Liang et al., 2020; Li et al., 2020), which aims to address the data-free challenge by adapting the pretrained source model to the unlabeled target domain.

The other practical concern is that the source data is usually collected from multiple domains with different underlying distributions, such as the street scene from different cities (Cordts et al., 2016) and the biomedical images with different modalities (Dou et al., 2018). Taking such practical consideration into account, multi-source domain adaptation (MSDA) (Guo et al., 2018; Peng et al., 2019) aims to adapt to the target domain by properly aggregating the knowledge from multiple source domains.

A more challenging scenario is to combine both the data-free and multi-source settings, i.e., the source data collected from multiple domains with distinct distributions is not accessible due to some practical constraints. For example, federated learning (Truong et al., 2021) aggregates the information learned from a group of heterogeneous users. To preserve user privacy, the data of each user is stored locally, and only the trained models are transmitted to the central server.

We consider the Multi-Source-Free Domain Adaptation (MSFDA) problem to overcome these two challenges. The

[1]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, USA [2]Department of Electrical & Computer Engineering, University of Florida, Gainesville, USA. Correspondence to: Maohao Shen <maohao@mit.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

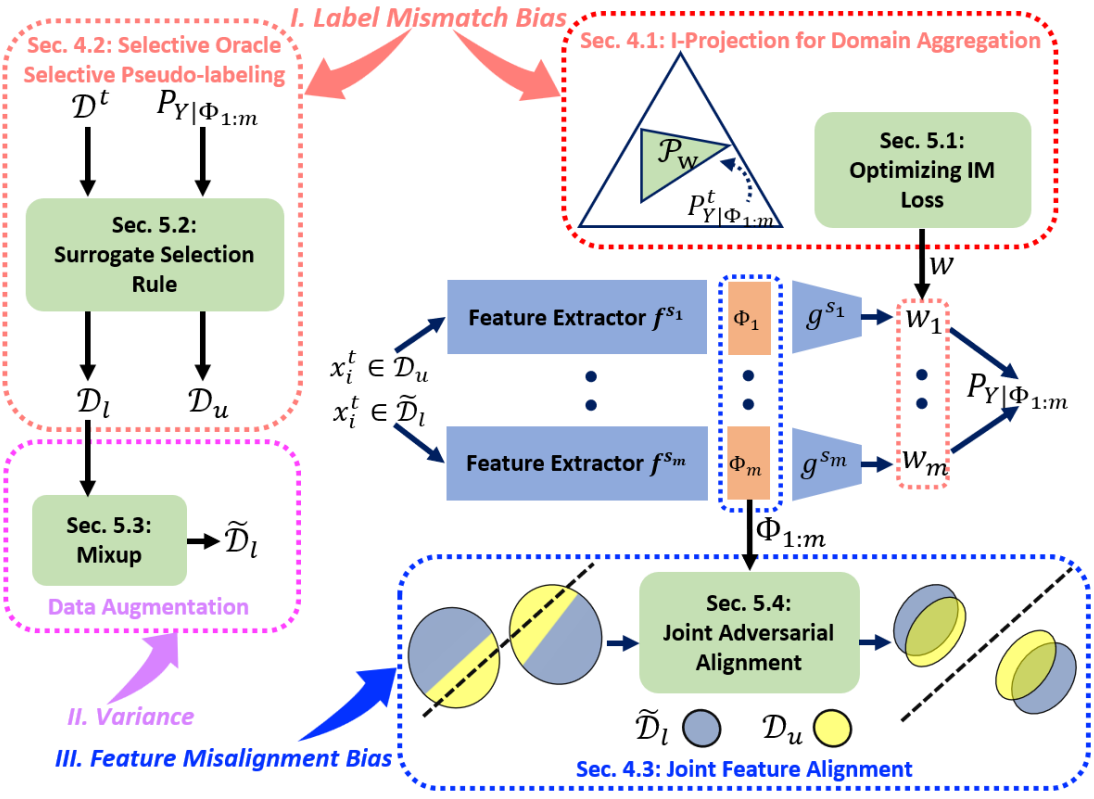

Figure 1. Schematic of our theoretical insights and algorithm design. More details on label mismatch bias, feature misalignment bias, and variance can be found in the corresponding sections, as shown in the figure.

MSFDA problem is less explored, and few methods have been proposed. Existing methods either propose new training objectives motivated by specific intuitions without a theoretical justification (Yang et al., 2021a; Kundu et al., 2022), or leverage on generating pseudo-labels using source models but not fully address the bias induced by the noisy pseudo-labeling procedure (Ahmed et al., 2021; Dong et al., 2021). In addition, unlike traditional domain adaptation approaches, none of these methods can explicitly address the domain shift issue in source and target domains. To this end, we aim to understand the fundamental limit of the MSFDA problem through theoretical analysis and then draw some insights for new algorithm design. We show that three factors control the generalization error of the MSFDA problem: *label mismatch bias*, *feature misalignment bias*, and *variance* depending on the number of training samples. As in Figure 1, we demonstrate how to balance the trade-off by providing three crucial insights: i.e., (1) Appropriate domain aggregation of multiple source domains can reduce the *label mismatch bias*, (2) selective pseudo-labeling a subset of data for training can further balance the *label mismatch bias* and the variance, and (3) Utilizing a joint feature alignment strategy to explicitly address the domain shift issue by reducing the *feature misalignment bias*.

Summary of contributions: (1)We develop an information-theoretic generalization error bound for the MSFDA problem, demonstrating an inherent bias-variance trade-off. (2) We provide theoretical understandings and draw insights into how to balance the bias and variance trade-off from three perspectives. (3) Motivated by our theoretical analysis, we empirically study the performance limit of the MSFDA problem by providing a performance upper bound and propose a novel algorithm. (4) Experiments across multiple representative benchmark datasets validate our theoretical results and demonstrate the superior performance of the proposed algorithm over existing methods.

2. Related Work

Multi-source Domain Adaptation: Multi-source domain adaptation aims to transfer knowledge from multiple distinct source domains to a target domain. Early theoretical works provide theoretical guarantees for later empirical works by formally characterizing the connection between the source and target domains. (Ben-David et al., 2010) introduces the $\mathcal{H}\Delta\mathcal{H}$ distance to measure the discrepancy between the target and source domains, and (Mansour et al., 2008) assumes that the target distribution can be approximated by

a linear combination of source distributions. Many existing algorithms aim to mitigate the distribution shift issue between source and target domains. Discrepancy-based methods try to align the domain distribution by minimizing discrepancy loss, such as maximum mean discrepancy (MMD) (Guo et al., 2018), Rényi-divergence (Hoffman et al., 2018), moment distance (Peng et al., 2019), and a combination of different discrepancy metrics (Guo et al., 2020). Adversarial methods align the features between source and target domains by training a feature extractor that fools the discriminator under different loss functions, including $\mathcal{H}$-divergence (Zhao et al., 2018), traditional GAN loss (Xu et al., 2018), and Wasserstein distance (Li et al., 2018; Wang et al., 2019; Zhao et al., 2020). Moreover, the domain weighting strategy is also widely used to quantify the contribution of each source domain, including uniform weights (Zhu et al., 2019), source model accuracy-based weights (Peng et al., 2019), Wasserstein distance-based weights (Zhao et al., 2020), and aggregating multiple domains using graph model (Wang et al., 2020)

Source-free Domain Adaptation: All the methods mentioned above require source data and cannot be directly applied to the data-free setting, and recent efforts have been made to tackle the source-free problem. (Xia et al., 2021) constructs a target-specific classifier to mitigate the domain discrepancy. Adversarial learning-based methods utilize generative models by either generating new samples from the source domain (Kurmi et al., 2021) or generating labeled samples from target distribution by conditional GAN (Li et al., 2020). Another strategy is using the pseudo-labeling technique. (Liang et al., 2020) uses self-supervised pseudo-labeling and maximizes the mutual information loss. Similarly, (Kim et al., 2020) proposes a confidence-based filtering method to further improve the quality of pseudo-labels.

Multi-source-free Domain Adaptation: To overcome both data-free and multi-source challenges, (Ahmed et al., 2021) adapts the self-supervised pseudo-labeling method in (Liang et al., 2020) to the multi-source setting. Similarly, (Dong et al., 2021) also focuses on improving the pseudo-labeling algorithm with a confident-anchor-induced pseudo-label generator. Other works focus on designing new training objectives, including (Yang et al., 2021a;b) that proposes a nearest-neighbor-based regularizer to encourage prediction consistency, and (Kundu et al., 2022) that designs new data augmentation techniques to balance the discriminability and transferability trade-off for source free domain adaptation. In contrast to many prior works that are often ad-hoc and lack theoretical justification, our work aims to design novel MSFDA algorithms by analyzing the MSFDA problem using information-theoretical tools. While (Dong et al., 2021) also conducts theoretical analysis, their results require specific assumptions, such as the meta-assumption of the data distribution. Our theoretical result is more general and does not require additional assumptions regarding the data distribution or models.

