# Rethinking Cross-Modal Fine-Tuning: Optimizing the Interaction Between Feature Alignment and Target Fitting

**Trong Khiem Tran**[1,3], **Manh Cuong Dao**[2], **Phi Le Nguyen**[3]
**Thao Nguyen Truong**[4], **Trong Nghia Hoang**[1†]

[1]Washington State University, [2]National University of Singapore
[4]National Institute of Advanced Industrial Science and Technology
[3]Hanoi University of Science and Technology, [†]Corresponding author

## Abstract

Adapting pre-trained models to unseen feature modalities has become increasingly important due to the growing need for cross-disciplinary knowledge integration. A key challenge here is how to align the representation of new modalities with the most relevant parts of the pre-trained model's representation space to enable accurate knowledge transfer. This requires combining feature alignment with target fine-tuning, but uncalibrated combinations can exacerbate misalignment between the source and target feature-label structures and reduce target generalization. Existing work however lacks a theoretical understanding of this critical interaction between feature alignment and target fitting. To bridge this gap, we develop a principled framework that establishes a provable generalization bound on the target error, which explains the interaction between feature alignment and target fitting through a novel concept of feature-label distortion. This bound offers actionable insights into how this interaction should be optimized for practical algorithm design. The resulting approach achieves significantly improved performance over state-of-the-art methods across a wide range of benchmark datasets.

## 1 INTRODUCTION

Modern applications increasingly require transferring representational knowledge from pre-trained foundation models (FMs) to new data modalities. This allows downstream tasks to benefit from the representational knowledge embedded in the pre-trained model when the pre-training data modalities contain relevant information. For example, in genomics, gene expression profiles can be leveraged to enrich the representation for tissue image data (Lin et al., 2024). Likewise, existing models pre-trained on broad vision (Liu et al., 2021b), language (Liu et al., 2019), or speech corpora (Radford et al., 2022) have been increasingly leveraged in domains with new data modalities not seen during pre-training. Recent studies have also shown promising transfers of vision and language models to modalities such as protein structures, cosmic ray signals, and human gestures (Shen et al., 2023).

These early results highlight the potential of cross-modal adaptation and underscore the need for more principled and generalizable approaches. Since a pre-trained FM is optimized to extract the most predictive patterns within its original representation space, positive knowledge transfer requires mapping new data modalities into the most relevant regions of that space. This raises a fundamental challenge:

**How can we translate new data featuring unseen modalities into an existing pre-trained representation space to enable effective cross-modal knowledge transfer?**

Addressing this challenge is non-trivial, since unlike in-modal fine-tuning, the source and target data distributions often have different statistical structures. Even when the source and target modalities are encoded into the same dimensional space, their resulting feature distributions can differ in covariance structure, higher-order interactions among covariates, and mode geometries. As the pre-trained FM implicitly leverages such distributional information to identify predictive patterns, a mismatch between the source and target representation distributions may cause it to activate spurious or irrelevant patterns, leading to negative transfer. Aligning the representation of new data modalities with relevant distributional geometries of the pre-trained representation space is therefore critical to ensure positive transfer (see Fig. 2).

Proceedings of the 29[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300. 

On the other hand, the pre-trained representation space often contains a broad landscape of heterogeneous geometries centered around different modes, many of which may be irrelevant or poorly aligned with the target task. Thus, to avoid negative transfer that harms performance, feature alignment must also be guided by target fitting using fine-tuning data. This guided alignment is non-trivial to formalize, as the interaction between feature alignment and target fitting and its effect on target generalization is not well understood. Existing work addressing this challenge has largely focused on heuristic combinations of feature alignment and target fitting, without providing guarantees on generalization performance for the (downstream) target task (Shen et al., 2023; Cai et al., 2024; Ma et al., 2024).

Notably, ORCA (Shen et al., 2023) aligns source and target representation distributions using optimal transport before fine-tuning the entire network, while PARE (Cai et al., 2024) introduces a gating mechanism to combine source and target features during fine-tuning. These methods have demonstrated promising empirical results on benchmarks such as NasBench360 (Tu et al., 2022), but depend on heuristic formulations without explicitly modeling the interaction between feature alignment and target fitting. MoNA (Ma et al., 2024) characterizes this interaction through a bi-level optimization framework. The inner loop identifies the target-optimal predictor for a given feature embedder, while the outer loop updates the embedder so that its combined representations and predictions align with the source task's feature-label semantics. This approach aims to reduce the misalignment between the source and target feature-label semantics under a candidate target representation using heuristic alignment measures, while fitting to target data. However, despite its empirical gains, the reliance on heuristic metrics and bi-level design leaves the theoretical link to optimal generalization on the target task unexamined. It remains unclear whether this characterization captures the most effective interaction between feature alignment and target fitting for cross-modal knowledge transfer.

