Theoretical Performance Guarantees for Partial Domain Adaptation via Partial Optimal Transport

Jayadev Naram [1] **Fredrik Hellström** [2] **Ziming Wang** [1] **Rebecka Jörnsten** [1] **Giuseppe Durisi** [1]

Abstract

In many scenarios of practical interest, labeled data from a target distribution are scarce while labeled data from a related source distribution are abundant. One particular setting of interest arises when the target label space is a subset of the source label space, leading to the framework of partial domain adaptation (PDA). Typical approaches to PDA involve minimizing a domain alignment term and a weighted empirical loss on the source data, with the aim of transferring knowledge between domains. However, a theoretical basis for this procedure is lacking, and in particular, most existing weighting schemes are heuristic. In this work, we derive generalization bounds for the PDA problem based on partial optimal transport. These bounds corroborate the use of the partial Wasserstein distance as a domain alignment term, and lead to theoretically motivated explicit expressions for the empirical source loss weights. Inspired by these bounds, we devise a practical algorithm for PDA, termed WARMPOT. Through extensive numerical experiments, we show that WARMPOT is competitive with recent approaches, and that our proposed weights improve on existing schemes.

1. Introduction

In unsupervised domain adaptation, one has access to a set of labeled source data and a set of unlabeled target data, drawn from different but related distributions. The aim is to use these data sets to learn a predictor that performs well on new instances from the target data distribution (Redko et al., 2019; Farahani et al., 2021). In contemporary practice, it is common for classifiers that are pre-trained on large, diverse domains to be deployed on smaller domains, characterized by a smaller label space. This motivates the framework of *unsupervised partial domain adaptation* (PDA), wherein the target label space is a subset of the source label space (Cao et al., 2018).

In PDA, the use of labeled source data from outlier classes during training typically has an adverse effect on test performance—a phenomenon termed *negative transfer* (Cao et al., 2018). To alleviate this issue, several heuristic schemes to weight the source data during training have been proposed (Zhang et al., 2018; Liang et al., 2020; Li et al., 2020; Gu et al., 2021; 2024). However, a theoretical motivation is lacking for most weight selections.

In this work, we provide theoretically motivated algorithms for PDA. Specifically, we derive generalization bounds on the target population loss and devise training strategies that minimize them. The bounds that we obtain involve a partial Wasserstein distance between the empirical feature distributions for the source and target data (Caffarelli & McCann, 2010). This motivates the popular strategy of learning a feature map that aligns source and target features, followed by a predictor trained to classify the labeled source features. We will refer to objectives that are minimized with this aim as *domain alignment terms*. While the standard Wasserstein distance has been widely used to analyze and design algorithms for domain adaptation (Courty et al., 2014; 2017b), its partial counterpart is crucial to handle the existence of outliers (Wang et al., 2024). Additionally, our bounds include weighted source training losses, where, in contrast to all results available in the literature, the weights arise constructively from the partial optimal transport problem associated with the partial Wasserstein distance. This enables a principled weight selection for addressing negative transfer that, unlike the aforementioned heuristics, comes with a clear theoretical motivation in terms of transport plans.

Our bounds come in two flavors. First, similar to Shen et al. (2018), one bound depends on the partial Wasserstein distance between the empirical distributions of the source and target features. Second, building upon the work of Courty et al. (2017a), we obtain a bound that incorporates estimates of the unknown labels of the target samples. This yields a partial transport problem involving the joint empirical distri-

[1]Chalmers University of Technology, Gothenburg, Sweden [2]University College London, London, England. Correspondence to: Jayadev Naram <jayadev@chalmers.se>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

bution of features and labels. Each bound depends on two parameters, which intuitively correspond to the expected portion of outliers in the source and target data sets, respectively. Concretely, the parameters determine the proportion of mass from each of the data sets that is accounted for in the transport problem.

Inspired by the bounds, we propose a novel algorithm for PDA, termed *weighted and regularized minimizer via partial optimal transport* (WARMPOT), whose performance we compare against state-of-the-art (SOTA) methods.

Contributions. In this work, we derive two new families of generalization bounds for PDA, and devise algorithms to minimize them. In particular, our bounds:

- explicitly depend on the learned feature map, motivating, for the first time in the context of PDA, the approach of partly aligning feature distributions;
- yield explicit weights for source data points with a principled motivation, giving a theoretically grounded way to tackle negative transfer;
- lead to algorithms that improve upon or are comparable to recent approaches to PDA;
- give rise to weights that, when combined with the ARPM algorithm of Gu et al. (2024), lead to SOTA results for the Office-Home data set.

Furthermore, compared to previous bounds obtained for the more restrictive domain-adaptation problem, our proof techniques directly yield bounds that depend on the Wassserstein distance between the empirical distributions of source and target features. In contrast, in existing bounds, such as the ones proposed by Courty et al. (2017a) and Shen et al. (2018), the Wasserstein distance involves the actual source and target distributions, and an additional concentration-of-measure step is required to express such bounds in terms of numerically computable empirical distributions.

2. Related Work

The problem of unsupervised domain adaptation was first formalized and analyzed by Ben-David et al. (2006). They derived a generalization bound in terms of the so called $\mathcal{H}$-divergence, defined in terms of a hypothesis class $\mathcal{H}$. This divergence is bounded and can be efficiently estimated if the VC dimension of $\mathcal{H}$ is finite (Ben-David et al., 2010). Motivated by this bound, Ganin et al. (2016) proposed domain-adversarial training, wherein an approximation of the $\mathcal{H}$-divergence is used as a domain alignment term. However, the worst-case nature of the VC dimension leads to bounds that are too weak to explain generalization in deep neural networks (Nagarajan & Kolter, 2019; Zhang et al., 2021).

In order to exploit the geometry of the data distributions, Courty et al. (2014; 2017b) proposed the use of optimal transport, and specifically the Wasserstein distance, for domain adaptation. This approach was theoretically supported by Redko et al. (2017) and Shen et al. (2018), who derived bounds in terms of the Wasserstein distance between the source and target input distributions. Notably, this alleviates the issues of uniform convergence associated with the $\mathcal{H}$-divergence. Based on these bounds, Shen et al. (2018) proposed a domain alignment term, computed using a Wasserstein generative adversarial network (GAN) operating on empirical feature distributions (Arjovsky et al., 2017). Courty et al. (2017a) derived a bound in terms of the joint source and target instance-label distributions, where estimates appear in place of the unknown target labels. Damodaran et al. (2018) drew inspiration from this bound to devise an algorithm using mini-batch optimal transport on the joint feature-label distribution. However, it is worth noting that the bounds reviewed so far all depend on the instance distributions, and do not incorporate the learned feature map. Hence, they do not fully motivate the typical practice of computing the Wasserstein distance between the empirical distributions of source and target features.

As mentioned, an important factor in solving the PDA problem is to appropriately weight the source data. Cao et al. (2018) proposed to use heuristic class-level weights based on the predictions on unlabeled target inputs. Li et al. (2020) used these class-level weights, along with a maximum mean discrepancy loss as the domain alignment term. In addition to domain-adversarial training, Zhang et al. (2018) and Cao et al. (2019) determined the weights based on how well a domain discriminator can predict whether a given input is from the source or target distribution. Liang et al. (2020) proposed entropy-aware weights, along with an advanced alignment strategy, while Gu et al. (2021; 2024) computed weights by minimizing the Wasserstein distance between a weighted source feature distribution and the target feature distribution. Nguyen et al. (2022) used the joint distribution partial Wasserstein distance as the domain alignment term, but with uniformly weighted source data. Wang et al. (2024) proposed a partial Wasserstein-GAN, an extension of Wasserstein-GAN to PDA (Wang et al., 2022). Specifically, they considered a class-level weighting scheme with the 1-partial Wasserstein distance as domain alignment term. Li & Chen (2022) derived a generalization bound for PDA in terms of model smoothness, and proposed to focus on smoothness rather than alignment to transfer knowledge between domains. Fatras et al. (2021) use a mini-batch joint distribution unbalanced optimal transport (UOT) cost as domain alignment term, along with uniformly weighted source data. Chang et al. (2022) consider a more general setting where the proposed algorithm uses the marginals of the UOT transport plan to compute binary weights for target samples.

Generalization bounds for domain adaptation containing a weighted source loss term are reported in the works of Tachet des Combes et al. (2020) and Luo & Ren (2024). These bounds do not use a generalized Wasserstein metric, such as the partial Wasserstein metric, as domain alignment term. Furthermore, the bounds rely on class-level weights defined in terms of unknown data distributions. These weights are then estimated using a method developed by Lipton et al. (2018). However, these estimates are only guaranteed to be accurate if the so called generalized label shift assumption (Tachet des Combes et al., 2020) holds exactly, i.e., if the feature representation $Z = f(X)$ of the instance X with label Y is such that $P(Z|Y = y) = Q(Z|Y = y)$ for source distribution P and target distribution Q.

In this work, similar to Nguyen et al. (2022); Wang et al. (2024), we use the partial Wasserstein distance as the domain alignment term. However, unlike prior work, we derive generalization bounds that both corroborate the choice of domain alignment term and lead to a theoretically motivated weighting scheme for the empirical source loss. Furthermore, unlike some of the empirical weighting methods used in the literature (e.g., class-level weights), our approach readily extends beyond classification settings.

3. Theoretical Results

We now present our main theoretical results. First, in Section 3.1, we formalize the problem setup and introduce the notation. In Section 3.2, we obtain bounds on the empirical target loss for a fixed sample, which we leverage to obtain generalization bounds in Section 3.3. Some useful definitions and results are recalled in Appendix A.

3.1. Problem Setup and Proposed Approach

We next introduce the notation that we will use throughout this section. We let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be the source domain, where $\mathcal{X} \subseteq \mathbb{R}^d$ is the input space, equipped with the sigma-algebra Σ_X, and $\mathcal{Y} \subseteq \mathbb{R}$ is the source label space, equipped with the sigma-algebra Σ_Y. We consider a joint probability distribution P_Z on $(\mathcal{Z}, \Sigma_X \otimes \Sigma_Y)$, called the *source distribution*. Similarly, we let $\tilde{\mathcal{Z}} = \mathcal{X} \times \tilde{\mathcal{Y}}$ be the target domain, where $\tilde{\mathcal{Y}}$, equipped with the sigma-algebra $\Sigma_{\tilde{Y}}$, is an unknown subset of $\mathcal{Y}$. Furthermore, we introduce a second joint probability distribution $Q_{\tilde{Z}}$ on $(\tilde{\mathcal{Z}}, \Sigma_X \otimes \Sigma_{\tilde{Y}})$, termed the *target distribution*.[1]

A hypothesis is a measurable function $w : \mathcal{X} \to \mathcal{Y}$. In order to discuss feature alignment, we express each hypothesis as $w = g \circ f$, where f is a feature extractor and g is a classifier. Throughout the paper, we will for simplicity consider bounded loss functions $\ell : \mathcal{Y} \times \mathcal{Y} \to [0, 1]$. We are interested in determining a hypothesis w within a suitably chosen hypothesis class $\mathcal{W}$ (to be introduced later) that minimizes the population target loss

$$L_{Q_{\tilde{Z}}}(w) = \mathbb{E}_{(X,Y)\sim Q_{\tilde{Z}}}[\ell(w(X), Y)]. \tag{1}$$

To do so, in the PDA setup considered in this paper, we have at our disposal a vector $\boldsymbol{z} = (z_1, \ldots, z_{n_s}) \in \mathcal{Z}^{n_s}$, with $z_i = (x_i, y_i)$, of labeled source instances drawn independently from P_Z. Additionally, we have a vector $\boldsymbol{t} = (\tilde{x}_1, \ldots, \tilde{x}_{n_t})$ of unlabeled target instances drawn independently from Q_X, which is the marginal distribution on $\mathcal{X}$ induced by $Q_{\tilde{Z}}$. Let $\tilde{\boldsymbol{z}} = (\tilde{z}_1, \ldots, \tilde{z}_{n_t}) \in \tilde{\mathcal{Z}}^{n_t}$, with $\tilde{z}_j = (\tilde{x}_j, \tilde{y}_j)$, be the corresponding vector of labeled target instances. Since this vector is not available to the learner, the learner cannot evaluate the empirical target loss

$$L_{\tilde{\boldsymbol{z}}}(w) = \frac{1}{n_t} \sum_{j=1}^{n_t} \ell(w(\tilde{x}_j), \tilde{y}_j). \tag{2}$$

To overcome this issue, we will obtain an upper bound on this quantity in Section 3.2. This bound contains a partial Wasserstein distance term and a weighted version of the empirical source loss, in which the weights are a function of the optimal coupling measure $\Pi^\star$ obtained when solving the optimization problem in the definition of the partial Wasserstein distance. This definition is provided below.