3. Problem Formulation

Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ denotes the input space and $\mathcal{Y} = \{1, \cdots, K\}$ denotes the label space. A domain is defined by a joint distribution P_{XY} on the instance space $\mathcal{Z}$. In this work, we aim to jointly adapt multiple pretrained models corresponding to m different source domains $\{P_{XY}^{s_j}\}_{j=1}^m$ to a new target domain P_{XY}^t. Let $\boldsymbol{h}^{s_j} : \mathcal{X} \to \Delta^{K-1}$ denote the pretrained model for source domain j, which is a function predicting the conditional distribution $P_{Y|X}^{s_j}$ in probability simple Δ^{K-1}, i.e., $\boldsymbol{h}^{s_j} = [h_1^{s_j}, \cdots, h_K^{s_j}]^\top$, $\sum_{k=1}^K h_k^{s_j} = 1$ and $h_k^{s_j} \geq 0$. Each pretrained model can be decomposed into a feature extractor $\boldsymbol{f}^{s_j} : \mathcal{X} \to \mathcal{F}_j$, followed by a classifier $\boldsymbol{g}^{s_j} : \mathcal{F}_j \to \Delta^{K-1}$, where $\mathcal{F}_j$ denotes the representation space of each source model. Thus, we can denote the prediction of input $\boldsymbol{x}$ using model $\boldsymbol{h}^{s_j}$ as $\boldsymbol{h}^{s_j}(\boldsymbol{x}) = (\boldsymbol{g}^{s_j} \circ \boldsymbol{f}^{s_j})(\boldsymbol{x})$. For any feature representations $\phi_j \in \mathcal{F}_j$, each classifier $\boldsymbol{g}^{s_j}(\boldsymbol{\phi}_j)$ induces a conditional distribution $P_{Y|\Phi_j=\phi_j}^{s_j}$.

Denote $\mathcal{D}^t \triangleq \{\boldsymbol{x}_i^t\}_{i=1}^n$ as the unlabeled target domain dataset, where $\boldsymbol{x}_i^t$ are i.i.d. generated from the target marginal distribution P_X^t. For any $\boldsymbol{x}_i^t \in \mathcal{D}^t$, we denote $\boldsymbol{\phi}_i^{s_j} = \boldsymbol{f}^{s_j}(\boldsymbol{x}_i^t)$ as the feature representation of sample i based on source model s_j. We aim to construct a target model $\boldsymbol{h}$ that aggregates information from multiple source models by modifying the feature mappings $\boldsymbol{f}^{s_j}$. Denote the final prediction as $\hat{y} = \boldsymbol{h}\left(\{\boldsymbol{f}^{s_j}(\boldsymbol{x})\}_{j=1}^m\right) = \boldsymbol{h}\left(\{\boldsymbol{\phi}_j\}_{j=1}^m\right)$. For the loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$, our ultimate goal is to obtain a target model that minimizes the population risk of the target domain, i.e., $\mathcal{L}_P(\boldsymbol{h}, P_{XY}^t) \triangleq \mathbb{E}_{P_{XY}^t}[\ell\left(\boldsymbol{h}\left(\{\boldsymbol{f}^{s_j}(X)\}_{j=1}^m\right), Y\right)]$.

4. Theoretical Analysis and Insights

In this section, we provide a theoretical analysis of the multi-source-free domain adaptation problem. We show that there exists an inherent bias and variance trade-off that needs to be balanced in our algorithm design. Suppose that for some unlabeled target samples $\boldsymbol{x}_i^t \in \mathcal{D}^t$, we can obtain its pseudo-label $\tilde{y}_i^t$ by leveraging the pretrained models, and denote the subset of pseudo-labeled data as $\mathcal{D}_l \triangleq \{(\boldsymbol{x}_i^t, \tilde{y}_i^t)\}_{i=1}^{n_l}$. To ensure that model learned by minimizing the empirical risk over $\mathcal{D}_l$, i.e.,

$$\mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l) \triangleq \frac{1}{n_l} \sum_{i=1}^{n_l} \ell\left(\boldsymbol{h}\left(\{\boldsymbol{\phi}_i^{s_j}\}_{j=1}^m\right), \tilde{y}_i^t\right) \tag{1}$$

generalizes well to the target domain, we have the following upper bound on generalization error, i.e., the difference between the population risk and the empirical risk.

Theorem 4.1. *(proved in Appendix A.1) Suppose that the samples of $\mathcal{D}_l$ are i.i.d. generated from the distribution $P^{\mathcal{D}_l}_{XY}$, the function space $\mathcal{H}$ has finite Natarajan dimension $d_N(\mathcal{H})$, then for any loss function bounded in $[0, M]$ and any $\boldsymbol{h}(\cdot) \in \mathcal{H}$, there exists a constant C such that with probability $1-\delta$,*

$$\begin{aligned}&\mathcal{L}_P(\boldsymbol{h}, P^t_{XY}) - \mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l)\\ \leq &\frac{M}{\sqrt{2}}\sqrt{\underbrace{D(P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}\|P^t_{Y|\Phi_{1:m}}|P^{\mathcal{D}_l}_{\Phi_{1:m}})}_{\text{Bias: label distribution mismatch}} + \underbrace{D(P^{\mathcal{D}_l}_{\Phi_{1:m}}\|P^t_{\Phi_{1:m}})}_{\text{Bias: feature misalignment}}}\\ &+ C\underbrace{\sqrt{\frac{d_N(\mathcal{H})\log K + \log\frac{1}{\delta}}{n_l}}}_{\text{Variance: number of labeled sample}}, \end{aligned} \tag{2}$$

where $\Phi_{1:m} \triangleq \{\boldsymbol{f}^{s_j}(X)\}_{j=1}^m$ are random vectors induced by different features mappings $\boldsymbol{f}^{s_j}: \mathcal{X} \to \mathcal{F}_j$.

Theorem 4.1 states that the generalization error can be controlled by a bias and variance trade-off. The first term can be viewed as bias, where KL divergence measures the discrepancy between two distributions. The first term can be further decomposed into two sub-terms: the bias due to the mismatch between the pseudo label distribution and the target label distribution; and the bias due to the misalignment of marginal feature distribution in the representation space. The second term in (2) can be interpreted as the variance since it only depends on the hypothesis space $\mathcal{H}$ and the number of the pseudo-labeled samples n_l.

In the following, we discuss how to balance the bias-variance trade-off by improving each term in the generalization error bound provided in Theorem 4.1. First, we make the connection between minimizing the *label mismatch bias* and domain aggregation of multiple source domains. Second, we reveal that selective pseudo-labeling is essential to prevent the *label mismatch bias* from being unbounded while also balancing the variance term. Finally, we show how a joint feature alignment approach is crucial for the MSFDA problem to reduce the *feature misalignment bias*.

4.1. Domain Aggregation

To leverage the pretrained source models and construct a labeling distribution $P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}$ that reduces the *label mismatch bias* $D(P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}\|P^t_{Y|\Phi_{1:m}}|P^{\mathcal{D}_l}_{\Phi_{1:m}})$, we need to aggregate the information from multiple source models. One naive approach is to weight the predictions from multiple models $P^{s_j}_{Y|\Phi_j}$ uniformly (Zhu et al., 2019). However, each source domain may have different transferability, and treating them equally is sub-optimal. Here, we consider a mixture distribution with non-negative domain weights $\boldsymbol{w}$, i.e., $P^w_{Y|\Phi_{1:m}} = \sum_{j=1}^m \boldsymbol{w}_j P^{s_j}_{Y|\Phi_j}$ to aggregate multiple source models. We denote the convex set of such mixture distribution as $\mathcal{P}_w = \{q \mid q = \sum_{j=1}^m \boldsymbol{w}_j P^{s_j}_{Y|\Phi_j}, \sum_{j=1}^m \boldsymbol{w}_j = 1\}$. In order to minimize the *label mismatch bias*, the pseudo label distribution $P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}$ should minimize its KL divergence to the target label distribution $P^t_{Y|\Phi_{1:m}}$, i.e.,

$$P^{\mathcal{D}_l}_{Y|\Phi_{1:m}} = \arg\min_{P\in\mathcal{P}_w} D\left(P\|P^t_{Y|\Phi_{1:m}}\right). \tag{3}$$

Notice that this is equivalent to let $P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}$ to be the *I-projection* (Csiszár, 1984) of target distribution onto $\mathcal{P}_w$, where we have the following proposition,

Proposition 4.1. *(proved in Appendix A.2) Let $\{P^{s_j}_{Y|\Phi_j}\}_{j=1}^m$ be a collection of source models, and let $P^t_{Y|\Phi_{1:m}}$ be any target label distribution, then there exists a mixture model with weights $\boldsymbol{w}^*$ such that*

$$D\Big(\sum_{j=1}^m \boldsymbol{w}^*_j P^{s_j}_{Y|\Phi_j}\|P^t_{Y|\Phi_{1:m}}\Big) \leq D\big(P^{s_j}_{Y|\Phi_j}\|P^t_{Y|\Phi_{1:m}}\big)$$

holds for any j.

Proposition 4.1 states that the mixture of source models can induce lower bias compared to using any single model, which justifies the benefits of aggregating the multiple source models to reduce the *label mismatch bias*. Besides, as the mixture distribution defines a convex hull over the probability simplex, increasing the number of source models will neither shrink the convex hull nor raise the approximation error. Therefore, utilizing more source domains with the optimal mixture weights $\boldsymbol{w}^*$ will benefit the domain adaptation problem. In practice, the target label distribution $P^t_{Y|\Phi_{1:m}}$ in (3) is not available, and we discuss how to approximate the mixture distribution in Section 5.1.