To bridge this gap, we introduce a principled framework that establishes a provable bound on the generalized target error, capturing the interaction between feature alignment and target fitting via a **feature-label distortion** concept. This distortion quantifies the complexity of the probabilistic transport map between the source and target feature-label predictive distributions under a given target feature representation. Intuitively, it measures the cross-modal transferability under a given target representation. A large distortion means low transferability which will cause target fitting to overfit when fine-tuning data is limited, thus decreasing generalized performance. This provides actionable insights into how this interaction should be optimized, offering a theoretically grounded guide for algorithm design. The above is substantiated with the following technical contributions:

**Theoretical Analysis.** We develop a theoretical bound that decomposes the generalized target error into: (i) the source task error, which serves as a fixed overhead; (ii) a distributional distance between the source and target representation distributions (i.e., **feature alignment**); (iii) the minimum entropy over a space of probabilistic transport maps between the source and target feature-label conditional distributions (i.e., **feature-label distortion**); and (iv) an alignment term that reflects how well the target predictor follows this transport (i.e., **target fitting**). To the best of our knowledge, this is the first generalization bound that captures the influence of both the pre-trained model's quality and the interaction between feature alignment and target fitting on cross-modal fine-tuning performance (Section 2).

**Algorithm Design.** We develop a practical algorithm to address the intractability of optimizing the (ii) feature alignment and (iii) feature-label distortion terms over the space of target feature representations and the induced probabilistic transport plans between the source and target feature-label conditional distributions while performing (iv) target fitting. Our approach constructs an optimizable surrogate that serves as a medium for selectively transferring source knowledge relevant to the target task. This surrogate is optimized in a preparatory stage to guide the initialization of the main fine-tuning procedure, enabling efficient and targeted cross-modal adaptation (Section 3).

**Evaluation.** We evaluate our approach on two comprehensive cross-modal fine-tuning benchmarks: (1) NAS-Bench-360 (Tu et al., 2022), which covers a broad set of tasks across ten distinct data modalities; and (2) PDEBench (Takamoto et al., 2022), which assesses model adaptation to simulated data derived from diverse families of partial differential equations (PDEs). Across both benchmarks, our method consistently outperforms recent state-of-the-art baselines, including ORCA (Shen et al., 2023), PARE (Cai et al., 2024), and MoNA (Ma et al., 2024), on a significant majority of tasks. These results underscore the importance of designing algorithms guided by an explicit generalization bound framework (Section 4).

## 2 THEORETICAL ANALYSIS

This section presents our main theoretical result. We begin by formalizing the cross-modal fine-tuning problem setting and introducing the key notations. We then establish a generalization bound that characterizes how the interaction between feature alignment and target fitting, under a given target feature representation, affects the generalized target performance. This provides a principled foundation for understanding and optimizing cross-modal adaptation (Section 3).

### 2.1 Problem Setting and Notations

Let $M_s \triangleq (\theta, p_s(z \mid \theta(\boldsymbol{x})))$ denote the learned embedder $\theta$ and the prediction map $p_s(z \mid \theta(\boldsymbol{x}))$ of an FM pre-trained on a source dataset $(\boldsymbol{X}, \boldsymbol{z}) = \{(\boldsymbol{x}_i, z_i)\}_{i=1}^n \sim D_s(\boldsymbol{x}, z)$. During fine-tuning, $(\boldsymbol{X}, \boldsymbol{z})$ might not be accessible, but we assume access to an in-modal proxy dataset $(\boldsymbol{X}^s, \boldsymbol{z}^s) = \{\boldsymbol{x}_i^s, z_i^s\}_{i=1}^m \sim D_s(\boldsymbol{x}, z)$ which is sampled from the same distribution.

Let $(\boldsymbol{X}^\tau, \boldsymbol{z}^\tau) = \{\boldsymbol{x}_i^\tau, z_i^\tau\}_{i=1}^\kappa \sim D_\tau(\boldsymbol{x}', z')$ denote the target fine-tuning dataset sampled from another data distribution $D_\tau$ over unseen modalities $\boldsymbol{x}'$ and label $z'$. We want to construct a target model $M_\tau \triangleq (\phi, p_\tau(z' \mid \phi(\boldsymbol{x}')))$ based on $M_s$ and $(\boldsymbol{X}^\tau, \boldsymbol{z}^\tau) \sim D_\tau(\boldsymbol{x}', z')$ in a principled manner.

To achieve this, we will establish a mathematical connection (see Theorem 3) between the generalized source/target losses (see Definition 1) and the alignment of their feature distributions (see Definition 2) under feature maps $\theta$ and $\phi$.