Definition 3.1 (Figalli, 2010, Eq. (2.1), Caffarelli & McCann, 2010, Eq. (1.8))**.** The partial Wasserstein distance with parameter α between two measures[2] P_X and $Q_{\tilde{X}}$ on $(\mathcal{X}, \Sigma_X)$ is defined as

$$\mathbb{PW}_\alpha(P_X, Q_{\tilde{X}}) = \inf_{\Pi \in \Gamma_\alpha(P_X, Q_{\tilde{X}})} \int c(x, \tilde{x}) \, \mathrm{d}\Pi(x, \tilde{x}), \tag{3}$$

where $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ is the so-called cost function (typically a metric) and $\Gamma_\alpha(P_X, Q_{\tilde{X}})$ is the set of all nonnegative measures Π on $\mathcal{X} \times \mathcal{X}$ for which $\Pi(\mathcal{X} \times \mathcal{X}) = \alpha$ and, for which, for all measurable sets $\mathcal{A}, \mathcal{B} \subseteq \mathcal{X}$, we have that $\Pi(\mathcal{A} \times \mathcal{X}) \leq P_X(\mathcal{A})$ and $\Pi(\mathcal{X} \times \mathcal{B}) \leq Q_{\tilde{X}}(\mathcal{B})$.

For the case in which the measures of interest are discrete and supported on m and n mass points, respectively, it is convenient to express P_X and $Q_{\tilde{X}}$ as vectors of dimension m and n, and the (coupling) measure as an $m \times n$ nonnegative (coupling) matrix Π with entries Π_{ij}. Then, one can rewrite (3) as

$$\mathbb{PW}_\alpha(P_X, Q_{\tilde{X}}) = \min_{\Pi \in \Gamma_\alpha(P_X, Q_{\tilde{X}})} \sum_{i=1}^{m} \sum_{j=1}^{n} c(x_i, \tilde{x}_j) \Pi_{ij} \tag{4}$$

where $\Gamma_\alpha(P_X, Q_{\tilde{X}})$ is the set of nonnegative matrices that satisfy[3] $\Pi \mathbf{1}_n \leq P_X$, $\Pi^T \mathbf{1}_m \leq Q_{\tilde{X}}$ and $\mathbf{1}_m^T \Pi \mathbf{1}_n = \alpha$.

[1] In the remainder of the paper, we will not specify sigma-algebras if they are clear from the context.

[2] We do not require that the two measures are probability measures. In particular, in our setup we will have $P_X(\mathcal{X}) \geq 1$.

[3] Here and throughout the paper, vector inequalities should be interpreted entry-wise.

3.2. Bounds on the Empirical Target Loss

Next, we present our main theoretical results: two bounds on the empirical target loss (2). Generally speaking, the bounds consist of four terms: (i) a weighted average of the loss computed on the labeled source instances, (ii) a partial Wasserstein term, (iii) a total variation term that allows us to make the bound explicit in the empirical target loss, and, similar to most theoretical bounds for domain adaptation available in the literature (Ben-David et al., 2006; Courty et al., 2017a; Shen et al., 2018), (iv) a non-computable term that dictates the difficulty of the PDA problem under consideration.

Drawing inspiration from Shen et al. (2018), we first present a bound in Theorem 3.2 in which the partial Wasserstein distance is between the empirical distributions of the source and target features. Then, drawing inspiration from Courty et al. (2017a), we extend it in Theorem 3.3 to the case in which the partial Wasserstein distance is between the joint empirical distribution of source features and labels and the joint empirical distribution of target features and predicted labels. While the feature-based approach can capture covariate shift, where only the marginal distributions on the input differ, a joint distribution-based approach is beneficial in the case of labeling distribution shift, *i.e.*, when the conditional distribution on labels given inputs also differ. Both bounds are in terms of the $\mathbb{PW}_\alpha$ distance. Partial domain adaptation is achieved by inflating the empirical source distribution by a parameter $1/\beta$, where $0 < \beta \leq 1$. Intuitively, the parameter β relates to the fraction of source instances we want to associate to target instances, whereas the parameter α corresponds to the fraction of target instances we want to consider. Hence, β allows for partial domain adaptation, while α can be used to avoid outliers in the target set.

Theorem 3.2 (Feature-based bound)**.** *Assume that the loss function ℓ is a metric on $\mathcal{Y}$ and consider the set $\mathcal{W}$ of hypotheses $w = g \circ f$ for which g is γ-Lipschitz with respect to ℓ. Let $P^f_{\boldsymbol{s}} = \frac{1}{n_s}\sum_{i=1}^{n_s}\delta_{f(x_i)}$ and $Q^f_{\boldsymbol{t}} = \frac{1}{n_t}\sum_{j=1}^{n_t}\delta_{f(\tilde{x}_j)}$ be the empirical source and target feature distributions, respectively, with feature extractor f. Then, for all $w \in \mathcal{W}$ and all $\alpha, \beta \in (0, 1]$,*

$$L_{\tilde{\boldsymbol{z}}}(w) \leq \sum_{i=1}^{n_s}\frac{p_i}{\alpha}\ell(w(x_i), y_i) + \frac{2}{\alpha}\mathbb{PW}_\alpha\Big(\tfrac{1}{\beta}P^f_{\boldsymbol{s}}, Q^f_{\boldsymbol{t}}\Big) + \frac{1}{2}\sum_{j=1}^{n_t}\left|\frac{1}{n_t} - \frac{q_j}{\alpha}\right| + 2L_f, \quad (5)$$

where the cost function in $\mathbb{PW}_\alpha$ is $c(x, \tilde{x}) = \gamma\|f(x) - f(\tilde{x})\|$, and the weights $\{p_i\}$ and $\{q_j\}$ are given by

$$p_i = (\Pi^\star \mathbf{1}_{n_t})_i, \quad i = 1, \ldots, n_s \quad (6)$$

$$q_j = \left((\Pi^\star)^T \mathbf{1}_{n_s}\right)_j, \quad j = 1, \ldots, n_t \quad (7)$$

with $\Pi^\star$ being the optimal coupling matrix in the definition of $\mathbb{PW}_\alpha(\frac{1}{\beta}P^f_{\boldsymbol{s}}, Q^f_{\boldsymbol{t}})$. Finally,

$$L_f = \min_{g' \in \mathcal{G}} \max_{\substack{z \in \boldsymbol{z} \cup \tilde{\boldsymbol{z}} \\ z = (x,y)}} \ell(g'(f(x)), y), \quad (8)$$

where $\mathcal{G}$ denotes the set of classifiers g' associated to hypotheses in $\mathcal{W}$, and where, with an abuse of notation, $\boldsymbol{z} \cup \tilde{\boldsymbol{z}}$ denotes the set of all labeled source and target instances.

The proof of Theorem 3.2 is provided in Appendix B. The weights p_i and q_j arise constructively from the partial transport problem corresponding to $\mathbb{PW}_\alpha(\frac{1}{\beta}P^f_{\boldsymbol{s}}, Q^f_{\boldsymbol{t}})$. The underlying intuition is that, when partially transporting the source data to the target data, the source samples that play the dominant role in the transportation plan should be the ones that are most similar to the target samples. Conversely, outliers are expected to be nearly ignored, leading to small values of the corresponding p_i and q_j. The connection between this transportation view and the actual prediction problem of interest is formalized in the proof of the bound, detailed in Appendix B.

Note that when $\alpha = 1$, we have that $q_j = 1/n_t$ and, hence, the third term on the right-hand side of (5) disappears. Setting $\alpha = 1$ is reasonable if we do not expect any outliers in the target set, and, hence, we want to consider all target instances equally. In contrast, $\alpha < 1$ and $\beta = 1$ may be suitable for the so called *open set* adaptation problem, where the target label space includes additional classes beyond the source label space (Panareda Busto & Gall, 2017).

Next, we present an analogous bound, in which joint feature-label distributions are used in place of feature-only distributions in the partial Wasserstein term.

Theorem 3.3 (Joint distribution-based bound)**.** *Assume that the loss function ℓ is a metric on $\mathcal{Y}$ and ζ-Lipschitz in each argument. Consider the set $\mathcal{W}$ of hypotheses $w = g \circ f$ for which g is γ-Lipschitz with respect to the Euclidean distance. Let $P^f_{\boldsymbol{z}} = \frac{1}{n_s}\sum_{i=1}^{n_s}\delta_{f(x_i), y_i}$ and $Q^w_{\boldsymbol{t}} = \frac{1}{n_t}\sum_{j=1}^{n_t}\delta_{f(\tilde{x}_j), w(\tilde{x}_j)}$ be the empirical joint source and estimated joint target distributions, respectively, for the hypothesis $w = g \circ f$. Then, for all $w \in \mathcal{W}$ and all $\alpha, \beta \in (0, 1]$,*

$$L_{\tilde{\boldsymbol{z}}}(w) \leq \sum_{i=1}^{n_s}\frac{\hat{p}_i}{\alpha}\ell(w(x_i), y_i) + \frac{1}{\alpha}\mathbb{PW}_\alpha\Big(\tfrac{1}{\beta}P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\Big) + \frac{1}{2}\sum_{j=1}^{n_t}\left|\frac{1}{n_t} - \frac{\hat{q}_j}{\alpha}\right| + \hat{L}_f, \quad (9)$$

where the underlying cost function for $\mathbb{PW}_\alpha$ is

$$c((x, y), (\tilde{x}, \tilde{y})) = \zeta\gamma\|f(x) - f(\tilde{x})\| + \ell(y, \tilde{y}), \quad (10)$$

the weights $\{\hat{p}_i\}$ and $\{\hat{q}_j\}$ are given by

$$\hat{p}_i = (\hat{\Pi}^\star \mathbf{1}_{n_t})_i, \quad i = 1, \ldots, n_s \quad (11)$$

$$\hat{q}_j = \left((\hat{\Pi}^\star)^T \mathbf{1}_{n_s}\right)_j, \quad j = 1, \ldots, n_t \tag{12}$$

with $\hat{\Pi}^\star$ being the optimal coupling matrix in the definition of $\mathbb{PW}_\alpha\left(\frac{1}{\beta} P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right)$, and

$$\hat{L}_f = \min_{g' \in \mathcal{G}} \left\{ \sum_{j=1}^{n_t} \frac{\hat{q}_j}{\alpha} \ell(g'(f(\tilde{x}_j)), \tilde{y}_j) \right\} + \Xi \tag{13}$$

with Ξ given in (63) (see Appendix C).

The proof of Theorem 3.3 is provided in Appendix C. Note that the cost in (10) coincides with the one proposed in Courty et al. (2017b). However, it is important to note that the values of the weights can differ between the two bounds. Indeed, while the weights are given by similar expressions, the underlying optimal coupling matrix differs in general. The same considerations on the role of the weights detailed after Theorem 3.2 also apply to Theorem 3.3. Furthermore, while both $\hat{L}_f$ in (13) and L_f in (8) relate to the difficulty of the PDA problem under consideration, they are incomparable in general. Finally, some terms in (5) have an extra factor of 2 compared to the corresponding terms in (9), and the underlying cost function in the partial Wasserstein distance in (9) has an extra term. This, in addition to the slight difference in assumptions on the loss, means that the two bounds given in (5) and (9) cannot be compared in general beyond the discussion above.