4.2. Selective Pseudo-labeling

Once we obtain the pseudo labeling distribution $P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}$, the next step is to generate pseudo-labels using this distribution. Most existing work generate pseudo-labels for all samples in $\mathcal{D}^t$, and the joint distribution of $\mathcal{D}_l$ is given by $P^{\mathcal{D}_l}_{\Phi_{1:m},Y} = P^t_{\Phi_{1:m}} \otimes P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}$. Although generating pseudo-labels for the entire $\mathcal{D}^t$ implies $n_l = n$, which reduces the variance term in Theorem 4.1, the *label mismatch bias* might be unbounded due to the approximation error in equation 3. To see this, if $P^t_{Y|\Phi_{1:m}}(Y = y|\{\Phi_j = \boldsymbol{\phi}^{s_j}_i\}_{j=1}^m) = 0$ and $P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}(Y = y|\{\Phi_j = \boldsymbol{\phi}^{s_j}_i\}_{j=1}^m) \neq 0$ for some $\boldsymbol{x}_i \in \mathcal{D}^t$ due to model mismatch, it leads to a large *label mismatch bias*, i.e., $D(P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}\|P^t_{Y|\Phi_{1:m}}|P^{\mathcal{D}_l}_{\Phi_{1:m}}) = \infty$.

The aforementioned issue can be mitigated by only generating pseudo-labels for a subset of $\mathcal{D}^t$. We start our discussion by assuming the following selective oracle, which gives a perfect subset selection. Such an assumption helps us to understand the performance limits of the MSFDA problem

revealed in Theorem 4.1, which is further validated by the empirical results provided in 6.2.

Definition 1. *(Selective Oracle) Given the labeling distribution for target domain data $P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}(\cdot|\cdot)$, the selective oracle identifies a subset of data $\mathcal{X}_{\mathcal{D}_l}$ for pseudo labeling, which satisfies the following criterion:*

$$P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}(\cdot|\{\Phi_j = \boldsymbol{f}^{s_j}(\boldsymbol{x})\}_{j=1}^m) = P^{t}_{Y|\Phi_{1:m}}(\cdot|\{\Phi_j = \boldsymbol{f}^{s_j}(\boldsymbol{x})\}_{j=1}^m), \forall \boldsymbol{x} \in \mathcal{X}_{\mathcal{D}_l}. \quad (4)$$

With the selective oracle, the *label mismatch bias* can be bounded and reduced to a negligible value. Note that the variance term is determined by the cardinality of subset $\mathcal{X}_{\mathcal{D}_l}$, i.e., $n_l = |\mathcal{X}_{\mathcal{D}_l}|$. The oracle induces a small variance term by selecting a sufficient number of data in $\mathcal{X}_{\mathcal{D}_l}$ while also ensuring a bounded *label mismatch bias*. In practice, when such an oracle is not available, we propose a surrogate selection technique in 5.2 to balance such a trade-off.

4.3. Joint Feature Alignment

Notice that there exists the distribution shift between data marginal distributions in the joint representation space for $\mathcal{D}_l$ and $\mathcal{D}_t$, i.e., $P^{\mathcal{D}_l}_{\Phi_{1:m}} \neq P^{\mathcal{D}_t}_{\Phi_{1:m}}$, where we have the following remark about the *feature misalignment bias*.

Remark 1. *(proved in Appendix A.3) The feature misalignment bias in the joint representation space induced by multiple source models is the upper bound of the average feature misalignment bias across each representation space, i.e.,*

$$D(P^{\mathcal{D}_l}_{\Phi_{1:m}} \| P^{t}_{\Phi_{1:m}}) \geq \frac{1}{m}\sum_{j=1}^{m} D(P^{\mathcal{D}_l}_{\Phi_j} \| P^{t}_{\Phi_j}). \quad (5)$$

Remark 1 states that $D(P^{\mathcal{D}_l}_{\Phi_{1:m}} \| P^{t}_{\Phi_{1:m}}) = 0$ ensures $\sum_{j=1}^m D(P^{\mathcal{D}_l}_{\Phi_j} \| P^{t}_{\Phi_j}) = 0$, but not vice versa. This highlights that the feature alignment needs to be conducted in the joint representation space instead of in each source domain separately to further reduce the *feature misalignment bias*, which motivates us to propose a joint adversarial feature alignment loss in 5.4.

5. Practical Algorithm

The theoretical analysis presented above sheds light on how to balance the bias and variance terms as revealed by the generalization error bound in Theorem 4.1. However, in practice, the target label information used in the previous analysis is not available. To this end, we introduce practical algorithms that incorporate these insights. First, we discuss methods for learning domain aggregation weights without target domain labels. Then, we propose a surrogate selection rule for the selective Oracle and introduce a joint adversarial feature alignment approach. Finally, we present an overall algorithm for solving the MSFDA problem.

5.1. Learning Domain Aggregation Weights

Obtaining the optimal labeling distribution $P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}$ through *I-projection* (equation 3) is not possible as the ground-truth target distribution is not accessible. However, we can make the following assumptions about the optimal mixture distribution. In general, the predictions of target data should be individually confident and globally diverse. First, for any input data $\boldsymbol{x}_i^t \in \mathcal{D}^t$, the mixture labeling distribution $\sum_{j=1}^m \boldsymbol{w}_j P^{s_j}_{Y|\Phi_j}(\cdot|\Phi_j = \boldsymbol{\phi}_i^{s_j})$ should be confident and lie in one of the corners of the probability simplex. Thus, it is appropriate to minimize the entropy of the mixture labeling distribution of individual input data. Second, the marginal label distribution, i.e., $P^{\mathcal{D}_l}_Y = \mathbb{E}_{\Phi_{1:m}}[P^{\mathcal{D}_l}_{Y|\Phi_{1:m}}] \approx \frac{1}{n}\sum_{i=1}^n \sum_{j=1}^m \boldsymbol{w}_j P^{s_j}_{Y|\Phi_j}(\cdot|\Phi_j = \boldsymbol{\phi}_i^{s_j})$ should be diverse and close to a uniform distribution. Notice that minimizing the KL divergence between the label distribution and a uniform distribution is equivalent to its entropy maximization.

Based on these intuitions, we propose to learn the weights $\boldsymbol{w}$ by optimizing the information maximization loss (Krause et al., 2010; Hu et al., 2017),

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \frac{1}{n}\sum_{i=1}^n \mathrm{H}\Big(\sum_{j=1}^m \boldsymbol{w}_j P^{s_j}_{Y|\Phi_j}(\cdot|\Phi_j = \boldsymbol{\phi}_i^{s_j})\Big) - \mathrm{H}\Big(\frac{1}{n}\sum_{i=1}^n\sum_{j=1}^m \boldsymbol{w}_j P^{s_j}_{Y|\Phi_j}(\cdot|\Phi_j = \boldsymbol{\phi}_i^{s_j})\Big), \quad (6)$$

where H denotes the entropy function.

5.2. Surrogate Selection Rule

The key to improving pseudo-labeling is to balance the trade-off between the second bias term and the variance term by only assigning pseudo-labels to a subset of the data $\mathcal{D}_l \triangleq \{(\boldsymbol{x}_i^t, \tilde{y}_i^t)\}_{i=1}^{n_l}$ using the selective oracle. However, the oracle, which ensures that the labeled subset $\mathcal{D}_l$ contains only correctly pseudo-labeled data, is not available in practice. To overcome this, we propose a simple yet effective selection rule as the surrogate. Specifically, we adopt a pseudo-label denoising trick to improve the quality of labeling distribution and a confidence query strategy to select new data for pseudo-labeling.

Pseudo-Label Denoising Pseudo-labeling techniques are widely used in semi-supervised learning (Lee et al., 2013; Shi et al., 2018; Rizve et al., 2021). However, the predictions made by the pretrained source models on target data can be very noisy, so it is crucial to combine them with other labeling criteria to improve the quality of the pseudo-labels. Inspired by (Zhang et al., 2021), we propose a prototype-based pseudo-label denoising method.

For each target sample $\boldsymbol{x}_i^t \in \mathcal{D}_t$, its pseudo-label $\tilde{y}_i^t$ is generated based on two different criteria: (1) the mixture

label distribution directly obtained from source models, i.e., $\boldsymbol{h}\left(\{\boldsymbol{f}^{s_j}(\boldsymbol{x})\}_{j=1}^m\right) = \sum_{j=1}^m \boldsymbol{w}_j^* P_{Y|\Phi_j}^{s_j}(\cdot|\Phi_j = \boldsymbol{\phi}_j)$, and (2) the mixture label distribution obtained using the clustering structure in the representation space, i.e., $\sum_{j=1}^m \boldsymbol{w}_j q^{s_j}(\boldsymbol{x}_i^t)$, where the label distribution $[q_1^{s_j}, \cdots, q_K^{s_j}]^\top$ is obtained using the distance to prototypes in feature space by ignoring the classifier $\boldsymbol{g}^{s_j}$, i.e.,

$$q_k^{s_j}(\boldsymbol{x}_i^t) = \frac{\exp\left(-\left\|\boldsymbol{\phi}_i^{s_j} - \boldsymbol{\eta}_k^{s_j}\right\|/\tau\right)}{\sum_{k'=1}^K \exp\left(-\left\|\boldsymbol{\phi}_i^{s_j} - \boldsymbol{\eta}_{k'}^{s_j}\right\|/\tau\right)}. \tag{7}$$

Where τ is the softmax temperature, and $\boldsymbol{\eta}_k^{s_j}$ is the prototype of k-th class computed using samples from $\mathcal{D}_l$, i.e.,

$$\boldsymbol{\eta}_k^{s_j} = \frac{\sum_{\boldsymbol{x}_i^t \in \mathcal{D}_l} \boldsymbol{\phi}_i^{s_j} \cdot \mathbb{1}\{\tilde{y}_i^t = k\}}{\sum_{\boldsymbol{x}_i^t \in \mathcal{D}_l} \mathbb{1}\{\tilde{y}_i^t = k\}}. \tag{8}$$