**Definition 1 (Generalized Error)** *The generalized source/target errors under feature maps $\theta/\phi$ are:*

$$\mathrm{err}_s(\theta) \triangleq -\mathbb{E}_{(\boldsymbol{x},z)\sim D_s}\Big[\log p_s\big(z \mid \theta(\boldsymbol{x})\big)\Big], \quad (1)$$

$$\mathrm{err}_\tau(\phi) \triangleq -\mathbb{E}_{(\boldsymbol{x}',z')\sim D_\tau}\Big[\log p_\tau\big(z' \mid \phi(\boldsymbol{x}')\big)\Big], \quad (2)$$

*which are the expected source and target prediction losses at $(\boldsymbol{x}, z) \sim D_s$ and $(\boldsymbol{x}', z') \sim D_\tau$.*

**Definition 2 (Feature Distribution)** *The feature distributions $D_s^\theta(\boldsymbol{u})/D_\tau^\phi(\boldsymbol{u})$ are defined as the push-forward of the source's and target's marginal input distributions $D_s(\boldsymbol{x})/D_\tau(\boldsymbol{x}')$ under the source and target feature maps, $\boldsymbol{u} = \theta(\boldsymbol{x})$ and $\boldsymbol{u} = \phi(\boldsymbol{x}')$, respectively.*

We also use $D_s^\theta(z \mid \boldsymbol{u})$ and $D_\tau^\phi(z' \mid \boldsymbol{u})$ to denote the source's and target's feature-label conditionals induced from the data distributions $D_s(\boldsymbol{x}, z)$ and $D_\tau(\boldsymbol{x}', z')$ under the source feature map $\boldsymbol{u} = \theta(\boldsymbol{x})$ and the target feature map $\boldsymbol{u} = \phi(\boldsymbol{x}')$, respectively.

### 2.2 Main Result

Our main result characterizes the generalized target loss $\mathrm{err}_\tau(\phi)$ in terms of the following key quantities:

**Overhead.** The generalized source loss $\mathrm{err}_s(\theta)$.

**Feature Alignment (FA).** A function of the distributional distance between source and target distributions, $D_s^\theta(\boldsymbol{u})$ and $D_\tau^\phi(\boldsymbol{u})$, over feature map $\theta$ and $\phi$. (see Definition 4).

**Feature Label Distortion (FLD).** A minimum entropy of a valid transport plan $\Lambda_{\boldsymbol{u}}^*(z' \mid z)$ over label pairs $(z', z)$ that maps the source conditional $D_s^\theta(z \mid \boldsymbol{u})$ to the target conditional $D_\tau^\phi(z' \mid \boldsymbol{u})$ over the representation $\boldsymbol{u} = \phi(\boldsymbol{x}')$ produced by the target feature map $\phi$ (see Definition 5).

**Target Fitting (FT).** A prediction alignment between the target predictor $p_\tau(z' \mid \boldsymbol{u} = \phi(\boldsymbol{x}'))$ and oracle predictor $D_\tau^\phi(z' \mid \boldsymbol{u} = \phi(\boldsymbol{x}'))$ under feature map $\phi$ over the presentation $\boldsymbol{u} = \phi(\boldsymbol{x}')$(see Definition 6).

An informal statement of our result is stated below.

**Theorem 3 (Informal Statement)** *Under arbitrary feature map $\theta$ and $\phi$, we have:*

$$\mathrm{err}_\tau(\phi) \le \mathrm{err}_s(\theta) + \textbf{\textit{Feature-Label Distortion}} + \textbf{\textit{Feature Alignment}} + \textbf{\textit{Target Fitting}}. \quad (3)$$

This result reveals how the pre-trained model's quality and the interplay between feature alignment and target fitting shape the cross-modal fine-tuning performance via a feature-label distortion measurement (Definition 5). The source loss acts as a fixed overhead, reflecting the influence of the source model. For FMs, this loss is often negligible. The remaining terms suggest that aligning features without careful calibration may inadvertently increase the semantic gap between source and target feature-label structures. For instance, when alignment induces representations that enlarge this gap, it can harm generalization by steering target fitting toward overfitting the prediction map to the target data in order to compensate for the poorly aligned representation structure.

This insight is made precise via the below formal definitions and theorem statement (see Theorem 7).

**Definition 4 (Feature Alignment)** *Let $\Delta$ denote the set of cost metrics $\delta$ (on the pre-trained representation space) such that the cross-entropy of the source prediction $\ell_s(\boldsymbol{u}) \triangleq -\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})} \log p_s(z \mid \boldsymbol{u})$ is $\tau_\delta$-Lipschitz with $\delta$:*

$$\big|\ell_s(\boldsymbol{u}_1) - \ell_s(\boldsymbol{u}_2)\big| \le \tau_\delta \cdot \delta\big(\boldsymbol{u}_1, \boldsymbol{u}_2\big). \quad (4)$$

*The feature alignment under target feature map $\phi$ is*

$$\boldsymbol{FA}(\phi,\theta) \triangleq \min_{\delta\in\Delta}\left\{\tau_\delta \cdot W_\delta\left(D_\tau^\phi(\boldsymbol{u}), D_s^\theta(\boldsymbol{u})\right)\right\}. \quad (5)$$

*where $W_\delta$ is Wasserstein-1 distance with cost metric $\delta$ (see definition in Appendix D.1).*

**Definition 5 (Feature-Label Distortion)** *Let $C_{\boldsymbol{u}}^*$ denote the set of valid transport plans that satisfy*

$$D_\tau^\phi(z' \mid \boldsymbol{u}) = \mathbb{E}_z\left[\Lambda_{\boldsymbol{u}}^*(z' \mid z)\right] \text{ with } z \sim D_s^\theta(z \mid \boldsymbol{u}). \quad (6)$$