3.3. PAC-Bayes Generalization Bounds

In Section 3.2, we derived bounds on the empirical target loss for a fixed hypothesis in the PDA setting. However, these results cannot be applied directly to learned hypotheses. To proceed, we will use the *PAC-Bayesian* approach (McAllester, 1999; Catoni, 2007), which will allow us to obtain loss bounds for a learned hypothesis. These bounds hold with high probability over the choice of the training source and target samples.

While a wide array of PAC-Bayes generalization bounds are available (Alquier, 2024; Hellström et al., 2025), we restrict ourselves to the following one for simplicity.

Lemma 3.4. *Suppose that there exists a function $R : \mathcal{W} \times \mathcal{Z}^{n_s} \times \tilde{\mathcal{Z}}^{n_t}$ such that, for all $(w, \boldsymbol{z}, \tilde{\boldsymbol{z}}) \in \mathcal{W} \times \mathcal{Z}^{n_s} \times \tilde{\mathcal{Z}}^{n_t}$,*

$$L_{\tilde{\boldsymbol{z}}}(w) \leq R(w, \boldsymbol{z}, \tilde{\boldsymbol{z}}). \tag{14}$$

Let Q_W be a prior distribution on $\mathcal{W}$ and $P_{W|\boldsymbol{Z},\boldsymbol{T}}$ a posterior distribution on $\mathcal{W}$ given the labeled source samples[4] $\boldsymbol{Z} \sim P_Z^{n_s}$ and the unlabeled target samples $\boldsymbol{T}$. Here, $\boldsymbol{T}$ is the projection on $\mathcal{X}^{n_t}$ of $\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}$. Then, for every fixed $\lambda > 0$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$ over $(\boldsymbol{Z}, \tilde{\boldsymbol{Z}})$,

$$\mathbb{E}_{P_{W|\boldsymbol{Z},\boldsymbol{T}}}[L_{Q_{\tilde{Z}}}(W)] \leq \mathbb{E}_{P_{W|\boldsymbol{Z},\boldsymbol{T}}}[R(W, \boldsymbol{Z}, \tilde{\boldsymbol{Z}})] + \frac{\lambda}{8n_t} + \frac{D_{\mathrm{KL}}(P_{W|\boldsymbol{Z},\boldsymbol{T}} \,||\, Q_W) + \log \frac{1}{\delta}}{\lambda} \tag{15}$$

where D_{KL} denotes the Kullback-Leibler (KL) divergence.

We provide the proof of Lemma 3.4 in Appendix D. Note that, in Section 3.2, we derived functions R that can be combined with Lemma 3.4 to yield generalization bounds for PDA. We present these bounds below, beginning with a feature-based bound, obtained by combining Theorem 3.2 and Lemma 3.4.

Corollary 3.5. *Suppose that the assumptions of Theorem 3.2 and Lemma 3.4 hold, and consider the same notation as used therein. Furthermore, denote the decomposition of the hypothesis W as $W = G \circ F$. Then, for every choice of $\lambda > 0$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$ over $\boldsymbol{Z} \sim P_Z^{n_s}$, $\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}$,*

$$\mathbb{E}_{P_{W|\boldsymbol{Z},\boldsymbol{T}}}[L_{Q_{\tilde{Z}}}(W)] \leq B + \mathbb{E}_{P_{W|\boldsymbol{Z},\boldsymbol{T}}}\left[\sum_{i=1}^{n_s} \frac{p_i}{\alpha} \ell(W(X_i), Y_i) + \frac{2}{\alpha} \mathbb{PW}_\alpha\left(\tfrac{1}{\beta} P^F_{\boldsymbol{S}}, Q^F_{\boldsymbol{T}}\right) + \frac{1}{2} \sum_{j=1}^{n_t} \left| \frac{1}{n_t} - \frac{q_j}{\alpha} \right| + 2L_F \right], \tag{16}$$

where $(X_i, Y_i) = Z_i$ are the entries of $\boldsymbol{Z}$, we let $\boldsymbol{S}$ denote the projection on $\mathcal{X}$ of $\boldsymbol{Z}$, and we use the shorthand

$$B = \frac{\lambda}{8n_t} + \frac{D_{\mathrm{KL}}(P_{W|\boldsymbol{Z},\boldsymbol{T}} \,||\, Q_W) + \log \frac{1}{\delta}}{\lambda}. \tag{17}$$

Next, we present a joint distribution-based bound, which follows by Theorem 3.3 and Lemma 3.4.

Corollary 3.6. *Suppose that the assumptions of Theorem 3.3 and Lemma 3.4 hold, and consider the same notation as used therein and in Corollary 3.5. Then, for every choice of $\lambda > 0$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$ over $\boldsymbol{Z} \sim P_Z^{n_s}$, $\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}$,*

$$\mathbb{E}_{P_{W|\boldsymbol{Z},\boldsymbol{T}}}[L_{Q_{\tilde{Z}}}(W)] \leq B + \mathbb{E}_{P_{W|\boldsymbol{Z},\boldsymbol{T}}}\left[\sum_{i=1}^{n_s} \frac{\hat{p}_i}{\alpha} \ell(W(X_i), Y_i) + \frac{1}{\alpha} \mathbb{PW}_\alpha\left(\tfrac{1}{\beta} P^F_{\boldsymbol{Z}}, Q^W_{\boldsymbol{T}}\right) + \frac{1}{2} \sum_{j=1}^{n_t} \left| \frac{1}{n_t} - \frac{\hat{q}_j}{\alpha} \right| + \hat{L}_F \right]. \tag{18}$$

The proof techniques that we use to derive these generalization bounds yield several key advantages compared to prior results obtained in the more restrictive context of domain adaptation by Shen et al. (2018) and Courty et al. (2017a). First, when relating the loss in the target domain to the loss

[4] We denote by $P_Z^{n_s}$ the n_s-fold product of P_Z. Similarly, $Q_{\tilde{Z}}^{n_t}$ stands for the n_t-fold product of $Q_{\tilde{Z}}$.

in the source domain in Section 3.2, we work directly with the empirical measures. Consequently, the partial Wasserstein distances in our results are fully empirical. In contrast, in earlier derivations for the case of domain adaptation, the Wasserstein distance is between population measures, which need to be related to their empirical counterparts via a concentration-of-measure step. This step adds an additional term to the bound, the need for which is obviated by our approach. Furthermore, while most existing domain-adaptation algorithms compute domain alignment terms on the basis of learned feature distributions, the underlying generalization bounds available in the literature depend on the fixed input distribution. In contrast, our bounds explicitly depend on the learned features, directly motivating the feature-based approaches most commonly used in practice.

4. Algorithms for PDA: WARMPOT

Now, motivated by the bounds on the empirical target loss in Section 3.2, we propose a family of algorithms for the PDA problem. Specifically, focusing on the first two computable terms of (9) in Theorem 3.3, we consider the following optimization problem:

$$\min_{w} \left\{ \sum_{i=1}^{n_s} \hat{p}_i \ell(w(x_i), y_i) + \mathbb{PW}_\alpha\left(\tfrac{1}{\beta} P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right) \right\}. \quad (19)$$

Here, the cost function in the definition of the partial Wasserstein distance is $c((x,y),(\tilde{x},\tilde{y})) = \eta_1 \|f(x) - f(\tilde{x})\| + \eta_2 \ell(y,\tilde{y}))$ and the weights $\hat{p}_i$ are given by (11). By setting $\eta_2 = 0$, we observe that $\mathbb{PW}_\alpha(\frac{1}{\beta} P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}})$ reduces to $\mathbb{PW}_\alpha(\frac{1}{\beta} P^f_{\boldsymbol{s}}, Q^f_{\boldsymbol{t}})$, which is the domain alignment term appearing in (5) in Theorem 3.2. Thus, the optimization problem in (19) allows us to draw inspiration from both Theorem 3.2 and Theorem 3.3.

The parameters η_1 and η_2 determine the impact of the inter-feature and inter-label distances respectively, which act as regularizers, while α and β control the alignment between the source and target distributions.

We refer to an algorithm that minimizes (19) as *weighted and regularized minimizer via partial optimal transport* (WARMPOT). To clarify the role of the parameters and the relation to prior work, we discuss two extreme cases:

- WARMPOT with $\beta = 1$. In this case, the partial Wasserstein term aligns an α fraction of the source distribution with an α fraction of the target distribution. This is reminiscent of the MPOT algorithm, which uses a mini-batch approximation of $\mathbb{PW}_\alpha$ to solve the PDA problem (Nguyen et al., 2022). The key differences are that (i): MPOT uses a different cost function, where the inter-feature cost is given by the squared distance $\|f(x) - f(\tilde{x})\|^2$, and (ii): MPOT uses uniform weights for the source sample losses, rather than the p_i of WARMPOT.
- WARMPOT with $\alpha = 1$. Here, the source distribution is scaled so that its total mass is $1/\beta > 1$. Of this mass, 1 unit is aligned with the entire target distribution, whose total mass is 1. The PWAN algorithm (Wang et al., 2024) uses this alignment approach along with the heuristic class-level weights of the BA^{3}US algorithm (Liang et al., 2020), detailed in Section 5.3.

The proposed WARMPOT algorithm can then be interpreted as aligning an α fraction of the β-scaled source distribution with an α fraction of the target distribution. The use of two parameters α and β allows for asymmetry in this domain alignment process, which is necessary if the proportion of outliers differs between the source and target data sets.

5. Experiments

We now experimentally evaluate our proposed algorithm, WARMPOT, for PDA tasks. Specifically, in Section 5.1, we detail our experimental setup. In Section 5.2, we discuss the implementation details of WARMPOT. In Section 5.3, we investigate the impact of our proposed weight choice. Then, in Section 5.4, we compare WARMPOT against existing PDA methods. Finally, in Section 5.5, we visualize the weights used in WARMPOT and provide insight into their role. Additional details on the experiments are provided in Appendix E.

5.1. Setup

In the experiments, we focus on the Office-Home data set (Venkateswara et al., 2017), which consists of images of 65 objects belonging to 4 different domains: **A**rt, **C**lipart, **P**roduct, and **R**eal-World. To construct a PDA task, we consider a source data set consisting of all labeled samples from one domain and a target data set consisting of unlabeled samples from the first 25 classes of another domain. We consider all 12 possible combinations of source and target domains. This PDA setup has been widely studied and was considered by, among others, Nguyen et al. (2022); Wang et al. (2022); Gu et al. (2024).

5.2. Implementation of WARMPOT

We consider hypotheses $w = g \circ f$ consisting of a ResNet50 (He et al., 2016) feature extractor f pretrained on ImageNet (Russakovsky et al., 2015) and a fully connected network with a hidden layer of dimension 256 as the classifier g. We solve (19) using stochastic gradient descent, and compute $\mathbb{PW}_\alpha(\frac{1}{\beta} P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}})$ for each mini-batch following the method proposed by Nguyen et al. (2022). Throughout, we set ℓ to be the cross-entropy loss and the parameters of the domain alignment term to be $(\alpha, \beta) = (0.8, 0.35)$. The values of all

Table 1. Test accuracy on the Office-Home dataset using the weight choices described in Section 5.3.