Assume the two labeling criterion are independent with each other, the probability that they agree on the same pseudo-label k is the product of these two mixture distributions, i.e., $p_k(\boldsymbol{x}_i^t) = \left(\sum_{j=1}^m \boldsymbol{w}_j^* h_k^{s_j}(\boldsymbol{x}_i^t)\right) \cdot \left(\sum_{j=1}^m \boldsymbol{w}_j^* q_k^{s_j}(\boldsymbol{x}_i^t)\right)$, and we can generate the pseudo-label accordingly, i.e.,

$$\tilde{y}_i^t = \arg\max_k p_k(\boldsymbol{x}_i^t). \tag{9}$$

Confidence Query Strategy The quantity $p_k(\boldsymbol{x}_i^t)$ can be interpreted as the confidence score of class k. The high confidence score implies the generated pseudo-label is more likely to be correct. Thus, we initialize the labeled subset $\mathcal{D}_l$ with N data with the highest confidence score and denote the remaining unlabeled subset as $\mathcal{D}_u$. Formally, we partition $\mathcal{D}^t$ into $\mathcal{D}_l$ and $\mathcal{D}_u$ based on the following selection rule,

$$\begin{cases} (\boldsymbol{x}_i^t, \tilde{y}_i^t) \in \mathcal{D}_l, & \text{if } \max_k p_k(\boldsymbol{x}_i^t) \geq \alpha_N, \\ \boldsymbol{x}_i^t \in \mathcal{D}_u, & \text{Otherwise.} \end{cases} \tag{10}$$

Here, α_N denotes the N-th largest confidence score. As shown in Algorithm 1, we query more confident data into $\mathcal{D}_l$ by increasing N at each iteration of training until $N = n$. More details about the confidence query strategy can be found in Appendix B.3.

5.3. Self-Training with Data Augmentation

The labeled subset $\mathcal{D}_l$ enables us to train the target model in a self-training manner. Notice that the variance term in Theorem 4.1 depends on the number of data in $\mathcal{D}_l$, i.e., $n_l = |\mathcal{X}_{\mathcal{D}_l}|$. Besides querying more pseudo-labeled data as mentioned above, we can further reduce the variance by leveraging data augmentation technique mixup (Zhang et al., 2017) to enlarge the subset $\mathcal{D}_l$, i.e., we construct an augmented dataset $\tilde{\mathcal{D}}_l$,

$$\begin{aligned} \tilde{\mathcal{D}}_l = &\mathcal{D}_l \cup \{(\boldsymbol{x}_{\text{aug}}^t, \tilde{y}_{\text{aug}}^t) \,|\boldsymbol{x}_{\text{aug}}^t = \lambda \boldsymbol{x}_i^t + (1-\lambda)\boldsymbol{x}_j^t, \\ &\tilde{y}_{\text{aug}}^t = \lambda \tilde{y}_i^t + (1-\lambda)\tilde{y}_j^t \;; (\boldsymbol{x}_i^t, \tilde{y}_i^t), (\boldsymbol{x}_j^t, \tilde{y}_j^t) \in \mathcal{D}_l\}, \end{aligned} \tag{11}$$

Algorithm 1 Selective Self-training for MSFDA

input pretrained source models $\{\boldsymbol{h}^{s_j} = \boldsymbol{g}^{s_j} \circ \boldsymbol{f}^{s_j}\}_{j=1}^m$, target domain unlabeled data $\mathcal{D}^t = \{\boldsymbol{x}_i^t\}_{i=1}^n$, and maximum iterations T

Initialize the domain aggregation weights $\boldsymbol{w}_j^0$ by optimizing (6) using pretrained source models $\{\boldsymbol{h}^{s_j}\}_{j=1}^m$.

Initialize the pseudo-label $\tilde{y}_i^t$ for each target data $\boldsymbol{x}_i^t$ by (9) using pretrained models $\{\boldsymbol{h}^{s_j}\}_{j=1}^m$ and initial domain aggregation weights $\boldsymbol{w}_j^0$.

for $\tau = 1, 2, \ldots, T$ **do**

- Set $N = N_0 + \frac{(n-N_0)\tau}{T}$.
- Update $\mathcal{D}_l$ and $\mathcal{D}_u$ by (10), and construct the augmented set $\tilde{\mathcal{D}}_l$ by (11).
- Update joint discriminators $\boldsymbol{d}^{(\tau)}$ with (13).
- Update feature extractor $\{\boldsymbol{f}^{s_j^{(\tau)}}\}_{j=1}^m$ with (14).
- Update the domain aggregation weights $\boldsymbol{w}_j^\tau$ by optimizing (6) using updated models $\{\boldsymbol{h}^{s_j^{(\tau)}}\}_{j=1}^m$.
- Update the pseudo-label $\tilde{y}_i^t$ using domain aggregation weights $\boldsymbol{w}_j^\tau$ and updated models $\{\boldsymbol{h}^{s_j^{(\tau)}}\}_{j=1}^m$ by (9).

end

output The final mixture models $\sum_{j=1}^m \boldsymbol{w}_j^T \boldsymbol{h}^{s_j^{(T)}}(\cdot)$.

where $\lambda \in [0, 1]$ is the mixup ratio. Since we cannot access the source training data, we fix the source model classifier $\boldsymbol{g}^{s_j}$ and update each feature extractor $\boldsymbol{f}^{s_j}$ to adapt the target domain, which implicitly incorporates the information from the source domain. Therefore, the cross-entropy loss for each feature extractor $\boldsymbol{f}^{s_j}$ is given by

$$\mathcal{L}_{\text{ce}}\left(\boldsymbol{f}^{s_j}, \tilde{\mathcal{D}}_l\right) = -\frac{1}{n_l} \sum_{\boldsymbol{x}_i^t, \tilde{y}_i^t \in \tilde{\mathcal{D}}_l} \sum_{k=1}^K \mathbb{1}\{\tilde{y}_i^t = k\} \log h_k^{s_j}(\boldsymbol{x}_i^t).$$

Moreover, motivated by the same intuition of learning domain aggregation weights in 5.1, we also utilize the information maximization loss to encourage the source models to make individual certain but globally diverse predictions, i.e.,

$$\mathcal{L}_{\text{IM}}\left(\boldsymbol{f}^{s_j}, \tilde{\mathcal{D}}_l\right) = \frac{1}{n_l} \sum_{\boldsymbol{x}_i^t \in \tilde{\mathcal{D}}_l} \mathrm{H}\left(\boldsymbol{h}^{s_j}(\boldsymbol{x}_i^t)\right) - \mathrm{H}\left(\overline{\boldsymbol{p}}^{s_j}\right), \tag{12}$$

where $\overline{p}_k^{s_j} \triangleq \frac{1}{n_l} \sum_{\boldsymbol{x}_i^t \in \tilde{\mathcal{D}}_l} h_k^{s_j}(\boldsymbol{x}_i^t)$. Different from previous works (Liang et al., 2020; Ahmed et al., 2021), we only apply information maximization loss on $\tilde{\mathcal{D}}_l$ to avoid over-confident but wrong pseudo labels in $\mathcal{D}_u$. Thus, the self-training objectives on $\tilde{\mathcal{D}}_l$ is given as $\mathcal{L}_{\tilde{\mathcal{D}}_l} = \mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{IM}}$.

5.4. Joint Adversarial Feature Alignment

As discussed in 4.3, the other essential component of the proposed method is the joint feature alignment. Denote the sub-

set of data not selected for pseudo-labeling as $\mathcal{D}_u = \mathcal{D}_t \setminus \mathcal{D}_l$, then we can align the distribution $P^{\mathcal{D}_l}_{\Phi_{1:m}}$ and $P^{t}_{\Phi_{1:m}}$ by enforcing feature alignment between $\mathcal{D}_l$ and $\mathcal{D}_u$. Intuitively, similar to traditional UDA where the source and target domain have different distributions, we can treat $\mathcal{D}_l$ as the labeled "source" data and $\mathcal{D}_u$ as the unlabeled "target" data, and enforce the joint feature alignment between $\mathcal{D}_l$ and $\mathcal{D}_u$.