*The feature-label distortion at representation $\boldsymbol{u}$ is*

$$\boldsymbol{FLD}(\boldsymbol{u}) \triangleq \min_{\Lambda_{\boldsymbol{u}}^*\in C_{\boldsymbol{u}}^*} \mathbb{E}_z\left[\mathbb{H}\left[\Lambda_{\boldsymbol{u}}^*(z' \mid z)\right]\right]. \quad (7)$$

*with $z \sim D_s^\theta(z \mid \boldsymbol{u})$. This characterizes the transferability of source knowledge $\theta$ to target task $\phi$ at $\boldsymbol{u}$.*

**Definition 6 (Target Fitting)** *Let $C_{\boldsymbol{u}}$ denote the set of valid transport plans $\Lambda_{\boldsymbol{u}}(z' \mid z)$ that satisfy:*

$$p_\tau(z' \mid \boldsymbol{u}) = \mathbb{E}_z\left[\Lambda_{\boldsymbol{u}}(z' \mid z)\right] \text{ with } z \sim D_s^\theta(z \mid \boldsymbol{u}). \quad (8)$$

*The alignment of the target prediction map $p_\tau(z' \mid \boldsymbol{u})$ with the oracle predictor $D_\tau^\phi(z' \mid \boldsymbol{u})$ at $\boldsymbol{u} = \phi(\boldsymbol{x}')$ is*

$$\mathbf{TF}(\boldsymbol{u}) \triangleq \min_{\Lambda_{\boldsymbol{u}}\in C_{\boldsymbol{u}}} \mathbb{E}_z\left[\mathbb{KL}\left(\Lambda_{\boldsymbol{u}}^+(.\mid z)\|\Lambda_{\boldsymbol{u}}(.\mid z)\right)\right] \text{ where}$$
$$\Lambda_{\boldsymbol{u}}^+(.\mid .) = \arg\min_{\Lambda_{\boldsymbol{u}}^*\in C_{\boldsymbol{u}}^*} \mathbb{E}_z\left[\mathbb{H}\left[\Lambda_{\boldsymbol{u}}^*(z' \mid z)\right]\right]. \quad (9)$$

with $z \sim D_s^\theta(z \mid \boldsymbol{u})$. Theorem 3 is formally stated as:

**Theorem 7 (Formal Statement)** *Given the above technical specification of feature alignment, feature-label distortion, and target fitting, we have*

$$\mathrm{err}_\tau(\phi) \le \mathrm{err}_s(\theta) + \boldsymbol{FA}(\phi,\theta) + \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\left[\boldsymbol{FLD}(\boldsymbol{u}) + \boldsymbol{TF}(\boldsymbol{u})\right]. \quad (10)$$

*A detailed proof is provided in Appendix A. Our empirical inspection in Fig. 5, Appendix B further shows that the bound is sufficiently tight.*

Theorem 7 operationalizes the earlier intuition by providing a complete algorithmic structure to measure the semantic gap in cross-modal fine-tuning under a candidate target representation, captured through both feature alignment and feature-label distortion. While prior work has relied on distributional alignment to support knowledge transfer into target fitting, the role of feature-label distortion in shaping transferability has been overlooked. This term exposes the representational discrepancy between the feature-label structures of the source and target domains. Lower feature-label distortion suggests that the target label can be more readily inferred from the source label information, establishing a consistent pattern that improves transferability from source to target. This insight reveals that minimizing feature alignment alone may not be sufficient for effective transfer. We will build on this to design an algorithm that incorporates source-informed regularization to steer away from representations that induce large semantic gap, guiding fine-tuning toward more transferable solutions (Section 3).

## 3 ALGORITHM DESIGN

This section introduces a new cross-modal fine-tuning algorithm named **RECRAFT** – **RE**thinking **CR**oss-Mod**A**l **F**ine-**T**uning – which is designed to bridge the semantic gap between source and target tasks in a cross-modal context by optimizing the interaction between feature alignment and target fitting. It is guided by the result of Theorem 7, which bounds the generalized target error by a combination of feature alignment (**FA**) in Eq. (5), target fitting (**TF**) in Eq. (9), as well as their interaction measured via feature-label distortion (**FLD**) in Eq. (7). RECRAFT optimizes this bound via a two-stage workflow as illustrated in Figure 1 below. Stage 1 learns the optimal target feature map $\phi$ via minimizing a combination of **FA** and **FLD** which defines the source-target semantic gap under $\phi$ (Section 3.1). Stage 2 optimizes a target prediction map to minimize the target fitting term **TF** based on the learned feature map $\phi$ (Section 3.2).

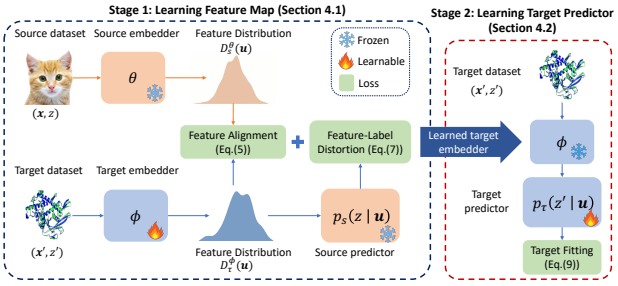


Figure 1: Overview of the RECRAFT algorithm.