Weighting scheme	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg
MPOT weights	62.2 (1.4)	81.2 (1.0)	88.6 (0.6)	73.0 (1.1)	77.0 (3.1)	79.2 (1.3)	74.4 (1.3)	61.7 (0.9)	85.1 (1.5)	80.0 (0.7)	65.9 (0.5)	83.8 (0.5)	76.0 (0.4)
BA^{3}US weights	**64.2** (1.5)	82.6 (2.0)	89.2 (0.4)	**75.5** (0.8)	76.4 (4.1)	80.2 (0.9)	**76.6** (1.4)	**62.9** (1.4)	**88.5** (1.1)	**81.3** (0.8)	**67.4** (0.6)	**86.7** (0.9)	**77.6** (0.4)
ARPM weights	55.9 (1.0)	75.8 (0.6)	87.1 (0.4)	69.7 (1.5)	73.7 (1.4)	75.2 (1.1)	72.4 (1.6)	55.8 (2.6)	81.2 (1.0)	80.5 (0.6)	63.7 (0.8)	83.9 (0.7)	72.9 (0.3)
WARMPOT (ours)	62.5 (1.2)	**83.0** (1.1)	**89.5** (0.3)	75.2 (1.1)	**78.4** (2.2)	**82.3** (1.3)	**76.6** (1.5)	61.4 (3.1)	88.0 (1.0)	81.1 (0.7)	66.5 (1.0)	86.6 (0.6)	**77.6** (0.7)

other hyperparameters are provided in Appendix E. [5] The results of a sensitivity analysis on α and β are discussed in Appendix G.

5.3. Comparison with Existing Weighting Schemes

In this section, we evaluate our weighting strategy. Specifically, we compare our choice for the weights in (11) with the following weighting schemes:

- MPOT weighting strategy (Nguyen et al., 2022), with uniform weights $\hat{p}_i = 1/n_s$.
- BA^{3}US weighting strategy (Liang et al., 2020), where

$$\hat{p}_i = \sum_{j=1}^{n_t} \mathbb{I}[w(\tilde{x}_j) = y_i]/n_t. \tag{20}$$

  Here, $\tilde{x}_j$ is a target instance and y_i is the label of the ith source instance x_i.
- ARPM weighting strategy (Gu et al., 2024), where

$$\min_{\substack{\hat{\boldsymbol{p}}\in\Delta \\ \hat{p}_i\geq 0}} \mathbb{W}_1 \left(\sum_{i=1}^{n_s} \hat{p}_i \delta_{f(x_i)}, \frac{1}{n_t} \sum_{j=1}^{n_t} \delta_{f(\tilde{x}_j)} \right). \tag{21}$$

  Here, $\hat{\boldsymbol{p}} = (\hat{p}_1, \ldots, \hat{p}_{n_s})$ and Δ is given by

$$\Delta = \left\{ \hat{\boldsymbol{p}} : \sum_{i=1}^{n_s} \hat{p}_i = 1, \sum_{i=1}^{n_s} \left(\hat{p}_i - \frac{1}{n_s} \right)^2 < \frac{\rho}{n_s} \right\} \tag{22}$$

  with ρ being a hyperparameter.

The results are presented in Table 1. Note that, in our weighting strategy, the weights are computed for each mini-batch at every iteration. On the contrary, in BA^{3}US and ARPM, the weights are computed on the entire dataset but only every n iterations. We set this weight update interval to be $n = 500$ for BA^{3}US and ARPM as suggested in Liang et al. (2020) and Gu et al. (2024) respectively. The experiments are repeated for 6 random seeds, and we report the average and the standard deviation. As seen in the table, WARMPOT results in better performance than MPOT and ARPM, and yield performance comparable to BA^{3}US. This illustrates that the weighting strategy suggested by the theoretical results reported in Section 3 is indeed effective. Results from a similar experiment using the ImageNet → Caltech data set (see Appendix F) confirm these findings. Interestingly, the ARPM weighting strategy can be seen as a variation of our weighting strategy. We discuss this connection in further detail in Appendix H.

5.4. Comparison with State of the Art

Next, we compare the performance of WARMPOT with the performance of alternative algorithms proposed in the literature for the Office-Home data set. The results of our analysis are detailed in Table 2.[6] First, we focus on comparisons to algorithms that, similar to WARMPOT, rely on a cost function of the form given in (19). Specifically, we consider MPOT (Nguyen et al., 2022) and PWAN (Wang et al., 2024). As shown in the table, WARMPOT achieves higher average test accuracy compared to PWAN and similar to that of MPOT.

In the same table, we broaden the comparison to a wider range of algorithms. As shown in the table, among the available algorithms, the best performance on Office-Home is achieved by ARPM (Gu et al., 2024), which relies on several heuristic loss terms beyond the weighted source loss. Motivated by our results in Section 5.3, which indicate that the WARMPOT weighting strategy yields better performance than that of ARPM, we also consider an algorithm that is identical to ARPM except that it uses the WARMPOT weights. We refer to this approach as ARPM+our-weights. Interestingly, ARPM+our-weights improves upon SOTA performance. This indicates that our theoretically motivated weighting strategy can lead to gains for PDA algorithms that involve a weighted source loss. We detail the choice of hyperparameters used to obtain the test results of ARPM+our-weights in Appendix E.

5.5. The WARMPOT Weights

In order to illustrate the behavior of the WARMPOT weights, we focus on a single task from the Office-Home data set, namely, the P→A task. In Fig. 1, we present the weights $\hat{p}_i$ in (11) obtained by evaluating the transport matrix $\hat{\Pi}^\star$ achieving $\mathbb{PW}_\alpha(\frac{1}{\beta} P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}})$ over the entire data set,

[5] Open source Python implementation of WARMPOT: https://github.com/JayD2106/WARMPOT.

[6] The test accuracy scores accompanied by standard deviation in Table 2 are obtained by reproducing the results reported in the corresponding papers.

Table 2. Test accuracy on the Office-Home dataset.

Algorithm	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg
ResNet-50 (He et al., 2016)	46.3	67.5	75.9	59.1	59.9	62.7	58.2	41.8	74.9	67.4	48.2	74.2	61.4
ADDA (Tzeng et al., 2017)	45.2	68.8	79.2	64.6	60.0	68.3	57.6	38.9	77.5	70.3	45.2	78.3	62.8
CDAN+E (Long et al., 2018)	47.5	65.9	75.7	57.1	54.1	63.4	59.6	44.3	72.4	66.0	49.9	72.8	60.7
IWAN (Zhang et al., 2018)	53.9	54.5	78.1	61.3	48.0	63.3	54.2	52.0	81.3	76.5	56.8	82.9	63.6
PADA (Cao et al., 2018)	52.0	67.0	78.7	52.2	53.8	59.0	52.6	43.2	78.8	73.7	56.6	77.1	62.1
ETN (Cao et al., 2019)	59.2	77.0	79.5	62.9	65.7	75.0	68.3	55.4	84.4	75.7	57.7	84.5	70.5
DRCN (Li et al., 2020)	54.0	76.4	83.0	62.1	64.5	71.0	70.8	49.8	80.5	77.5	59.1	79.9	69.0
BA^{3}US (Liang et al., 2020)	60.6	83.2	88.4	71.8	72.8	83.4	75.5	61.6	86.5	79.3	62.8	86.1	76.0
ISRA+BA^{3}US (Xiao et al., 2021)	64.7	83.0	89.1	75.7	75.5	85.4	78.5	64.2	88.1	81.3	65.3	86.7	78.2
SHOT++ (Liang et al., 2021)	65.0	85.8	**93.4**	78.8	77.4	87.3	79.3	66.0	89.6	81.3	68.1	86.8	79.9
SPDA (Guo et al., 2022)	64.2	87.8	88.0	74.3	75.1	79.1	79.4	58.9	85.1	81.4	67.4	84.1	77.1
APDA-CI (Lin et al., 2022)	61.7	86.9	90.5	77.2	76.9	83.8	79.6	63.8	88.5	85.0	65.8	86.2	78.8
CLA (Yang et al., 2023)	66.7	85.6	90.9	75.6	76.9	86.8	78.8	67.4	88.7	81.7	66.9	87.8	79.5
RAN (Wu et al., 2023)	63.3	83.1	89.0	75.0	74.5	82.9	78.0	61.2	86.7	79.9	63.5	85.0	76.8
JUMBOT (Fatras et al., 2021)	62.7	77.5	84.4	76.0	73.3	80.5	74.7	60.8	85.1	80.2	66.5	83.9	75.5
STCPDA (He et al., 2023)	63.1	87.8	90.1	77.2	75.4	85.6	**81.4**	62.4	**90.5**	82.6	69.5	88.2	79.5
SLM (Sahoo et al., 2023)	61.1	84.0	91.4	76.5	75.0	81.8	74.6	55.6	87.8	82.3	57.8	83.5	76.0
SAN++ (Cao et al., 2022)	61.3	81.6	88.6	72.8	76.4	81.9	74.5	57.7	87.2	79.7	63.8	86.1	76.0
IDSP (Li & Chen, 2022)	60.8	80.8	87.3	69.3	76.0	80.2	74.7	59.2	85.3	77.8	61.3	85.7	74.9
MOT (Luo & Ren, 2023)	63.1	86.1	92.3	78.7	**85.4**	**89.6**	79.8	62.3	89.7	83.8	67.0	89.6	80.6
AR (Gu et al., 2021)	67.4	85.3	90.0	77.3	70.6	85.2	79.0	64.8	89.5	80.4	66.2	86.4	78.3
ARPM (Gu et al., 2024)	67.4 (2.5)	**88.4** (1.4)	92.7 (1.1)	79.9 (2.1)	82.6 (2.6)	87.0 (0.7)	78.8 (3.1)	**69.1** (1.6)	89.6 (0.3)	86.0 (1.0)	69.5 (2.2)	**89.7** (1.1)	81.7 (0.7)
MPOT (Nguyen et al., 2022)	64.6	80.6	87.1	76.4	77.6	83.5	77.0	63.7	87.6	81.4	68.5	87.3	77.9
PWAN (Wang et al., 2024)	63.3 (1.8)	84.1 (1.8)	89.3 (0.6)	76.7 (1.1)	75.6 (2.0)	83.8 (1.8)	76.6 (0.8)	60.7 (2.2)	86.7 (0.8)	80.1 (0.6)	64.4 (0.5)	86.6 (0.7)	77.3 (0.5)
WARMPOT (ours)	62.5 (1.2)	83.0 (1.1)	89.5 (0.3)	75.2 (1.1)	78.4 (2.2)	82.3 (1.3)	76.6 (1.5)	61.4 (3.1)	88.0 (1.0)	81.1 (0.7)	66.5 (1.0)	86.6 (0.6)	77.6 (0.7)
ARPM+our-weights	**69.0** (1.9)	87.2 (1.7)	92.8 (0.7)	**81.0** (1.1)	83.4 (2.7)	86.0 (2.8)	79.9 (2.2)	**69.1** (0.8)	90.2 (0.9)	**86.6** (0.8)	**69.7** (2.3)	88.7 (1.0)	**82.0** (0.4)

at the end of the training process. For illustrative purposes, we normalize the $\hat{p}_i$ by $1/(\beta n_s)$ to obtain values in the interval $[0, 1]$. The left plot in Fig. 1 shows the distribution of all weights on source instances. The middle and the right plots show the distribution of the normalized weights on the shared and the outlier class instances, respectively. The weight proportion assigned to outlier source samples in WARMPOT is just 6.04%, even though 58.41% of the source samples belong to outlier classes. A total of 64.4% outlier class instances have been assigned to the smallest bin, which helps in minimizing the effect of negative transfer.

6. Conclusion

In this work, we obtain generalization bounds for PDA tasks. In particular, our bounds depend on a partial Wasserstein distance, and hence provide a theoretical motivation for using it as a domain alignment term. While several existing algorithms in the literature take such an approach, a theoretical justification was previously missing. Furthermore, our bounds constructively give rise to explicit source data weights, which can help alleviate negative transfer. In contrast, prior work used heuristic weight choices, which were not directly motivated by theoretical considerations.

Inspired by our bounds, we propose the algorithm WARMPOT to minimize them. Through numerical experiments, we demonstrate that WARMPOT is competitive with recent approaches to PDA. Furthermore, we show that the performance of the SOTA algorithm ARPM is improved when its weighting scheme is replaced with that of WARMPOT. This, along with an additional ablation study, corroborates the utility of our proposed weights.

It should be noted that an exact minimization of our bounds is prohibitively expensive from a computational standpoint, and hence, WARMPOT relies on some approximations. Additional performance gains may be obtained by optimizing this implementation. Furthermore, the SOTA algorithm ARPM includes additional loss terms that aim to reduce prediction uncertainty and improve robustness. Such quantities are not explicitly present in our generalization bounds, but are studied by Gu et al. (2024, Thm. 1). A promising direction for future research is to explore these aspects within our theoretical framework, potentially enabling more powerful algorithms.