Specifically, we adopt the adversarial training strategy proposed in (Ganin & Lempitsky, 2015; Xu et al., 2018) to perform feature alignment. First, we combine the features extracted from each source model to construct a joint feature representation, i.e., $\{\boldsymbol{f}^{s_j}(\boldsymbol{x}_i^t)\}_{j=1}^m$. Then, we train a separate neural network $\boldsymbol{d} : \mathcal{F}_{1:m} \to [0, 1]$ as the joint discriminator to distinguish the joint features of $\tilde{\mathcal{D}}_l$ and $\mathcal{D}_u$, and update multiple feature extractor $\boldsymbol{f}^{s_j}$ together to fool the single discriminator. Formally, the joint feature alignment loss is given by the following adversarial loss,

$$\begin{aligned}\mathcal{L}_{\text{adv}}\Big(\{\boldsymbol{f}^{s_j}\}_{j=1}^m, \boldsymbol{d}, \tilde{\mathcal{D}}_l, \mathcal{D}_u\Big) \\ = \frac{1}{n_l} \sum_{\boldsymbol{x}_i^t \in \tilde{\mathcal{D}}_l} \ln \boldsymbol{d}(\{\boldsymbol{f}^{s_j}(\boldsymbol{x}_i^t)\}_{j=1}^m) \\ + \frac{1}{n - n_l} \sum_{\boldsymbol{x}_i^t \in \mathcal{D}_u} \ln\big(1 - \boldsymbol{d}(\{\boldsymbol{f}^{s_j}(\boldsymbol{x}_i^t)\}_{j=1}^m)\big).\end{aligned} \tag{13}$$

In summary, we fix the classifier $\boldsymbol{g}^{s_j}$ for each source model, and alternating training the joint discriminator $\boldsymbol{d}$ to maximize $\mathcal{L}_{\text{adv}}$ and training each feature extractor $\boldsymbol{f}^{s_j}$ to minimize the combined loss, i.e.,

$$\begin{aligned}&\max_{\boldsymbol{d}} \mathcal{L}_{\text{adv}}\Big(\{\boldsymbol{f}^{s_j}\}_{j=1}^m, \boldsymbol{d}, \tilde{\mathcal{D}}_l, \mathcal{D}_u\Big) \\ &\min_{\{\boldsymbol{f}^{s_j}\}_{j=1}^m} \sum_{j=1}^m \lambda_{\text{ce}} \mathcal{L}_{\text{ce}}\Big(\boldsymbol{f}^{s_j}, \tilde{\mathcal{D}}_l\Big) + \lambda_{\text{IM}} \mathcal{L}_{\text{IM}}\Big(\boldsymbol{f}^{s_j}, \tilde{\mathcal{D}}_l\Big) \\ &+ \lambda_{\text{adv}} \mathcal{L}_{\text{adv}}\Big(\{\boldsymbol{f}^{s_j}\}_{j=1}^m, \boldsymbol{d}, \tilde{\mathcal{D}}_l, \mathcal{D}_u, \Big),\end{aligned} \tag{14}$$

where λ_{ce}, λ_{IM} and λ_{adv} are hyper-parameters that balance different regularization terms.

5.5. Algorithm

The overall algorithm is shown in Algorithm 1. All the models are retrained using the updated pseudo-labels at each iteration. As the performance of each model increases, we expect to see more correctly labeled samples in $\tilde{\mathcal{D}}_l$. When the algorithm converges, the final prediction of the target data $\boldsymbol{x}_i^t$ is obtained by the mixture models.

6. Experiment Results

In this section, we describe the experiment settings in 6.1, present the performance with selective oracle in 6.2 and the results with surrogate selection rule in 6.3, and discuss our takeaways in 6.4. Additional experiment results and implementation details can be found in Appendix B.

6.1. Settings

Datasets We conduct extensive evaluations of our methods using the following four benchmark datasets: **Digits-Five** (Peng et al., 2019) contains five different domains, including MNIST (MN), SVHN (SV), USPS (US), MNIST-M (MM), and Synthetic Digits (SY). **Office-31** (Saenko et al., 2010) contains 31 categories collected from three different office environments, including Amazon(A), Webcam (W), and DSLR (D). **Office-Home** (Venkateswara et al., 2017) is a more challenging dataset with 65 categories collected from four different office environments, including Art (A), Clipart (C), Real-world (R), and Product (P). **DomainNet** (Peng et al., 2019) is so far the largest and most challenging domain adaptation benchmark, which contains about 0.6 million images with 345 categories collected from six different domains, including Clipart (C), Infograph (I), Painting (P), Quickdraw (Q), Real (R), and Sketch (S).

Baselines To demonstrate the solid empirical performance of our method, we mainly compare it with the recently proposed SOTA multi-source-free domain adaptation method DECISION (Ahmed et al., 2021), CAiDA (Dong et al., 2021) and NRC (Yang et al., 2021a). We provide another baseline by evaluating ensemble prediction accuracy on the target domain using the pretrained source models, and denote it as Source-ens. In addition, we also include several SOTA single-source-free domain adaptation methods, including BAIT (Yang et al., 2020), SFDA (Kim et al., 2020), SHOT (Liang et al., 2020), and MA (Li et al., 2020). These single-source-free methods also do not require access to the source data, and we compare their multi-source ensemble results by taking the average of predictions from the multiple retrained source models after adaptation.

6.2. Results with Selective Oracle

In this experiment, we have access to the selective oracle in Definition 1 for $\mathcal{D}_l$ and unlabeled set $\mathcal{D}_u$ selection. The purpose here is to connect our proposed theorem to empirical performance by demonstrating the level of performance improvement that can be achieved for the MSFDA problem with a perfect subset selection. The results on Digits-Five, Office-31, Office-Home, and DomainNet are shown in the corresponding Tables, denoted as "Ours (Selective Oracle)". Across all the datasets, it can be observed that our method with selective oracle achieves a considerable performance gain, including more than 10% gain on challenging datasets Office-Home and DomainNet. Although these empirical results can only be achieved with the selective oracle, it validates our theoretical analysis and the significance of balancing the bias-variance trade-off in the MSFDA problem.

Table 1. **Results on Digit-Five (5 domains):** MN, SV, US, MM, and SY stand for MNIST, SVHN, USPS, MNIST-M, and Synthetic Digits datasets, respectively. Source-ens denotes the ensemble prediction of multiple pretrained source models.

Setting	Method	MN	SV	US	MM	SY	Avg
Single-source	BAIT (Yang et al., 2020)	96.2	60.6	96.7	87.6	90.5	86.3
	SFDA (Kim et al., 2020)	95.4	57.4	95.8	86.2	84.8	83.9
	SHOT (Liang et al., 2020)	98.9	58.3	97.7	90.4	83.9	85.8
	MA (Li et al., 2020)	98.4	59.1	98.0	90.8	84.5	86.2
Multi-source	Source-ens	96.7	76.8	93.8	66.7	77.6	82.3
	DECISION (Ahmed et al., 2021)	99.2	82.6	97.8	93.0	97.5	94.0
	CAiDA (Dong et al., 2021)	99.1	83.3	**98.6**	93.7	98.1	94.6
	Ours (Surrogate)	**99.2**	**90.7**	98.4	**97.4**	**98.4**	**96.8**
	Ours (Selective Oracle)	99.4	93.8	99.0	98.4	99.5	98.0

Table 2. **Results on Office-31 (3 domains):** A, D, and W stand for Amazon, DSLR, and Webcam datasets, respectively.

Setting	Method	D, W→ A	A, D→ W	A, W→ D	Avg
Single-source	BAIT (Yang et al., 2020)	71.1	98.5	98.8	89.5
	SFDA (Kim et al., 2020)	73.2	93.8	96.7	87.9
	SHOT (Liang et al., 2020)	75.0	94.9	97.8	89.3
	MA (Li et al., 2020)	75.2	96.1	97.3	89.5
Multi-source	Source-ens	62.5	95.8	98.7	86.0
	DECISION (Ahmed et al., 2021)	75.4	98.4	99.6	91.1
	CAiDA (Dong et al., 2021)	75.8	**98.9**	99.8	91.6
	Ours (Surrogate)	**77.6**	98.7	**99.8**	**92.0**
	Ours (Selective Oracle)	85.5	99.2	100.0	94.9

Table 3. **Results on Office-Home (4 domains):** A, C, R, and P stand for Art, Clipart, Real-world, and Product datasets, respectively.

Setting	Method	C, R, P→ A	A, R, P→ C	A, C, P→ R	A, C, R→ P	Avg
Single-source	BAIT (Yang et al., 2020)	71.1	59.6	77.2	79.4	71.8
	SFDA (Kim et al., 2020)	69.3	57.5	76.8	79.1	70.7
	SHOT (Liang et al., 2020)	72.2	59.3	82.9	82.8	74.3
	MA (Li et al., 2020)	72.5	57.4	81.7	82.3	73.5
Multi-source	Source-ens	67.0	52.1	78.6	74.8	68.1
	DECISION (Ahmed et al., 2021)	74.5	59.4	83.6	84.4	75.5
	CAiDA (Dong et al., 2021)	75.2	60.5	84.2	84.7	76.2
	NRC (Yang et al., 2021a)	70.6	60.0	84.6	83.5	74.7
	Ours (Surrogate)	**75.6**	**62.8**	**84.8**	**85.3**	**77.1**
	Ours (Selective Oracle)	86.9	85.1	92.2	95.7	90.0

6.3. Results with Surrogate Selection Rule

In practice, as selective oracle is not available, we use the proposed surrogate selection rule discussed in 5.2. The results are denoted as "Ours (Surrogate)". On the most challenging datasets, Office-Home and DomainNet, our proposed method can still outperform all baseline methods in terms of average accuracy. The SOTA method (Dong et al., 2021) shows strong performance over other baselines, but our proposed method can still outperform it with significant improvements over several domain adaptation tasks.

Digits-Five Our method achieves a performance gain of 14.5% over the ensemble of pretrained source models on this dataset. In addition, for some challenging tasks where the source model ensemble performs poorly on the target domain, e.g., M-MNIST, our proposed method outperforms the SOTA method by a large margin of 3.7%.

Office-31 It can be seen from Table 2 that most baseline methods perform very well on the tasks A, W $\rightarrow$ D and A, D $\rightarrow$ W, which implies that Domain D and W are similar. Moreover, our proposed method can still exhibit a further improvement on the D, W $\rightarrow$ A task.

Office-Home This large dataset is more challenging than the Office-31 dataset. For the most difficult task of this dataset: A, R, P $\rightarrow$ C, our approach significantly outperforms the SOTA method (Dong et al., 2021) by 2.3%.

DomainNet This is the most challenging domain adaptation benchmark so far. Our method still shows superior performance over all baseline methods.