To elaborate on this design, we note that a direct minimization of the theoretical bound in Eq. (10) of Theorem 7 is however unstable due to the entangled effect of optimizing both the target prediction map $p_\tau(z' \mid \boldsymbol{u})$ and feature map $\boldsymbol{u} = \phi(\boldsymbol{x}')$ on the oracle and learnable transport sets $C_{\boldsymbol{u}}^*$ (see Definition 5) and $C_{\boldsymbol{u}}$ (see Definition 6) in complex and interdependent ways. In particular, changes in the representation $\phi$ simultaneously alter the alignment between source and target feature distributions and reshape the transport land-

scape $C^*_{\boldsymbol{u}}$ on which the target predictor $p_\tau(z' \mid \boldsymbol{u})$ is optimized via minimizing **TF** in Eq. (9).

This coupling complicates optimization as it continually moves the optimization target for $p_\tau(z' \mid \boldsymbol{u})$ which destabilizes convergence. To mitigate this, we adopt a two-stage approach that decomposes the bound minimization into (1) finding a feature map $\phi$ that minimizes the semantic gap $\mathbf{FA}(\phi,\theta) + \mathbb{E}_{D^\phi_\tau(\boldsymbol{u})}[\mathbf{FLD}(\boldsymbol{u})]$ between the source and target task, thus maximizing transferability (Section 3.1); and (2) learning a predictor $p_\tau(z' \mid \boldsymbol{u})$ based on the learned $\phi$ (Section 3.2) via minimizing $\mathbb{E}_{D^\phi_\tau(\boldsymbol{u})}[\mathbf{TF}(\boldsymbol{u})]$. This decomposition stabilizes the optimization by removing interdependencies: the first stage depends only on $\phi$ while the second stage optimizes $p_\tau$ given $\phi$, avoiding a moving target. Figure 1 provides an overview of our algorithm. Its detailed pseudocode is provided in Appendix G.

### 3.1 Learning Feature Map

To minimize the semantic gap between source and target, we aim to solve the following:

$$\phi \;=\; \operatorname{argmin}_\phi \Big\{ \mathbf{FA}\big(\phi,\theta\big) + \mathbb{E}_{D^\phi_\tau(\boldsymbol{u})}\big[\mathbf{FLD}(\boldsymbol{u})\big] \Big\} \;. \quad (11)$$

As mentioned earlier, this reveals a more principled approach to conditioning the feature map. Prior work such as (Shen et al., 2023) often minimizes **FA** primarily to align the source and target representations, without accounting for its effect on feature-label distortion (**FLD**), which captures the semantic misalignment between the source and target feature-label structures induced by $\phi$. As **FLD** is incorporated into Eq. (11), our formulation discourages feature maps that increase semantic misalignment which would cause the target fitting stage to compensate improperly during training and reduce transfer performance. To minimize Eq. (11), we develop effective optimization surrogates for both **FA** and **FLD** as they are not directly tractable. This is detailed next.

**A. Feature Alignment Loss.** Following Definition 4,

$$\mathbf{FA}(\theta,\phi) \;\;=\;\; \min_{\delta\in\Delta} \Big\{ \tau_\delta W_\delta\big(D^\phi_\tau(\boldsymbol{u}), D^\theta_s(\boldsymbol{u})\big) \Big\} \;, \quad (12)$$

where $\Delta$ is the set of cost metrics $\delta$ for the Wasserstein distance $W_\delta$ such that the cross-entropy source prediction $\ell_s(\boldsymbol{u}) \triangleq -\mathbb{E}_{D^\theta_s(z|\boldsymbol{u})}[\log p_s(z \mid \boldsymbol{u})]$ is $\tau_\delta$-Lipschitz with respect to $\delta(\boldsymbol{u},\boldsymbol{u}')$ (see Definition 4).

To effectively constrain the search over $\Delta$ to metric regimes that impose low $\tau_\delta$ on $\ell_s(\boldsymbol{u})$, our main approach is to view $\tau_\delta = \omega$ as a hyperparameter to be determined using a proxy dataset $P_s = (\boldsymbol{X}^s, \boldsymbol{z}^s)$ of the source task. Given $\omega$, we will adapt the source prediction map $p_s(z \mid \boldsymbol{u}) \simeq p_s(z \mid \boldsymbol{u};\gamma)$ so that its imposed Lipschitz constant on $\ell_s(\boldsymbol{u})$ under feature map $\boldsymbol{u} = \theta(\boldsymbol{x})$ is around $O(\omega)$. This is achieved via

$$\text{minimize}\; \mathbb{E}_{\boldsymbol{x}\sim P_s} \max\Big(0, \big\|\nabla_{\boldsymbol{u}} \ell_s(\theta(\boldsymbol{x});\gamma)\big\|_\delta - \omega\Big)^2, \quad (13)$$