Acknowledgements

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none which we feel must be

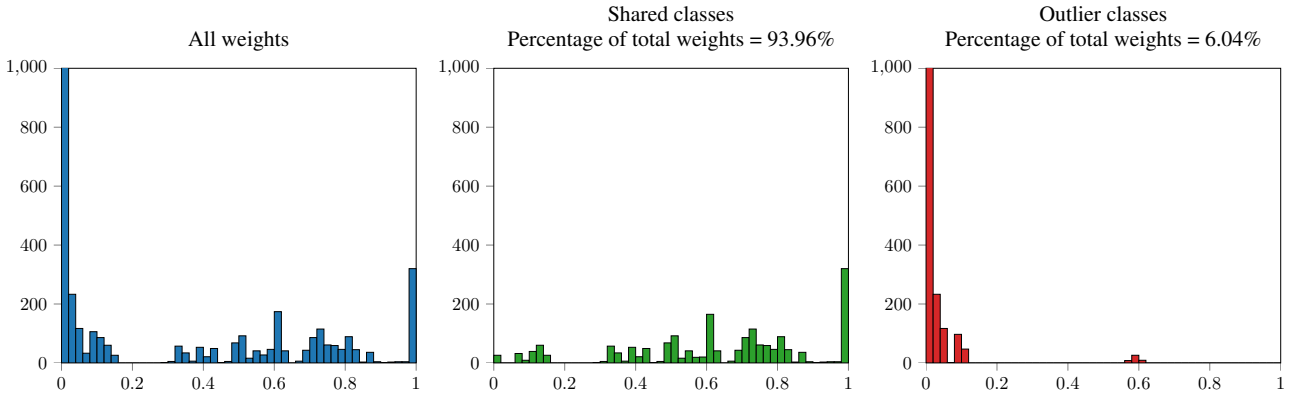

Figure 1. The distribution of WARMPOT weights for the task P→A. Most of the weights of the outlier classes are close to zero, suggesting that most of the outliers are successfully omitted when training the classifier.

specifically highlighted here.

References

Alquier, P. User-friendly introduction to PAC-Bayes bounds. *Found. Trends Mach. Learn.*, 17(2):174–303, Jan. 2024.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *Proc. Int. Conf. Mach. Learning (ICML)*, Sydney, Australia, Aug. 2017.

Ben-David, S., Blitzer, J., Crammer, K., and Pereira, F. Analysis of representations for domain adaptation. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, Canada, Dec. 2006.

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

Boyd, S. and Vandenberghe, L. *Convex Optimization*. Cambridge Univ. Press, Cambridge, UK, 2004.

Caffarelli, L. A. and McCann, R. J. Free boundaries in optimal transport and Monge-Ampere obstacle problems. *Ann. Math.*, 171(2):673–730, Mar. 2010.

Cao, Z., Ma, L., Long, M., and Wang, J. Partial adversarial domain adaptation. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, Sep. 2018.

Cao, Z., You, K., Long, M., Wang, J., and Yang, Q. Learning to transfer examples for partial domain adaptation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, June 2019.

Cao, Z., You, K., Zhang, Z., Wang, J., and Long, M. From big to small: Adaptive learning to partial-set domains. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 45(2): 1766–1780, Mar. 2022.

Catoni, O. *PAC-Bayesian Supervised Classification: the Thermodynamics of Statistical Learning*. IMS Lecture Notes Monogr. Ser., 56, Beachwood, OH, USA, 2007.

Chang, W., Shi, Y., Tuan, H., and Wang, J. Unified optimal transport framework for universal domain adaptation. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, New Orleans, Louisiana, USA, Nov. 2022.

Courty, N., Flamary, R., and Tuia, D. Domain adaptation with regularized optimal transport. In *Proc. Mach. Learn. Knowl. Discov. Databases (ECML PKDD)*, Nancy, France, Sep. 2014.

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. Joint distribution optimal transportation for domain adaptation. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, USA, Dec. 2017a.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal transport for domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 39(9):1853–1865, Oct. 2017b.

Damodaran, B. B., Kellenberger, B., Flamary, R., Tuia, D., and Courty, N. DeepJDOT: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, Sep. 2018.

Farahani, A., Voghoei, S., Rasheed, K., and Arabnia, H. R. A brief review of domain adaptation. In *Adv. Data Sci. Inf. Eng.*, Springer, Cham, Switzerland, Oct. 2021.

Fatras, K., Séjourné, T., Flamary, R., and Courty, N. Unbalanced minibatch optimal transport; applications to

domain adaptation. In *Proc. Int. Conf. Mach. Learning (ICML)*, Virtual Conference, July 2021.

Figalli, A. The optimal partial transport problem. *Arch. Ration. Mech. Anal.*, 195(2):533–560, Jan. 2010.

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D. J., Tavenard, R., Tong, A., and Vayer, T. POT: Python optimal transport. *J. Mach. Learn. Res. (JMLR)*, 22(78):1–8, Jan. 2021.

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

Griffin, G., Holub, A., and Perona, P. Caltech 256, Apr. 2022.

Gu, X., Yu, X., Sun, J., Xu, Z., et al. Adversarial reweighting for partial domain adaptation. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Virtual Conference, Dec. 2021.

Gu, X., Yu, X., Yang, Y., Sun, J., and Xu, Z. Adversarial reweighting with α-power maximization for domain adaptation. *Int. J. Comput. Vis. (IJCV)*, 132(10):4768–4791, May 2024.

Guo, P., Zhu, J., and Zhang, Y. Selective partial domain adaptation. In *Proc. Br. Mach. Vis. Conf. (BMVC)*, London, UK, Nov. 2022.

He, C., Li, X., Xia, Y., Tang, J., Yang, J., and Ye, Z. Addressing the overfitting in partial domain adaptation with self-training and contrastive learning. *IEEE Trans. Circuits Syst. Video Technol.*, 34(3):1532–1545, July 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, USA, June 2016.

Hellström, F., Durisi, G., Guedj, B., and Raginsky, M. Generalization bounds: Perspectives from information theory and PAC-Bayes. *Found. Trends Mach. Learn.*, 18(1): 1–223, Jan. 2025.

Li, S., Liu, C. H., Lin, Q., Wen, Q., Su, L., Huang, G., and Ding, Z. Deep residual correction network for partial domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 43(7):2329–2344, Jan. 2020.

Li, W. and Chen, S. Partial domain adaptation without domain alignment. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 45(7):8787–8797, Dec. 2022.

Liang, J., Wang, Y., Hu, D., He, R., and Feng, J. A balanced and uncertainty-aware approach for partial domain adaptation. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Virtual Conference, Aug. 2020.

Liang, J., Hu, D., Wang, Y., He, R., and Feng, J. Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 44(11):8602–8617, Aug. 2021.

Lin, K.-Y., Zhou, J., Qiu, Y., and Zheng, W.-S. Adversarial partial domain adaptation by cycle inconsistency. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Tel Aviv, Israel, Oct. 2022.

Lipton, Z., Wang, Y.-X., and Smola, A. Detecting and correcting for label shift with black box predictors. In *Proc. Int. Conf. Mach. Learning (ICML)*, Stockholm, Sweden, July 2018.

Long, M., Cao, Y., Wang, J., and Jordan, M. Learning transferable features with deep adaptation networks. In *Proc. Int. Conf. Mach. Learning (ICML)*, Lille, France, July 2015.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. Conditional adversarial domain adaptation. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Montréal, Canada, Dec. 2018.

Luo, Y.-W. and Ren, C.-X. MOT: Masked optimal transport for partial domain adaptation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, Canada, June 2023.

Luo, Y.-W. and Ren, C.-X. When invariant representation learning meets label shift: Insufficiency and theoretical insights. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 46(12):9407–9422, June 2024.

McAllester, D. A. PAC-Bayesian model averaging. In *Proc. Conf. Comput. Learn. Theory (COLT)*, Santa Cruz, CA, USA, July 1999.

Nagarajan, V. and Kolter, J. Z. Uniform convergence may be unable to explain generalization in deep learning. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, Canada, Dec. 2019.

Nguyen, K., Nguyen, D., Pham, T., Ho, N., et al. Improving mini-batch optimal transport via partial transportation. In *Proc. Int. Conf. Mach. Learning (ICML)*, Baltimore, MA, USA, July 2022.

Ohnishi, Y. and Honorio, J. Novel change of measure inequalities with applications to PAC-Bayesian bounds and Monte Carlo estimation. In *Proc. Int. Conf. Artif. Intell. Stat. (AISTATS)*, Virtual Conference, Apr. 2021.

Panareda Busto, P. and Gall, J. Open set domain adaptation. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017.

Redko, I., Habrard, A., and Sebban, M. Theoretical analysis of domain adaptation with optimal transport. In *Proc. Mach. Learn. Knowl. Discov. Databases (ECML PKDD)*, Skopje, Macedonia, Sep. 2017.

Redko, I., Morvant, E., Habrard, A., Sebban, M., and Bennani, Y. *Advances in domain adaptation theory*. Elsevier, Oxford, UK, 2019.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis. (IJCV)*, 115(3):211–252, Apr. 2015.

Sahoo, A., Panda, R., Feris, R., Saenko, K., and Das, A. Select, label, and mix: Learning discriminative invariant feature representations for partial domain adaptation. In *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, Waikoloa, Hawaii, USA, Jan. 2023.

Shen, J., Qu, Y., Zhang, W., and Yu, Y. Wasserstein distance guided representation learning for domain adaptation. In *Proc. AAAI Conf. Artif. Intell. (AAAI)*, New Orleans, Louisiana, USA, Apr. 2018.

Tachet des Combes, R., Zhao, H., Wang, Y.-X., and Gordon, G. J. Domain adaptation with conditional distribution matching and generalized label shift. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Virtual Conference, Dec. 2020.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. Adversarial discriminative domain adaptation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, Hawaii, USA, July 2017.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, Hawaii, USA, July 2017.

Wainwright, M. J. *High-Dimensional Statistics: a Non-Asymptotic Viewpoint*. Cambridge Univ. Press, Cambridge, U.K., 2019.

Wang, Z.-M., Xue, N., Lei, L., and Xia, G.-S. Partial Wasserstein adversarial network for non-rigid point set registration. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, Virtual Conference, Apr. 2022.

Wang, Z.-M., Xue, N., Lei, L., Jörnsten, R., and Xia, G.-S. Partial distribution matching via partial Wasserstein adversarial networks. *arXiv*, Sep. 2024.

Wu, K., Wu, M., Chen, Z., Jin, R., Cui, W., Cao, Z., and Li, X. Reinforced adaptation network for partial domain adaptation. *IEEE Trans. Circuits Syst. Video Technol.*, 33 (5):2370–2380, Nov. 2023.

Xiao, W., Ding, Z., and Liu, H. Implicit semantic response alignment for partial domain adaptation. In *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Virtual Conference, Dec. 2021.

Yang, C., Cheung, Y.-M., Ding, J., Tan, K. C., Xue, B., and Zhang, M. Contrastive learning assisted-alignment for partial domain adaptation. *IEEE Trans. Neural Netw. Learn. Syst.*, 34(10):7621–7634, Feb. 2023.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3):107–115, Feb. 2021.

Zhang, J., Ding, Z., Li, W., and Ogunbona, P. Importance weighted adversarial nets for partial domain adaptation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, June 2018.

A. Preliminaries

We recall some definitions that are used in the main text as well as in the appendices. We will also establish a useful lemma.

Definition A.1. A function $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ is called a metric on $\mathcal{X}$ if it is real-valued, finite, and nonnegative, and if for all $a, b, c \in \mathcal{X}$:

(i) $\rho(a, b) = 0$ if and only if $a = b$,

(ii) $\rho(a, b) = \rho(b, a)$ (symmetry),

(iii) $\rho(a, c) \leq \rho(a, b) + \rho(b, c)$ (triangle inequality).