Table 4. **Results on DomainNet (6 domains):** C, I, P, Q, R, and S stand for Clipart, Infograph, Painting, Quickdraw, Real, and Sketch, respectively.

Setting	Method	C	I	P	Q	R	S	Avg
Single-source	BAIT (Yang et al., 2020)	57.5	22.8	54.1	14.7	64.6	49.2	43.8
	SFDA (Kim et al., 2020)	57.2	23.6	55.1	16.4	65.5	47.3	44.2
	SHOT (Liang et al., 2020)	58.6	25.2	55.3	15.3	70.5	52.4	46.2
	MA (Li et al., 2020)	56.8	24.3	53.5	15.7	66.3	48.1	44.1
Multi-source	Source-ens	49.4	20.8	48.3	10.6	63.8	46.4	39.9
	DECISION (Ahmed et al., 2021)	61.5	21.6	54.6	18.9	67.5	51.0	45.9
	CAiDA (Dong et al., 2021)	63.6	20.7	54.3	19.3	**71.2**	51.6	46.8
	NRC (Yang et al., 2021a)	65.8	**24.1**	56.0	16.0	69.2	53.4	47.4
	Ours (Surrogate)	**66.5**	21.6	**56.7**	**20.4**	70.5	**54.4**	**48.4**
	Ours (Selective Oracle)	76.5	32.8	64.7	34.6	77.8	62.4	58.1

Table 5. **Ablation Study Results on Digit-Five**

Method	MN	SV	US	MM	SY	Avg
w/o denoise	99.1	90.2	98.0	97.4	98.2	96.6
w/o Domain Wights	98.9	88.7	98.0	97.1	95.4	95.6
IM All-Data	98.3	90.4	97.5	96.9	96.0	95.8
w/o alignment	99.1	90.4	98.4	97.3	98.4	96.7
w/o mixup	99.1	89.9	98.3	96.9	97.4	96.3
Ours (Surrogate)	**99.2**	**90.7**	**98.4**	**97.4**	**98.4**	**96.8**

Table 6. **Ablation Study Results on on Office-31**

Method	D, W→ A	A, D→ W	A, W→ D	Avg
w/o denoise	76.1	98.5	99.8	91.5
w/o Domain Wights	76.4	97.9	99.8	91.4
IM All-Data	74.4	98.0	99.4	90.6
w/o alignment	77.2	98.4	99.8	91.8
w/o mixup	76.6	**98.9**	99.8	91.8
Ours (Surrogate)	**77.6**	98.7	**99.8**	**92.0**

6.4. Ablation Study

In order to quantify the contributions of each technique discussed in Section 5: the learned domain weights, the pseudo-label denoising method, IM loss evaluated only on $\mathcal{D}_l$, the joint feature alignment loss $\mathcal{L}_{\text{adv}}$, and the mixup data augmentation, we take the Digit-Five and Office-31 datasets as examples to perform the ablation study. Specifically, we evaluate the performance of the following variants of our proposed method: (1) w/o Domain Weights, using uniform weights instead of our proposed domain weights. (2) w/o denoise, removing the pseudo-label denoising method. (3) IM All-Data, apply IM loss on all target domain data instead of only applying on $\mathcal{D}_l$. (4) w/o alignment, removing the joint feature alignment loss $\mathcal{L}_{\text{adv}}$. (5) w/o mixup, removing the data augmentation technique mixup, which will induce a larger variance. It can be observed from Table 5 and Table 6 that without each of these elements, the performance will degrade. These ablation study results further validate the effectiveness of each of these techniques to balance the bias and variance trade-off discovered by our theorem.

7. Concluding Remarks

Our study reveals the significance of balancing the bias-variance trade-off for the MSFDA problem through information-theoretic analysis, which also provides insights into the algorithm design. The empirical results obtained using a selective oracle assumption demonstrate the performance limit of the MSFDA problem, and it is noteworthy that there is still room for improvement in this source-free setting. As such, identifying a more effective surrogate selection rule remains a promising avenue for future research.

Acknowledgement

This work was supported, in part, by the MIT-IBM Watson AI Lab under Agreement No. W1771646, NSF under Grant No, CCF-1816209, and ARL and the USAF AI Accelerator under Cooperative Agreement No. FA8750-19-2-1000.

References

Ahmed, S. M., Raychaudhuri, D. S., Paul, S., Oymak, S., and Roy-Chowdhury, A. K. Unsupervised multi-source domain adaptation without access to source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10103–10112, 2021.

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

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

Cover, T. M. *Elements of information theory*. John Wiley & Sons, 1999.

Csiszár, I. Sanov property, generalized I-projection and a conditional limit theorem. *The Annals of Probability*, pp. 768–793, 1984.

Daniely, A., Sabato, S., Ben-David, S., and Shalev-Shwartz, S. Multiclass learnability and the ERM principle. In *Proceedings of the 24th Annual Conference on Learning Theory*, pp. 207–232. JMLR Workshop and Conference Proceedings, 2011.

Dong, J., Fang, Z., Liu, A., Sun, G., and Liu, T. Confident anchor-induced multi-source free domain adaptation. *Advances in Neural Information Processing Systems*, 34, 2021.

Dou, Q., Ouyang, C., Chen, C., Chen, H., and Heng, P.-A. Unsupervised cross-modality domain adaptation of ConvNets for biomedical image segmentations with adversarial loss. *arXiv preprint arXiv:1804.10916*, 2018.

Ganin, Y. and Lempitsky, V. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pp. 1180–1189. PMLR, 2015.

Guo, H., Pasunuru, R., and Bansal, M. Multi-source domain adaptation for text classification via distancenet-bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7830–7838, 2020.

Guo, J., Shah, D. J., and Barzilay, R. Multi-source domain adaptation with mixture of experts. *arXiv preprint arXiv:1809.02256*, 2018.

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

Hoffman, J., Mohri, M., and Zhang, N. Algorithms and theory for multiple-source adaptation. *arXiv preprint arXiv:1805.08727*, 2018.

Hu, W., Miyato, T., Tokui, S., Matsumoto, E., and Sugiyama, M. Learning discrete representations via information maximizing self-augmented training. In *International conference on machine learning*, pp. 1558–1567. PMLR, 2017.

Kim, Y., Cho, D., Han, K., Panda, P., and Hong, S. Domain adaptation without source data. *arXiv preprint arXiv:2007.01524*, 2020.

Krause, A., Perona, P., and Gomes, R. Discriminative clustering by regularized information maximization. *Advances in neural information processing systems*, 23, 2010.

Kundu, J. N., Kulkarni, A. R., Bhambri, S., Mehta, D., Kulkarni, S. A., Jampani, V., and Radhakrishnan, V. B. Balancing discriminability and transferability for source-free domain adaptation. In *International Conference on Machine Learning*, pp. 11710–11728. PMLR, 2022.

Kurmi, V. K., Subramanian, V. K., and Namboodiri, V. P. Domain impression: A source data free domain adaptation method. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 615–625, 2021.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Lee, D.-H. et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, pp. 896, 2013.

Li, R., Jiao, Q., Cao, W., Wong, H.-S., and Wu, S. Model adaptation: Unsupervised domain adaptation without source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9641–9650, 2020.

Li, Y., Murias, M., Major, S., Dawson, G., and Carlson, D. E. Extracting relationships by multi-domain matching. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 6799–6810, 2018.

Liang, J., Hu, D., and Feng, J. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pp. 6028–6039. PMLR, 2020.

Long, M., Cao, Y., Wang, J., and Jordan, M. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pp. 97–105. PMLR, 2015.

Long, M., Cao, Y., Cao, Z., Wang, J., and Jordan, M. I. Transferable representation learning with deep adaptation networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(12):3071–3085, 2018.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation with multiple sources. *Advances in neural information processing systems*, 21, 2008.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1406–1415, 2019.

Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. D. *Dataset shift in machine learning*. Mit Press, 2008.

Rizve, M. N., Duarte, K., Rawat, Y. S., and Shah, M. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv preprint arXiv:2101.06329*, 2021.

Saenko, K., Kulis, B., Fritz, M., and Darrell, T. Adapting visual category models to new domains. In *European conference on computer vision*, pp. 213–226. Springer, 2010.

Shi, W., Gong, Y., Ding, C., Tao, Z. M., and Zheng, N. Transductive semi-supervised deep learning using min-max features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 299–315, 2018.

Truong, N., Sun, K., Wang, S., Guitton, F., and Guo, Y. Privacy preservation in federated learning: An insightful survey from the GDPR perspective. *Computers & Security*, 110:102402, 2021.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7167–7176, 2017.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5018–5027, 2017.

Wang, H., Yang, W., Lin, Z., and Yu, Y. Tmda: Task-specific multi-source domain adaptation via clustering embedded adversarial training. In *2019 IEEE International Conference on Data Mining (ICDM)*, pp. 1372–1377. IEEE, 2019.

Wang, H., Xu, M., Ni, B., and Zhang, W. Learning to combine: Knowledge aggregation for multi-source domain adaptation. In *European Conference on Computer Vision*, pp. 727–744. Springer, 2020.

Xia, H., Zhao, H., and Ding, Z. Adaptive adversarial network for source-free domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9010–9019, 2021.

Xu, R., Chen, Z., Zuo, W., Yan, J., and Lin, L. Deep cocktail network: Multi-source unsupervised domain adaptation with category shift. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3964–3973, 2018.

Yang, S., Wang, Y., van de Weijer, J., Herranz, L., and Jui, S. Unsupervised domain adaptation without source data by casting a bait. *arXiv preprint arXiv:2010.12427*, 2020.