with respect to $\delta$ and $\omega$. Here, $\gamma$ can be selected as the parameters of the last layer in the pre-trained prediction map $p_s(z \mid \boldsymbol{u})$. This generalizes a prior practice on Lipschitz conditioning in (Shen et al., 2018). The hyperparameter $\omega$ can be selected to have smallest value without degrading the predictive performance of the source model on the proxy dataset $P_s$. In our experiment, this optimal value ranges between 0.3 and 0.5 across different source models under the Euclidean metric $\delta \equiv \ell_2$ (see Appendix F). Thus, the feature alignment (**FA**) can be surrogated with:

$$\mathbf{FA}(\theta,\phi) \;\simeq\; L_{\mathbf{FA}}(\phi) \triangleq \omega \cdot W_{\ell_2}\Big(D^\phi_\tau(\boldsymbol{u}), D^\theta_s(\boldsymbol{u})\Big) \;, \quad (14)$$

where $W_{\ell_2}$ is the Wasserstein-1 distance with norm-2 cost metric $\ell_2$. See Appendix D.2 for more details.

**B. Feature-Label Distortion Loss.** We will now construct a surrogate for feature-label distortion,

$$\begin{aligned}
\mathbb{E}_{D^\phi_\tau(\boldsymbol{u})}\Big[\mathbf{FLD}(\boldsymbol{u})\Big] &= \mathbb{E}_{(\boldsymbol{u})}\left[\min_{\Lambda^*_{\boldsymbol{u}}} \mathbb{E}_z\Big[\mathbb{H}\Big[\Lambda^*_{\boldsymbol{u}}(.\mid z)\Big]\Big]\right] && (15)\\
&\le \mathbb{H}_{(\phi,p_s)}\big[Z' \mid Z,\; \boldsymbol{U}\big] && (16)\\
&\le \mathbb{H}_{(\phi,p_s)}\big[Z' \mid Z\big] && (17)\\
&= \mathbb{H}_{(\phi,p_s)}[Z', Z] \;-\; \mathbb{H}_{(\phi,p_s)}[Z], && (18)
\end{aligned}$$

where the first inequality hold due to the minimum over the choice of the oracle transport $\Lambda^*_{\boldsymbol{u}}(z' \mid z)$ and the definition of conditional entropy. The second inequality holds due the information-never-hurt property of entropy (Cover and Thomas, 2006). The last equality holds from the chain rule. Eq. (18) allows us to bypass the inaccessible oracle transport $\Lambda_{\boldsymbol{u}}(z' \mid z)$ and estimate it using empirical methods (Nguyen et al., 2020). For each data point $(\boldsymbol{x}', z')$ in the target dataset, we can generate a pseudo source label for it via $z \sim P_s(z \mid \boldsymbol{u})$ under the target feature map $\boldsymbol{u} = \phi(\boldsymbol{x}')$. Using these statistics, we can approximate $P_{(\phi,p_s)}(z,z') \simeq C(z,z')/\kappa$ where $C(z,z')$ counts the number of times we observe $(z,z')$ via the above pseudo source simulation; and $\kappa$ is the number of target data points. Likewise, we estimate $P_{(\phi,p_s)}(z) \simeq \sum_{z'} P_{(\phi,p_s)}(z,z')$. This allows us to approximate

$$\begin{aligned}
\mathbb{H}_{(\phi,p_s)}[Z',Z] &= -\sum_z \sum_{z'} P_{(\phi,p_s)}\big(z,z'\big) \log P_{(\phi,p_s)}\big(z,z'\big)\\
\mathbb{H}_{(\phi,p_s)}[Z] &= -\sum_z P_{(\phi,p_s)}(z) \log P_{(\phi,p_s)}(z) \;. \quad (19)
\end{aligned}$$

The surrogate for **FLD** is thus defined as

$$\mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\Big[\mathbf{FLD}(\boldsymbol{u})\Big] \simeq L_{\mathbf{FLD}}(\phi) \triangleq \mathbb{H}_{(\phi,p_s)}\big[Z' \mid Z\big] \quad (20)$$
$$= \mathbb{H}_{(\phi,p_s)}[Z', Z] \; - \; \mathbb{H}_{(\phi,p_s)}[Z]. \quad (21)$$

Combining Eq. (14) and Eq. (21), we obtain the following surrogate for Eq. (11),

$$\phi \;=\; \text{argmin}_\phi\Big(L_{\mathbf{FA}}(\phi) \; + \; L_{\mathbf{FLD}}(\phi)\Big) \;, \quad (22)$$

which can be minimized effectively using standard numerical optimization method.