We shall also use the so-called reverse triangle inequality,

$$|\rho(a, b) - \rho(a, c)| \leq \rho(b, c) \tag{23}$$

which can be readily obtained from the properties in Definition A.1.

Definition A.2. A function $g : \mathbb{R}^d \to \mathcal{Y}$ is γ-Lipschitz with respect to a metric ρ on $\mathcal{Y}$ if for all $t, t' \in \mathbb{R}^d$

$$\rho(g(t), g(t')) \leq \gamma \|t - t'\|. \tag{24}$$

Definition A.3. The total variation distance between two discrete probability distributions P and Q on $\mathcal{Z}$ is defined as

$$\mathrm{TV}(P, Q) = \frac{1}{2} \sum_{z \in \mathcal{Z}} |P(z) - Q(z)|. \tag{25}$$

Lemma A.4. *Assume that the loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ is a metric on $\mathcal{Y}$. Let $\mathcal{W}$ be the set of all hypotheses $w = g \circ f$ such that g is γ-Lipschitz with respect to ℓ. Let L_f and $\boldsymbol{z} \cup \tilde{\boldsymbol{z}}$ be defined as in Theorem 3.2. Then, for all pairs $z = (x, y)$ and $\tilde{z} = (\tilde{x}, \tilde{y})$ in $\boldsymbol{z} \cup \tilde{\boldsymbol{z}}$, we have*

$$|\ell(w(x), y) - \ell(w(\tilde{x}), \tilde{y})| \leq 2\gamma \|f(x) - f(\tilde{x})\| + 2L_f. \tag{26}$$

Proof. Consider two arbitrary pairs (x, y) and $(\tilde{x}, \tilde{y})$ in $\boldsymbol{z} \cup \tilde{\boldsymbol{z}}$. Furthermore, fix f and let g^* be a classifier achieving L_f in (8). Then,

$$\begin{aligned}
\ell(w(x), y) &= \ell(g(f(x)), y) && (27)\\
&\leq \ell(g(f(x)), g^*(f(x))) + \ell(g^*(f(x)), y) && (28)\\
&\leq \ell(g(f(x)), g^*(f(x))) + L_f && (29)\\
&\leq \ell(g(f(x)), g(f(\tilde{x}))) + \ell(g(f(\tilde{x})), g^*(f(x))) + L_f && (30)\\
&\leq \gamma \|f(x) - f(\tilde{x})\| + \ell(g(f(\tilde{x})), g^*(f(x))) + L_f && (31)\\
&\leq \gamma \|f(x) - f(\tilde{x})\| + \ell(g(f(\tilde{x})), g^*(f(\tilde{x}))) + \ell(g^*(f(\tilde{x})), g^*(f(x))) + L_f && (32)\\
&\leq 2\gamma \|f(x) - f(\tilde{x})\| + \ell(g(f(\tilde{x})), g^*(f(\tilde{x}))) + L_f && (33)\\
&\leq 2\gamma \|f(x) - f(\tilde{x})\| + \ell(g(f(\tilde{x})), \tilde{y}) + \ell(\tilde{y}, g^*(f(\tilde{x}))) + L_f && (34)\\
&\leq 2\gamma \|f(x) - f(\tilde{x})\| + \ell(w(\tilde{x}), \tilde{y}) + 2L_f. && (35)
\end{aligned}$$

Here, (27) follows because $w = g \circ f$; in (28) we used the triangle inequality; in (29) we used the max-min inequality (Boyd & Vandenberghe, 2004, Eq. (5.46)) as well as the definition of L_f; in (30) we again used the triangle inequality; in (31) we used that g is γ-Lipschitz; in (32) we used the triangle inequality; in (33) we used that g^* is γ-Lipschitz; in (34) we used the triangle inequality; and finally, (35) follows again from the max-min inequality and the definition of L_f. Similarly, starting with $\ell(w(\tilde{x}), \tilde{y})$ and proceeding analogously, we conclude that

$$\ell(w(\tilde{x}), \tilde{y}) \leq 2\gamma \|f(x) - f(\tilde{x})\| + \ell(w(x), y) + 2L_f. \tag{36}$$

Combining (35) and (36), we obtain the desired result. □

B. Proof of Theorem 3.2

Note that there may be duplicate features in $\{f(x_i)\}_{i=1}^{n_s}$ and $\{f(\tilde{x}_i)\}_{i=1}^{n_t}$. Hence, strictly speaking, $P_{\boldsymbol{s}}^f$ and $Q_{\boldsymbol{t}}^f$ are probability vectors whose dimensions are given by the number of distinct features, and multiplicities need to be accounted for. However, in our proof, this yields the same result as if we treat the duplicate values as separate features with identical cost values. Hence, for simplicity but without loss of generality, we assume that the features in both $\{f(x_i)\}_{i=1}^{n_s}$ and $\{f(\tilde{x}_i)\}_{i=1}^{n_t}$ are distinct. This allows us to view $P_{\boldsymbol{s}}^f$ and $Q_{\boldsymbol{t}}^f$ as probability vectors of dimensions n_s and n_t respectively, where all entries of each vector are equal, *i.e.*, $P_{\boldsymbol{s}}^f = [\frac{1}{n_s}, \ldots, \frac{1}{n_s}]^T$ and $Q_{\boldsymbol{t}}^f = [\frac{1}{n_t}, \ldots, \frac{1}{n_t}]^T$.

Now, define the $n_s \times n_t$ cost matrix C with entries $C_{ij} = \gamma \|f(x_i) - f(\tilde{x}_j)\|$. We consider the partial Wasserstein distance between $\frac{1}{\beta} P_{\boldsymbol{s}}^f$ and $Q_{\boldsymbol{t}}^f$, which is given by (see the definition in (4))

$$\mathbb{PW}_\alpha \left(\tfrac{1}{\beta} P_{\boldsymbol{s}}^f, Q_{\boldsymbol{t}}^f \right) = \min_{\Pi \in \Gamma_\alpha \left(\frac{1}{\beta} P_{\boldsymbol{s}}^f, Q_{\boldsymbol{t}}^f \right)} \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} C_{ij} \Pi_{ij}, \tag{37}$$

where $\Gamma_\alpha \left(\frac{1}{\beta} P_{\boldsymbol{s}}^f, Q_{\boldsymbol{t}}^f \right) = \left\{ \Pi \in \mathbb{R}^{n_s \times n_t} \ : \ \Pi \mathbf{1}_{n_t} \leq \frac{1}{\beta} P_{\boldsymbol{s}}^f, \Pi^T \mathbf{1}_{n_s} \leq Q_{\boldsymbol{t}}^f, \mathbf{1}_{n_s}^T \Pi \mathbf{1}_{n_t} = \alpha \right\}$.

Let $\Pi^\star \in \Gamma_\alpha \left(\frac{1}{\beta} P_{\boldsymbol{s}}^f, Q_{\boldsymbol{t}}^f \right)$ attain the minimum in (37). Then

$$\begin{aligned}
2\mathbb{PW}_\alpha \left(\frac{1}{\beta} P_{\boldsymbol{s}}^f, Q_{\boldsymbol{t}}^f \right) &= \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} 2 C_{ij} \Pi_{ij}^\star && (38) \\
&= \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} 2\gamma \|f(x_i) - f(\tilde{x}_j)\| \Pi_{ij}^\star && (39) \\
&\geq \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} (|\ell(w(\tilde{x}_j), \tilde{y}_j) - \ell(w(x_i), y_i)| - 2L_f) \Pi_{ij}^\star && (40) \\
&\geq \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} (\ell(w(\tilde{x}_j), \tilde{y}_j) - \ell(w(x_i), y_i) - 2L_f) \Pi_{ij}^\star && (41) \\
&= \sum_{j=1}^{n_t} \ell(w(\tilde{x}_j), \tilde{y}_j) \sum_{i=1}^{n_s} \Pi_{ij}^\star - \sum_{i=1}^{n_s} \ell(w(x_i), y_i) \sum_{j=1}^{n_t} \Pi_{ij}^\star - 2L_f \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} \Pi_{ij}^\star && (42) \\
&= \sum_{j=1}^{n_t} \ell(w(\tilde{x}_j), \tilde{y}_j) ((\Pi^\star)^T \mathbf{1}_{n_s})_j - \sum_{i=1}^{n_s} \ell(w(x_i), y_i) (\Pi^\star \mathbf{1}_{n_t})_i - 2L_f \mathbf{1}_{n_s}^T \Pi^\star \mathbf{1}_{n_t} && (43) \\
&= \sum_{j=1}^{n_t} \ell(w(\tilde{x}_j), \tilde{y}_j) q_j - \sum_{i=1}^{n_s} \ell(w(x_i), y_i) p_i - 2L_f \alpha. && (44)
\end{aligned}$$

Here, in (38) we used (37); in (39) we used the definition of the cost function; (40) follows from Lemma A.4; and finally, in (44) we used that $\mathbf{1}_{n_s}^T \Pi^\star \mathbf{1}_{n_t} = \alpha$ by definition, and we set $p_i = (\Pi^\star \mathbf{1}_{n_t})_i$, $i = 1, \ldots, n_s$ and $q_j = \left((\Pi^\star)^T \mathbf{1}_{n_s} \right)_j$, $j = 1, \ldots, n_t$. The inequality just obtained can be rewritten as

$$\sum_{j=1}^{n_t} \ell(w(\tilde{x}_j), \tilde{y}_j) q_j \leq \sum_{i=1}^{n_s} \ell(w(x_i), y_i) p_i + 2\mathbb{PW}_\alpha \left(\frac{1}{\beta} P_{\boldsymbol{s}}^f, Q_{\boldsymbol{t}}^f \right) + 2\alpha L_f. \tag{45}$$

Now, let $\boldsymbol{q} = [q_1, \ldots, q_{n_t}]^T$ and define the following empirical distributions on $\tilde{\boldsymbol{z}}$:

$$Q_{\tilde{\boldsymbol{z}}} = \sum_{j=1}^{n_t} \frac{1}{n_t} \delta_{\tilde{z}_j}, \qquad Q_{\tilde{\boldsymbol{z}}}^{\boldsymbol{q}} = \sum_{j=1}^{n_t} \frac{q_j}{\alpha} \delta_{\tilde{z}_j}. \tag{46}$$

Using that the loss function is supported on $[0, 1]$, we perform the following change of measure (Ohnishi & Honorio, 2021, Lemma 4):

$$\mathbb{E}_{(X,Y) \sim Q_{\tilde{\boldsymbol{z}}}} [\ell(w(X), Y)] \leq \mathbb{E}_{(X,Y) \sim Q_{\tilde{\boldsymbol{z}}}^{\boldsymbol{q}}} [\ell(w(X), Y)] + \mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q_{\tilde{\boldsymbol{z}}}^{\boldsymbol{q}}) \tag{47}$$

where the total variation distance $\mathrm{TV}(\cdot,\cdot)$ was introduced in Definition A.3. This implies that

$$\frac{1}{n_t}\sum_{j=1}^{n_t}\ell(w(\tilde{x}_j),\tilde{y}_j) \leq \frac{1}{\alpha}\sum_{j=1}^{n_t}\ell(w(\tilde{x}_j),\tilde{y}_j)q_j + \mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\boldsymbol{q}}_{\tilde{\boldsymbol{z}}}) \tag{48}$$

or, equivalently,

$$L_{\tilde{\boldsymbol{z}}}(w) \leq \frac{1}{\alpha}\sum_{j=1}^{n_t}\ell(w(\tilde{x}_j),\tilde{y}_j)q_j + \mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\boldsymbol{q}}_{\tilde{\boldsymbol{z}}})\,. \tag{49}$$

Finally, we note that

$$\mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\boldsymbol{q}}_{\tilde{\boldsymbol{z}}}) = \frac{1}{2}\sum_{j=1}^{n_t}\left|\frac{1}{n_t} - \frac{q_j}{\alpha}\right|. \tag{50}$$

We substitute (50) in (49) and (49) in (45) to get the desired result.