Yang, S., van de Weijer, J., Herranz, L., Jui, S., et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. *Advances in Neural Information Processing Systems*, 34:29393–29405, 2021a.

Yang, S., Wang, Y., van de Weijer, J., Herranz, L., and Jui, S. Generalized source-free domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8978–8987, 2021b.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Zhang, P., Zhang, B., Zhang, T., Chen, D., Wang, Y., and Wen, F. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12414–12424, 2021.

Zhao, H., Zhang, S., Wu, G., Moura, J. M., Costeira, J. P., and Gordon, G. J. Adversarial multiple source domain adaptation. *Advances in neural information processing systems*, 31:8559–8570, 2018.

Zhao, S., Wang, G., Zhang, S., Gu, Y., Li, Y., Song, Z., Xu, P., Hu, R., Chai, H., and Keutzer, K. Multi-source distilling domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 12975–12983, 2020.

Zhu, Y., Zhuang, F., and Wang, D. Aligning domain-specific distribution and classifier for cross-domain classification

from multiple sources. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5989–5996, 2019.

A. Derivations & Proofs

A.1. Proof of Theorem 4.1

The gap between the empirical risk $\mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l)$ over $\mathcal{D}_l$ and $\mathcal{L}_P(\boldsymbol{h}, P^t_{XY})$ can be written as

$$\begin{aligned}
&\mathcal{L}_P(\boldsymbol{h}, P^t_{XY}) - \mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l) \\
&= \mathcal{L}_P(\boldsymbol{h}, P^t_{XY}) - \mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{XY}) + \mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{XY}) - \mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l) \\
&= \mathcal{L}_P(\boldsymbol{h}, P^t_{\Phi_{1:m},Y}) - \mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{\Phi_{1:m},Y}) + \mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{\Phi_{1:m},Y}) - \mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l) \\
&\leq |\mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{\Phi_{1:m},Y}) - \mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l)| + |\mathcal{L}_P(\boldsymbol{h}, P^t_{\Phi_{1:m},Y}) - \mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{\Phi_{1:m},Y})|,
\end{aligned} \tag{15}$$

where $\mathcal{L}_P(\boldsymbol{h}, P^t_{\Phi_{1:m},Y}) \triangleq \mathbb{E}_{P^t_{\Phi_{1:m},Y}}[\ell(\boldsymbol{h}(\Phi_{1:m}), Y)]$ explicitly represents $\mathcal{L}_P(\boldsymbol{h}, P^t_{XY})$ using $\Phi_{1:m}$. We note that the first term is simply the generalization error of supervised learning using n_l i.i.d. samples generated from the distribution $P^{\mathcal{D}_l}_{\Phi_{1:m},Y}$. Since $\boldsymbol{h} \in \mathcal{H}$ has finite Natarajan dimension, by Natarajan dimension theory (see (Daniely et al., 2011) equation (6)), the following upper bound holds for some constant $C > 0$ with probability at least $1 - \delta$,

$$|\mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{\Phi_{1:m},Y}) - \mathcal{L}_E(\boldsymbol{h}, \mathcal{D}_l)| \leq C\sqrt{\frac{d_N(\mathcal{H})\log K + \log\frac{1}{\delta}}{n_l}}, \tag{16}$$

where K is the number of different classes for the label.

As for the second term in (15), it can be upper bounded via the Donsker-Varadhan variational representation of the relative entropy between two probability measures P and Q defined on $\mathcal{X}$:

$$D(P\|Q) = \sup_{g \in \mathcal{G}} \left\{ \mathbb{E}_P[g(X)] - \log \mathbb{E}_Q[e^{g(X)}] \right\}, \tag{17}$$

where the supremum is over all measurable functions $\mathcal{G} = \{g : \mathcal{X} \to \mathbb{R}, \text{ s.t. } \mathbb{E}_Q[e^{g(X)}] < \infty\}$. It then follows that for any $\lambda \in \mathbb{R}$,

$$D(P^{\mathcal{D}_l}_{\Phi_{1:m},Y} \| P^t_{\Phi_{1:m},Y}) \geq \mathbb{E}_{P^{\mathcal{D}_l}_{\Phi_{1:m},Y}}[\lambda\ell(\boldsymbol{h}(\Phi_{1:m}), Y)] - \log \mathbb{E}_{P^t_{\Phi_{1:m},Y}}[e^{\lambda\ell(\boldsymbol{h}(\Phi_{1:m}),Y)}]. \tag{18}$$

Since the loss function ℓ is bounded between $[0, M]$, we can show that $\ell(\boldsymbol{h}(\Phi_{1:m}), Y)$ is $\frac{M}{2}$-sub-Gaussian, i.e.,

$$\log \mathbb{E}_{P^t_{\Phi_{1:m},Y}}\left[e^{\lambda(\ell(\boldsymbol{h}(\Phi_{1:m}),Y) - \mathbb{E}_{P^t_{\Phi_{1:m},Y}}[\ell(\boldsymbol{h}(\Phi_{1:m}),Y)])} \right] \leq \frac{M^2\lambda^2}{8}. \tag{19}$$

Thus, the following inequality holds for all $\lambda \in \mathbb{R}$,

$$\begin{aligned}
&D(P^{\mathcal{D}_l}_{\Phi_{1:m},Y} \| P^t_{\Phi_{1:m},Y}) \\
&\geq \mathbb{E}_{P^{\mathcal{D}_l}_{\Phi_{1:m},Y}}[\lambda\ell(\boldsymbol{h}(\Phi_{1:m}), Y)] - \log \mathbb{E}_{P^t_{\Phi_{1:m},Y}}[e^{\lambda\ell(\boldsymbol{h}(\Phi_{1:m}),Y)}] \\
&\geq \lambda\left(\mathbb{E}_{P^{\mathcal{D}_l}_{\Phi_{1:m},Y}}[\ell(\boldsymbol{h}(\Phi_{1:m}), Y)] - \mathbb{E}_{P^t_{\Phi_{1:m},Y}}[\ell(\boldsymbol{h}(\Phi_{1:m}), Y)] \right) - \frac{M^2\lambda^2}{8} \\
&= \lambda\left(\mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{\Phi_{1:m},Y}) - \mathcal{L}_P(\boldsymbol{h}, P^t_{\Phi_{1:m},Y}) \right) - \frac{M^2\lambda^2}{8},
\end{aligned} \tag{20}$$

which gives a non-negative parabola in λ, whose discriminant must be non-positive, which implies

$$\begin{aligned}
\left|\mathcal{L}_P(\boldsymbol{h}, P^t_{\Phi_{1:m},Y}) - \mathcal{L}_P(\boldsymbol{h}, P^{\mathcal{D}_l}_{\Phi_{1:m},Y})\right| &\leq \sqrt{\frac{M^2}{2} D(P^{\mathcal{D}_l}_{\Phi_{1:m},Y} \| P^t_{\Phi_{1:m},Y})} \\
&= \frac{M}{\sqrt{2}}\sqrt{D(P^{\mathcal{D}_l}_{\Phi_{1:m}} \| P^t_{\Phi_{1:m}}) + D(P^{\mathcal{D}_l}_{Y|\Phi_{1:m}} \| P^t_{Y|\Phi_{1:m}} | P^{\mathcal{D}_l}_{\Phi_{1:m}})}.
\end{aligned} \tag{21}$$

Combining (21) with (16) completes the proof.

A.2. Proof of Proposition 4.1

Let $\mathcal{P}_w = \{q \mid q = \sum_{j=1}^m \boldsymbol{w}_j P_{Y|\Phi_j}^{s_j}, \ \sum_{j=1}^m \boldsymbol{w}_j = 1\}$ denotes the set of mixture distribution. It is easy to show set $\mathcal{P}_w$ is convex and closed. There are two cases, i.e., whether the target distribution lies within this convex set or not,

- If $P_{Y|\Phi_{j=1:m}}^t \in \mathcal{P}_w$, then there exists a $\boldsymbol{w}^*$ such that $D\left(\sum_{j=1}^m \boldsymbol{w}_j^* P_{Y|\Phi_j}^{s_j} \| P_{Y|\Phi_{j=1:m}}^t\right) = 0$, and this proposition is trivilly true.
- If $P_{Y|\Phi_{j=1:m}}^t \notin \mathcal{P}_w$, let $\sum_{j=1}^m \boldsymbol{w}_j^* P_{Y|\Phi_j}^{s_j}$ to be the I-projection of $P_{Y|\Phi_{j=1:m}}^t$ onto $\mathcal{P}_w$, then by the Pythagoras' theorem of information geometry (see (Cover, 1999) Theorem 12.6.1), we have

$$D\Big(P_{Y|\Phi_j}^{s_j} \| P_{Y|\Phi_{j=1:m}}^t\Big) \geq D\Big(P_{Y|\Phi_j}^{s_j} \| \sum_{j=1}^m \boldsymbol{w}_j^* P_{Y|\Phi_j}^{s_j}\Big) + D\Big(\sum_{j=1}^m \boldsymbol{w}_j^* P_{Y|\Phi_j}^{s_j} \| P_{Y|\Phi_{j=1:m}}^t\Big), \tag{22}$$

  as $P_{Y|\Phi_j}^{s_j} \in \mathcal{P}_w$. Thus, it follows that

$$D\Big(\sum_{j=1}^m \boldsymbol{w}_j^* P_{Y|\Phi_j}^{s_j} \| P_{Y|\Phi_{j=1:m}}^t\Big) \leq D\left(P_{Y|\Phi_j}^{s_j} \| P_{Y|\Phi_{j=1:m}}^t\right). \tag{23}$$

A.3. Proof of Remark 1

Following the chain rule of KL divergence, we have $D(P_{\Phi_{1:m}}^{\mathcal{D}_l} \| P_{\Phi_{1:m}}^t) = D(P_{\Phi_j}^{\mathcal{D}_l} \| P_{\Phi_j}^t) + D(P_{\Phi_{\{1:m\}\setminus j}|\Phi_j}^{\mathcal{D}_l} \| P_{\Phi_{\{1:m\}\setminus j}|\Phi_j}^t)$ holds for any $j = 1:m$, which implies $D(P_{\Phi_{1:m}}^{\mathcal{D}_l} \| P_{\Phi_{1:m}}^t) \geq D(P_{\Phi_j}^{\mathcal{D}_l} \| P_{\Phi_j}^t)$ holds for any j, following by the non-negativity of KL divergence. Therefore, we can conclude that $D(P_{\Phi_{1:m}}^{\mathcal{D}_l} \| P_{\Phi_{1:m}}^t) \geq \frac{1}{m} \sum_{j=1}^m D(P_{\Phi_j}^{\mathcal{D}_l} \| P_{\Phi_j}^t)$.