### 3.2 Learning Target Predictor

Given the learned target feature map $\phi$ (Section 3.1), we parameterize the target predictor,

$$p_\tau(z' \mid \boldsymbol{u}) \;=\; \mathbb{E}_z\big[\Lambda_{\boldsymbol{u}}(z' \mid z)\big] \text{ with } z \sim D_s^\theta(z \mid \boldsymbol{u}) \;, \quad (23)$$

and a learnable transport $\Lambda_{\boldsymbol{u}}(z' \mid z)$. For convenience, we can also approximate $D_s^\theta(z \mid \boldsymbol{u})$ with $p_s(z \mid \boldsymbol{u})$ since the predictive map $p_s$ of a pre-trained foundation model often capture well the source's feature-label conditional. Consequently, the learning focuses on $\Lambda_{\boldsymbol{u}}(z' \mid z)$. We can now parameterize $\Lambda_{\boldsymbol{u}}(z' \mid z) = \Lambda_{\boldsymbol{u}}(z' \mid z; \varphi)$ and optimize its parameterization $\varphi$ via:

$$\varphi \;=\; \text{argmin}_\varphi - \mathbb{E}_{(\boldsymbol{x}', z')}\left[\log p_\tau(z' \mid \phi(\boldsymbol{x}'))\right] \;, \quad (24)$$
$$\;=\; \text{argmin}_\varphi - \mathbb{E}_{(\boldsymbol{x}', z')}\Big[\log \mathbb{E}_z\Big[\Lambda_{\phi(\boldsymbol{x}')}(z' \mid z; \varphi)\Big]\Big] \quad (25)$$

where $(\boldsymbol{x}', z') \sim (\boldsymbol{X}^\tau, \boldsymbol{z}^\tau)$. This will make the target predictor $p_\tau(z' \mid \boldsymbol{u}) = \mathbb{E}_z[\Lambda_{\boldsymbol{u}}(z' \mid z; \varphi)]$ approach the target's feature-label conditional $D_\tau^\phi(z' \mid \boldsymbol{u})$.

Since $D_\tau^\phi(z' \mid \boldsymbol{u}) = \mathbb{E}_z[\Lambda_{\boldsymbol{u}}^*(z' \mid z)]$ (see Definition 5), aligning $p_\tau(z' \mid \boldsymbol{u})$ with $D_\tau^\phi(z' \mid \boldsymbol{u})$ will decrease the gap between $\Lambda_{\boldsymbol{u}}(z' \mid z; \varphi)$ and $\Lambda_{\boldsymbol{u}}^*(z' \mid z)$. This will in turn reduce the target fitting term (**TF**) in the target generalization bound in Eq. (10), Theorem 7. See Appendix B for more details.

## 4 EMPIRICAL ANALYSIS

This section presents a detailed empirical evaluation of our proposed method RECRAFT on two popular benchmarks for cross-modal fine-tuning. These include NASBench-360 (Tu et al., 2022) and PDEBench (Takamoto et al., 2022). NAS-Bench-360 is an extensive benchmark comprising a variety of tasks across 10 distinct data modalities. PDEBench features simulated data from a variety of partial differential equations (PDEs). See Appendix J for additional details.

### 4.1 Implementation Details

We follow the experiment protocol of ORCA (Shen et al., 2023), using RoBERTa (Liu et al., 2019) for 1D tasks and Swin Transformers (Liu et al., 2021a) for 2D tasks, with CoNLL-2003 and CIFAR-10 as proxy datasets, respectively. Other settings such as learning rates, epochs, and optimizers are taken from ORCA. Full details are provided in Appendix H.

### 4.2 Results on NAS-Bench-360

The NAS-Bench-360 benchmark (Tu et al., 2022) comprises a diverse set of 10 tasks featuring specialized modalities such as protein sequences, PDE solver, audio, and genetic data, among others. In this set of experiments, we compare 4 types of baselines: (1) hand-designed solution models (Tu et al., 2022); (2) general-purpose models that accepted arbitrary inputs converted to byte arrays (without fine-tuning) such as Perceiver IO (Jaegle et al., 2021); (3) neural architecture search (NAS) methods (no knowledge transfer of existing pre-trained FMs) such as DASH (Shen et al., 2022); and (4) fine-tuning approaches including naive fine-tuning (NFT) and cross-modal fine-tuning methods such as ORCA (Shen et al., 2023), PARE (Cai et al., 2024), MoNA (Ma et al., 2024), and our proposed method RECRAFT.

Table 2 reports the prediction errors achieved by the above baselines across 10 diverse tasks on the NAS-Bench-360 benchmark. It can be observed that RECRAFT achieves the lowest prediction error rates on 8 out of 10 tasks, and second lowest error rate on 1 task. RECRAFT also achieves the best average rank among all baselines. This demonstrates RECRAFT's robustness in bridging the semantic gap between source and target tasks. We also provide the ablations across several tasks isolating contributions: NFT vs. FA-only vs. RECRAFT, as shown in Table 7 (Appendix E), RECRAFT achieves the best performance across all tasks. Overall, the result underscores the importance of accounting for feature-label distortion during representation alignment (Definition 5) which was overlooked in previous work.