C. Proof of Theorem 3.3

As in Appendix B, we assume for simplicity that we can view $P^f_{\boldsymbol{z}}$ and $Q^w_{\boldsymbol{t}}$ as probability vectors of dimensions n_s and n_t respectively, where all entries of each vector are equal, *i.e.*, $P^f_{\boldsymbol{z}} = [\frac{1}{n_s},\ldots,\frac{1}{n_s}]^T$ and $Q^w_{\boldsymbol{t}} = [\frac{1}{n_t},\ldots,\frac{1}{n_t}]^T$.

We now consider an $n_s \times n_t$ cost matrix C with entries $C_{ij} = \zeta\gamma\|f(x_i) - f(\tilde{x}_j)\| + \ell(y_i, w(\tilde{x}_j))$. We consider the partial Wasserstein distance between $\frac{1}{\beta}P^f_{\boldsymbol{z}}$ and $Q^w_{\boldsymbol{t}}$, which is given by (see the definition in (4))

$$\mathbb{PW}_\alpha\left(\tfrac{1}{\beta}P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right) = \min_{\Pi\in\Gamma_\alpha\left(\frac{1}{\beta}P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right)}\sum_{i=1}^{n_s}\sum_{j=1}^{n_t}C_{ij}\Pi_{ij}, \tag{51}$$

where $\Gamma_\alpha\left(\frac{1}{\beta}P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right) = \left\{\Pi\in\mathbb{R}^{n_s\times n_t} \;:\; \Pi\mathbf{1}_{n_t} \leq \frac{1}{\beta}P^f_{\boldsymbol{z}}, \Pi^T\mathbf{1}_{n_s} \leq Q^w_{\boldsymbol{t}}, \mathbf{1}^T_{n_s}\Pi\mathbf{1}_{n_t} = \alpha\right\}$. We define $Q_{\tilde{\boldsymbol{z}}}$ and $Q^{\hat{\boldsymbol{q}}}_{\tilde{\boldsymbol{z}}}$ in the same way as in (46), where $\hat{\boldsymbol{q}} = [\hat{q}_1,\ldots,\hat{q}_{n_t}]^T$. Then, given the feature map f, for every fixed hypothesis $w'\in\mathcal{W}$ that can be decomposed as $w' = g'\circ f$, we have

$$\alpha L_{\tilde{\boldsymbol{z}}}(w) \leq \alpha\mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\hat{\boldsymbol{q}}}_{\tilde{\boldsymbol{z}}}) + \sum_{j=1}^{n_t}\hat{q}_j\ell(w(\tilde{x}_j),\tilde{y}_j) \tag{52}$$

$$\leq \alpha\mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\hat{\boldsymbol{q}}}_{\tilde{\boldsymbol{z}}}) + \sum_{j=1}^{n_t}\hat{q}_j\left(\ell(w(\tilde{x}_j),w'(\tilde{x}_j)) + \ell(w'(\tilde{x}_j),\tilde{y}_j)\right) \tag{53}$$

$$= \alpha\mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\hat{\boldsymbol{q}}}_{\tilde{\boldsymbol{z}}}) + \sum_{i=1}^{n_s}\hat{p}_i\ell(w'(x_i),y_i) + \sum_{j=1}^{n_t}\hat{q}_j\ell(w'(\tilde{x}_j),\tilde{y}_j) + \sum_{j=1}^{n_t}\hat{q}_j\ell(w(\tilde{x}_j),w'(\tilde{x}_j)) - \sum_{i=1}^{n_s}\hat{p}_i\ell(w'(x_i),y_i). \tag{54}$$

Here, (52) follows from (49); in (53) we used that the weights $\{\hat{q}_j\}$ are nonnegative as well as triangle inequality; to obtain (54) we just summed and subtracted the term $\sum_{i=1}^{n_s}\hat{p}_i\ell(w'(x_i),y_i)$. We now focus on the last two terms of (54). Let

$\hat{\Pi}^\star$ be the coupling matrix achieving $\mathbb{PW}_\alpha\left(\frac{1}{\beta}P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right)$. We have

$$\sum_{j=1}^{n_t} \hat{q}_j \ell(w(\tilde{x}_j), w'(\tilde{x}_j)) - \sum_{i=1}^{n_s} \hat{p}_i \ell(w'(x_i), y_i) = \sum_{j=1}^{n_t} \ell(w(\tilde{x}_j), w'(\tilde{x}_j)) \sum_{i=1}^{n_s} \hat{\Pi}^\star_{ij} - \sum_{i=1}^{n_s} \ell(w'(x_i), y_i) \sum_{j=1}^{n_t} \hat{\Pi}^\star_{ij} \tag{55}$$

$$= \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \hat{\Pi}^\star_{ij} \left(\ell(w(\tilde{x}_j), w'(\tilde{x}_j)) - \ell(w'(x_i), y_i)\right) \tag{56}$$

$$\leq \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \hat{\Pi}^\star_{ij} \left|\ell(w(\tilde{x}_j), w'(\tilde{x}_j)) - \ell(w'(x_i), y_i)\right| \tag{57}$$

$$\leq \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \hat{\Pi}^\star_{ij} \Big[\left|\ell(w(\tilde{x}_j), w'(\tilde{x}_j)) - \ell(w(\tilde{x}_j), w'(x_i))\right| \tag{58}$$

$$+ \left|\ell(w(\tilde{x}_j), w'(x_i)) - \ell(w'(x_i), y_i)\right| \Big]$$

$$\leq \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \hat{\Pi}^\star_{ij} \left[\zeta \left|w'(\tilde{x}_j) - w'(x_i)\right| + \ell(w(\tilde{x}_j), y_i)\right] \tag{59}$$

$$\leq \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \hat{\Pi}^\star_{ij} \left[\zeta\gamma \left\|f(\tilde{x}_j) - f(x_i)\right\| + \ell(w(\tilde{x}_j), y_i)\right] \tag{60}$$

$$= \mathbb{PW}_\alpha\left(\frac{1}{\beta}P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right). \tag{61}$$

Here, (55) follows from the definitions of $\hat{p}_i$ in (11) and $\hat{q}_j$ in (12); (59) follows because the loss is ζ-Lipschitz and because of the reverse triangle inequality (23); and (60) follows since g' is γ-Lipschitz with respect to the Euclidean distance.

By substituting (61) into (54) and decomposing w' as $w' = g' \circ f$, we obtain

$$L_{\tilde{\boldsymbol{z}}}(w) \leq \frac{1}{\alpha}\mathbb{PW}_\alpha\left(\frac{1}{\beta}P^f_{\boldsymbol{z}}, Q^w_{\boldsymbol{t}}\right) + \mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\hat{\boldsymbol{q}}}_{\tilde{\boldsymbol{z}}}) + \sum_{i=1}^{n_s} \frac{\hat{p}_i}{\alpha}\ell(g'(f(x_i)), y_i) + \sum_{j=1}^{n_t} \frac{\hat{q}_j}{\alpha}\ell(g'(f(\tilde{x}_j)), \tilde{y}_j)\,. \tag{62}$$

Next, we define

$$\Xi = \min_{g'\in\mathcal{G}}\left\{\sum_{i=1}^{n_s} \frac{\hat{p}_i}{\alpha}\ell(g'(f(x_i)), y_i) + \sum_{j=1}^{n_t} \frac{\hat{q}_j}{\alpha}\ell(g'(f(\tilde{x}_j)), \tilde{y}_j)\right\}$$

$$- \left(\min_{g'\in\mathcal{G}}\left\{\sum_{i=1}^{n_s} \frac{\hat{p}_i}{\alpha}\ell(g'(f(x_i)), y_i)\right\} + \min_{g'\in\mathcal{G}}\left\{\sum_{j=1}^{n_t} \frac{\hat{q}_j}{\alpha}\ell(g'(f(\tilde{x}_j)), \tilde{y}_j)\right\}\right). \tag{63}$$

We now minimize over g' in the two summations of (62), and note that

$$\min_{g'\in\mathcal{G}}\left\{\sum_{i=1}^{n_s} \frac{\hat{p}_i}{\alpha}\ell(g'(f(x_i)), y_i) + \sum_{j=1}^{n_t} \frac{\hat{q}_j}{\alpha}\ell(g'(f(\tilde{x}_j)), \tilde{y}_j)\right\}$$

$$\leq \sum_{i=1}^{n_s} \frac{\hat{p}_i}{\alpha}\ell(g(f(x_i)), y_i) + \min_{g'\in\mathcal{G}}\left\{\sum_{j=1}^{n_t} \frac{\hat{q}_j}{\alpha}\ell(g'(f(\tilde{x}_j)), \tilde{y}_j)\right\} + \Xi. \tag{64}$$

We obtain the desired result by recalling that $g(f(x_i)) = w(x_i)$, by using the definition of $\hat{L}_f$, and by writing out the explicit form of $\mathrm{TV}(Q_{\tilde{\boldsymbol{z}}}, Q^{\hat{\boldsymbol{q}}}_{\tilde{\boldsymbol{z}}})$, as per (50).

D. Proof of Lemma 3.4

The proof follows that of Alquier (2024, Thm. 2.1), with adjustments to account for the non-standard posterior and the use of $R(W, \boldsymbol{Z}, \tilde{\boldsymbol{Z}})$ in place of the empirical loss.

For a fixed $w \in \mathcal{W}$, we let $U_j^{(w)} = L_{Q_{\tilde{Z}}}(w) - \ell(w(\tilde{X}_j), \tilde{Y}_j)$. Note that the $\ell(w(\tilde{X}_j), \tilde{Y}_j)$ are supported on an interval of width 1, are independent and identically distributed, and satisfy $\mathbb{E}[\ell(w(\tilde{X}_j), \tilde{Y}_j)] = L_{Q_{\tilde{Z}}}(w)$. Hence, we can apply Hoeffding's inequality (Wainwright, 2019, Prop. 2.5) to find that, for every $t > 0$,

$$\mathbb{E}_{\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}} \left[e^{t \sum_{j=1}^{n_t} U_j^{(w)}} \right] \leq e^{\frac{t^2 n_t}{8}}. \tag{65}$$

Now, note that $\sum_{j=1}^{n_t} U_j^{(w)} = n_t (L_{Q_{\tilde{Z}}}(w) - L_{\tilde{\boldsymbol{Z}}}(w))$. Hence, setting $t = \lambda / n_t$ for some $\lambda > 0$, we have

$$\mathbb{E}_{\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}} \left[e^{\lambda \left(L_{Q_{\tilde{Z}}}(w) - L_{\tilde{\boldsymbol{Z}}}(w) \right)} \right] \leq e^{\frac{\lambda^2}{8 n_t}} \tag{66}$$

Now, it follows from the upper bound in (14) that

$$\mathbb{E}_{\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}} \left[e^{\lambda \left(L_{Q_{\tilde{Z}}}(w) - R(w, \boldsymbol{z}, \tilde{\boldsymbol{Z}}) \right)} \right] \leq \mathbb{E}_{\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}} \left[e^{\lambda \left(L_{Q_{\tilde{Z}}}(w) - L_{\tilde{\boldsymbol{Z}}}(w) \right)} \right]. \tag{67}$$

By combining (66) and (67) and averaging over $\boldsymbol{Z} \sim P_Z^{n_s}$ and $W \sim Q_W$ we find that, collecting factors on the left-hand side,

$$\mathbb{E}_{\boldsymbol{Z} \sim P_Z^{n_s}, \tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}, W \sim Q_W} \left[e^{\lambda \left(L_{Q_{\tilde{Z}}}(W) - R(W, \boldsymbol{Z}, \tilde{\boldsymbol{Z}}) \right) - \frac{\lambda^2}{8 n_t}} \right] \leq 1. \tag{68}$$