B. Additional Experiment Results

B.1. Visualization

To better understand why it is crucial to use the selective pseudo labeling technique and joint feature alignment to reduce the two bias terms, we provide t-SNE plots in feature space to visualize the difference between $\mathcal{D}_l$ and $\mathcal{D}_u$, and each of them is labeled in two ways: using ground-truth labels and using pseudo labels directly generated from source models. We take task D, W $\rightarrow$ A of the Office-31 dataset as our example and extract the target domain features from pretrained source models. The results are presented in Figure 2. From the figure, we can find that: (1) There are mismatches between pseudo labels of $\mathcal{D}_u$ and ground truth labels, i.e., the pseudo labels of $\mathcal{D}_u$ are much noisier than $\mathcal{D}_l$. Thus, we should only use data in D_l for training to reduce the first bias term. (2) The feature of samples in $\mathcal{D}_l$ are more separately clustered compared to $\mathcal{D}_u$, which implies the existence of a distribution shift between $\mathcal{D}_l$ and $\mathcal{D}_u$. To mitigate this discrepancy, we align the feature distributions for both $\mathcal{D}_l$ and $\mathcal{D}_u$ by minimizing the joint adversarial feature alignment loss $\mathcal{L}_{\text{adv}}$ in the proposed method.

B.2. Implementation Details

For a fair comparison, we follow the experiment settings in previous works (Ahmed et al., 2021; Liang et al., 2020). For the Digits-Five benchmark, we use a variant of LeNet (LeCun et al., 1998) as our pretrained model. We resize the image samples from different digit domains to the same size (32×32) and convert the gray-scale images to RGB. For all office benchmarks and DomainNet, we use the pretrained ResNet-50 (He et al., 2016) as the backbone of the feature extractor, followed by a bottleneck layer with batch normalization and a classifier layer with weight normalization. We train the models on different source domain datasets and then retrain the pretrained models on the remaining single target domain dataset. The weights of the classifier are frozen during model training. The maximum number of training iterations is set to 20. All experiments are implemented in PyTorch using Tesla V100 GPUs with 32 GB memory.

B.3. Details about Query Strategy

The query strategy in (10) relies on a confidence threshold α_N to select data in $\mathcal{D}_l$ and $\mathcal{D}_u$. α_N is defined as N-th largest confidence score among all the data's confidence score, where N is the number of data to be labeled. The initial number

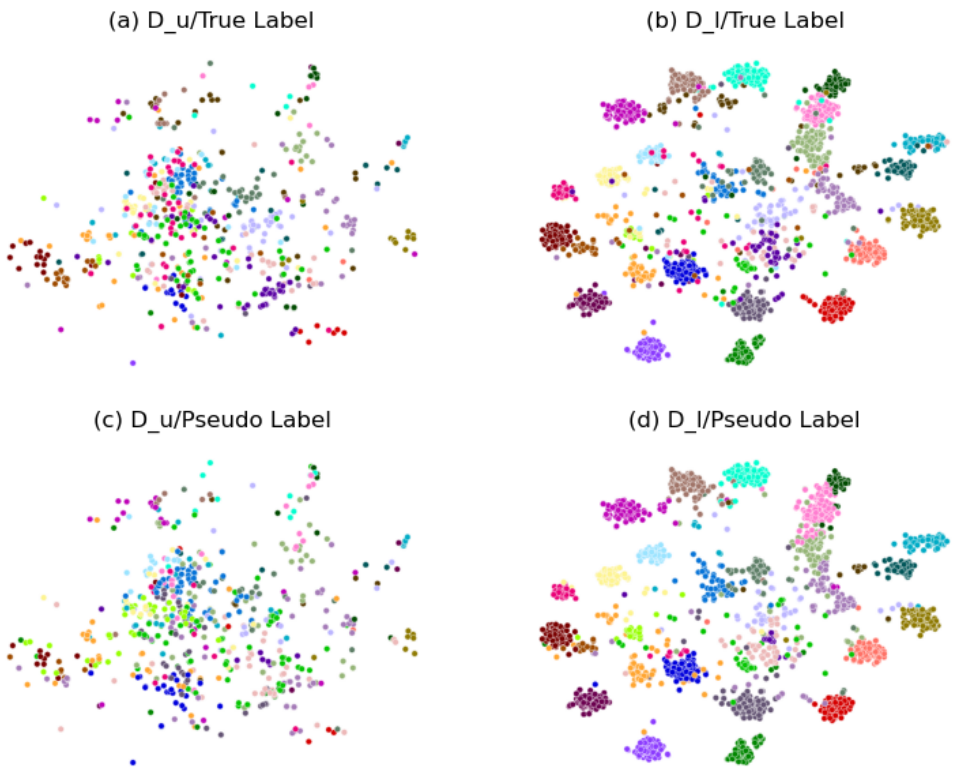

Figure 2. t-SNE visualization on D, W $\rightarrow$ A domain adaptation task of Office-31 dataset: (a) plot of $\mathcal{D}_u$ (705 data) labeled with ground-truth label. (b) plot of $\mathcal{D}_l$ (2112 data) labeled with the ground-truth label. (c) plot of $\mathcal{D}_u$ (705 data) labeled with pseudo-label. (d) plot of $\mathcal{D}_l$ (2112 data) labeled with pseudo-label.

Table 7. **The impact of λ_α on Office-31 (3 domains):** A, D, and W stand for Amazon, DSLR, and Webcam datasets, respectively.

Method	D,W$\rightarrow$ A	A,D$\rightarrow$ W	A,W$\rightarrow$ D	Avg
$\lambda_\alpha = 0.5$	77.8	98.2	99.8	91.9
$\lambda_\alpha = 0.6$	77.6	98.7	99.8	92.0
$\lambda_\alpha = 0.8$	76.4	98.4	100.0	91.6
$\lambda_\alpha = 1.0$	75.9	97.7	100.0	91.2

Table 8. **The impact of λ_α on Digit-Five (5 domains):** MN, SV, US, MM, and SY stand for MNIST, SVHN, USPS, MNIST-M, and Synthetic Digits datasets, respectively.

Method	MN	SV	US	MM	SY	Avg
$\lambda_\alpha = 0.5$	99.2	90.7	98.4	97.4	98.4	96.8
$\lambda_\alpha = 0.7$	99.1	90.4	98.6	97.3	98.4	96.8
$\lambda_\alpha = 1.0$	98.8	90.6	98.7	96.9	97.1	96.4

for labeled data N_0 is set to be the number of data whose confidence score is larger than an initial threshold, i.e., $\lambda_\alpha \cdot \overline{p}$, where $\overline{p}$ denotes the mean confidence score across all the data, and λ_α is a hyper-parameter. We conduct an ablation study to investigate the impact of λ_α using Office-31 and Digit-Five dataset, the results are showing in Table 7 and Table 8. Empirically, we do not observe significant variance of performance with different choices of λ_α. Intuitively, larger λ_α selects a smaller subset (variance dominates), and smaller λ_α selects more samples (bias dominates). A balance of bias and variance trade-off is crucial in achieving good performance.

B.4. Hyper-parameter

For model optimization, we use SGD optimizer with momentum value 0.9 and weight decay 10^{-3}, the learning rate for backbone, bottleneck layer, and classifier layer is set to 10^{-2}, 10^{-2} and 10^{-3}, respectively. For domain aggregation weights optimization, we also use SGD optimizer with learning rate 10^{-1} without weight decay. The other hyper-parameters are summarized in Table 9, where bs denotes the batch size, and itr denotes the training iteration.

Table 9. Hyper-parameter

Task	bs	itr	λ_{ce}	λ_{IM}	λ_{adv}	λ_{α}
Office-31 (all domains)	32	20	0.1	1.0	1.0	0.6
Digit-Five (all domains)	64	20	0.2	1.0	1.0	0.5
Office-Home (domain A & C)	32	10	0.2	1.0	1.0	0.7
Office-Home (domain R)	32	10	0.2	1.0	1.0	0.4
Office-Home (domain P)	32	10	0.2	1.0	1.0	0.3
DomainNet (domain C & P)	32	10	0.2	1.0	1.0	0.5
DomainNet (domain I, R, S)	32	10	0.2	1.0	1.0	0.7
DomainNet (domain Q)	32	10	0.2	1.0	1.0	1.0