In the remainder of the proof, we will suppress the explicit distributions in the expectation notation when they are clear from context. We now apply the Donsker-Varadhan variational formula (Alquier, 2024, Lemma 2.2) to conclude that, for a given posterior $P_{W|\boldsymbol{Z},\boldsymbol{T}}$,

$$\mathbb{E}_{\boldsymbol{Z}, \tilde{\boldsymbol{Z}}} \left[e^{\lambda \mathbb{E}_{W \sim P_{W|\boldsymbol{Z},\boldsymbol{T}}} [L_{Q_{\tilde{Z}}}(W) - R(W, \boldsymbol{Z}, \tilde{\boldsymbol{Z}})] - D_{\mathrm{KL}}(P_{W|\boldsymbol{Z},\boldsymbol{T}} \,||\, Q_W) - \frac{\lambda^2}{8 n_t}} \right] \leq 1. \tag{69}$$

Next, by the Chernoff bound (Wainwright, 2019, Eq. (2.5)), we have that for every $s > 0$,

$$\begin{aligned}
&\mathbb{P}_{\boldsymbol{Z}, \tilde{\boldsymbol{Z}}} \left[\lambda \mathbb{E}_{W \sim P_{W|\boldsymbol{Z},\boldsymbol{T}}} [L_{Q_{\tilde{Z}}}(W) - R(W, \boldsymbol{Z}, \tilde{\boldsymbol{Z}})] - D_{\mathrm{KL}}(P_{W|\boldsymbol{Z},\boldsymbol{T}} \,||\, Q_W) - \frac{\lambda^2}{8 n_t} > s \right] \\
&\leq \mathbb{E}_{\boldsymbol{Z}, \tilde{\boldsymbol{Z}}} \left[e^{\lambda \mathbb{E}_{W \sim P_{W|\boldsymbol{Z},\boldsymbol{T}}} [L_{Q_{\tilde{Z}}}(W) - R(W, \boldsymbol{Z}, \tilde{\boldsymbol{Z}})] - D_{\mathrm{KL}}(P_{W|\boldsymbol{Z},\boldsymbol{T}} \,||\, Q_W) - \frac{\lambda^2}{8 n_t}} \right] e^{-s} \qquad (70) \\
&\leq e^{-s}. \qquad (71)
\end{aligned}$$

Setting $s = \log(1/\delta)$, we thus conclude that with probability at most δ over $\boldsymbol{Z} \sim P_Z^{n_s}$, $\tilde{\boldsymbol{Z}} \sim Q_{\tilde{Z}}^{n_t}$,

$$\lambda \mathbb{E}_{W \sim P_{W|\boldsymbol{Z},\boldsymbol{T}}} [L_{Q_{\tilde{Z}}}(W) - R(W, \boldsymbol{Z}, \tilde{\boldsymbol{Z}})] - D_{\mathrm{KL}}(P_{W|\boldsymbol{Z},\boldsymbol{T}} \,||\, Q_W) - \frac{\lambda^2}{8 n_t} > \log \frac{1}{\delta}. \tag{72}$$

We obtain the desired result by considering the complementary event and re-arranging terms.

E. Additional Details on the Experiments

We compute the domain alignment term $\mathbb{PW}_\alpha \left(\frac{1}{\beta} P_{\boldsymbol{z}}^f, Q_{\boldsymbol{t}}^w \right)$ in (19) using the entropic partial Wasserstein solver from the POT library (Flamary et al., 2021), with regularization constant $\varepsilon = 7.0$ in all the experiments, which is selected to avoid numerical instabilities. We set the maximum number of iterations to 5000. Following Nguyen et al. (2022), we linearly increase α from 0.01 to α_{max} for the first 2500 iterations, and keep it constant for the last 2500 iterations. Through a parameter search, we obtained the following values for the hyperparameters: $\alpha_{\text{max}} = 0.8, \eta_1 = 0.125, \eta_2 = 1.75, \beta = 0.35$. We use a batch size of 65, and set the learning rate of stochastic gradient descent to 0.001. We used the same values for these hyperparameters in all our experiments. For ARPM+our-weights, we set $1/\beta = 3$, while all other hyperparameters are the same as those in ARPM (Gu et al., 2024).

Table 3. Test accuracy on the ImageNet → Caltech dataset using the weight choices described in Section 5.3.

Weighting scheme	Test accuracy
MPOT weights	78.6 (1.2)
BA^{3}US weights	84.7 (0.7)
ARPM weights	79.2 (1.4)
WARMPOT (ours)	**84.8** (0.1)

Table 4. Test accuracy on the ImageNet → Caltech dataset.

Algorithm	Test accuracy
ResNet-50 (He et al., 2016)	69.7
DAN (Long et al., 2015)	71.3
DANN (Ganin et al., 2016)	70.8
IWAN (Zhang et al., 2018)	78.1
PADA (Cao et al., 2018)	75.0
ETN (Cao et al., 2019)	83.2
DRCN (Li et al., 2020)	75.3
BA^{3}US (Liang et al., 2020)	84.0
ISRA+BA^{3}US (Xiao et al., 2021)	85.3
SLM (Sahoo et al., 2023)	82.3
SAN++ (Cao et al., 2022)	83.3
AR (Gu et al., 2021)	85.4 (0.2)
ARPM (Gu et al., 2024)	84.1 (1.4)
PWAN (Wang et al., 2024)	**86.0 (0.5)**
WARMPOT (ours)	84.8 (0.1)
ARPM+our-weights	85.1 (0.9)

F. Additional Numerical Results

In this section, we discuss the results obtained by repeating the experiments described in Section 5.3 on the ImageNet → Caltech dataset, where ImageNet (Russakovsky et al., 2015) consists of 1000 classes and Caltech-256 (Griffin et al., 2022) consists of 256 classes.

We first compare different weighting schemes. Following Gu et al. (2024), we set the weight update intervals of ARPM and BA^{3}US to 2000. The experiment is repeated for 3 random seeds, and we report the average and the standard deviation. The results are presented in Table 3. WARMPOT results in better performance than MPOT and ARPM, and yields performance comparable to BA^{3}US.

Then we compare WARMPOT against alternative algorithms. We set $\alpha_{\max} = 0.08$, $\eta_1 = 0.92$, $\eta_2 = 5.47$, $\beta = 0.72$, $\varepsilon = 5.59$ for WARMPOT. The results are shown in Table 4. Once more, we observe that ARPM+our-weights achieves better performance than ARPM, highlighting the effectiveness of the WARMPOT weights.

G. Sensitivity Analysis for Alignment Parameters

In order to assess the impact on performance of the hyperparameters in the domain alignment term, we conduct a sensitivity analysis on $\alpha_{\max}$, β, on the ImageNet → Caltech dataset. In this analysis, we set the following values for the other hyperparameters: $\eta_1 = 0.92$, $\eta_2 = 5.47$, $\varepsilon = 5.59$. In the experiment on $\alpha_{\max}$, we set $\beta = 0.72$, while in the experiment on β, we set $\alpha_{\max} = 0.08$.

As seen in Fig. 2 (right), varying β over the entire range $(0, 1]$ has a limited impact on performance, with variations not exceeding 2%. We observe a similar trend whenever $\alpha_{\max}$ is varied within the range $(0, 0.1]$. When $\alpha_{\max} > 0.1$, however, we see a significant drop in performance, caused most likely by the large number of outliers in the source sample. These

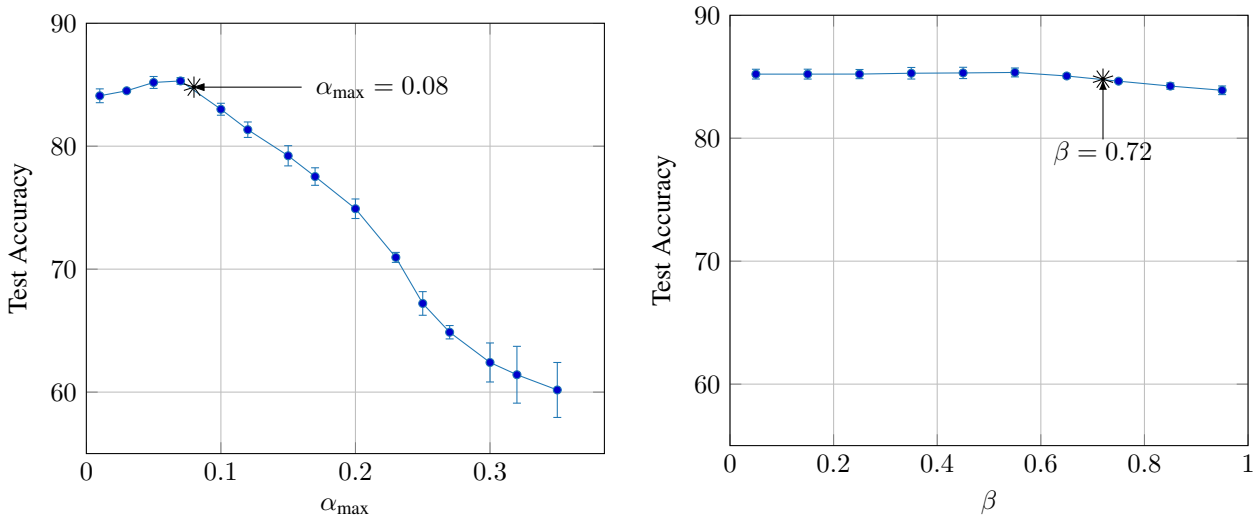

Figure 2. The effect of changing alignment parameters $\alpha_{\max}$ and β on test accuracy of ImageNet $\rightarrow$ Caltech.

results indicate that the specific choice of these parameters has a minor impact over a range of reasonable values.

H. Relation between WARMPOT and ARPM Weights

The weights $\hat{\boldsymbol{p}} = (\hat{p}_1, \ldots, \hat{p}_{n_s})$ used in the ARPM algorithm of Gu et al. (2024) are defined as the solution of the following Wasserstein-1 type problem between the source and target distributions:

$$\min_{\hat{\boldsymbol{p}} \in \Delta} \mathbb{W}_1 \left(\sum_{i=1}^{n_s} \hat{p}_i \delta_{f(x_i)}, \frac{1}{n_t} \sum_{j=1}^{n_t} \delta_{f(\tilde{x}_j)} \right). \tag{73}$$

Here, the constraint set Δ is defined as

$$\Delta = \begin{cases} \hat{p}_i \geq 0 & \text{(74a)} \\ \sum_{i=1}^{n_s} \hat{p}_i = 1 & \text{(74b)} \\ \sum_{i=1}^{n_s} \left(\hat{p}_i - \frac{1}{n_s} \right)^2 \leq \frac{\rho}{n_s} & \text{(74c)} \end{cases}$$

where ρ is a hyperparameter. Interestingly, the weights $\hat{\boldsymbol{p}} = (\hat{p}_1, \ldots, \hat{p}_{n_s})$ of WARMPOT, given in (11), can be equivalently expressed, for the case $\alpha = 1$, as the solution of the same optimization problem as in (73), but with the constraint set Δ replaced by Γ defined as:

$$\Gamma = \begin{cases} \hat{p}_i \geq 0 & \text{(75a)} \\ \sum_{i=1}^{n_s} \hat{p}_i = 1 & \text{(75b)} \\ \hat{p}_i \leq \frac{1}{\beta n_s}. & \text{(75c)} \end{cases}$$

Here, β is a hyperparameter. Note that the only difference between the two sets of constraints is that they use different ways to control the magnitude of $\hat{p}_i$ in (74c) and (75c). Compared to the ARPM constraint Δ, our constraint Γ is not only simpler and more theoretically grounded, but also more intuitive: (75c) controls the maximum target sample mass that can be matched to a single source sample, while the corresponding ARPM constraint (74c) is harder to interpret. As demonstrated in our numerical results in Table 2, our weights also lead to better performance.

Note finally that ARPM solves (73) only approximately: it first solves a Wasserstein-1 problem with fixed $\hat{p}_i = 1/n_s$ using a Wasserstein-GAN, and then fixes this learned Wasserstein-GAN and updates the weights $\hat{p}_i$.