### 4.3 Results on PDEBench

PDEBench comprises multiple scientific datasets with simulated data from a wide variety of partial differential equations (PDEs) in physics. We compare the fine-tuning performance of RECRAFT with those of naive fine-tuning (NFT) and prior work on cross-modal fine-tuning methods such as ORCA (Shen et al., 2023), PARE (Cai et al., 2024), and MoNA (Ma et al., 2024). Table 1 reports the prediction error achieved by

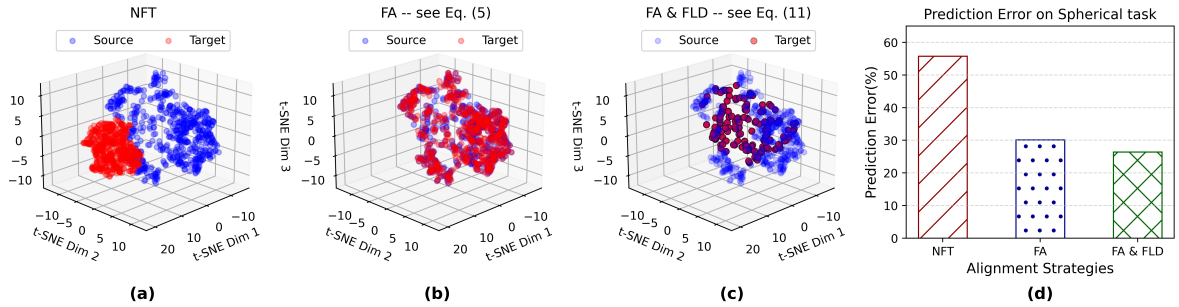


Figure 2: Visualizations of representation alignment (via tSNE) under 3 settings: (a) naive fine-tuning (NFT), which ignores alignment; (b) minimization of feature alignment (**FA**) via Eq. (5); and (c) minimizing a sum of **FA** and feature-label distortion (**FLD**) via Eq. (11). The corresponding predictive errors are shown in (d). NFT exhibits no alignment, while minimizing **FA** leads to exhaustive alignment. Both results in suboptimal performance. In contrast, minimizing **FA** + **FLD** enables selective alignment and achieves the best performance.

Table 1: Prediction errors (↓) incurred by the tested baselines across tasks in PDEBench (Takamoto et al., 2022). The results of MoNA (Ma et al., 2024) are quoted from the corresponding paper due to unavailable code. RECRAFT achieves best performance on 7 out of 8 tasks and overall average ranking of 1.25. The columns **#1** and **#2** report the number of times a method achieve best and second best performance, respectively. See Appendix E for additional details on the error bars.

| **Model** | **Darcy** (2D) nRMSE | **Advection** (1D) nRMSE | **Burgers** (1D) nRMSE | **Diffusion-Sorption** (1D) nRMSE | **Shallow Water** (2D) nRMSE | **Diffusion-Reaction** (2D) nRMSE | **Diffusion-Reaction** (1D) nRMSE | **Navier-Stokes** (1D) nRMSE | **Avg. Rank** | **#1** | **#2** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NFT | 0.085 | 0.0140 | 0.0130 | 3.1E-3 | 6.1E-3 | 0.830 | 9.2E-3 | 0.863 | 5.000 | 0 | 0 |
| ORCA | 0.081 | 0.0098 | 0.0120 | 1.8E-3 | 6.0E-3 | 0.820 | 3.2E-3 | 0.066 | 3.250 | 0 | 1 |
| PARE | 0.081 | **0.0032** | 0.0114 | 1.9E-3 | 5.9E-3 | 0.820 | 2.9E-3 | 0.068 | 3.000 | 1 | 5 |
| MoNA | **0.079** | 0.0088 | 0.0114 | **1.6E-3** | 5.7E-3 | 0.818 | **2.8E-3** | 0.054 | 1.875 | 3 | 4 |
| RECRAFT | **0.079** | 0.0078 | **0.0108** | **1.6E-3** | **5.4E-3** | **0.817** | **2.8E-3** | **0.050** | **1.250** | **7** | 1 |

the above baselines across all tasks. It is observed that RECRAFT performs best in 7/8 tasks and second best in the remaining task. It thus achieves the best overall average rank of 1.25. This is consistent with our earlier observation on the NAS-Bench-360 benchmark (Tu et al., 2022), which interestingly demonstrates the effective physic-tuning capability of RECRAFT. It also outperforms existing specialized physic-informed methods such as Fourier neural operators (Li et al., 2021b) in 4/8 tasks as shown in Table 3 in Appendix E.

### 4.4 Impact of Minimizing Semantic Gap

This section presents additional empirical evidence to support the insight in Theorem 7 that minimizing the **semantic gap** in Eq. (11) (Section 3.1) tightens the upper bound on the generalized target error, thereby improving cross-modal fine-tuning performance. To validate this connection, we track the prediction error and the corresponding semantic gap across optimization iterations used to learn the feature map in stage 1 (see Section 3.1) on three representative tasks (ECG, NinaPro, and DeepSEA) from NAS-Bench-360. As shown in Fig. 3, we observe a strong positive correlation between semantic gap and prediction error, with Pearson correlation coefficients of 0.996 (ECG), 0.965 (NinaPro), and 0.989 (DeepSEA). These results highlight the practical relevance of the theoretical bound in Theorem 7 and affirm the value of semantic gap minimization as an effective principle for representation learning in cross-modal fine-tuning.

We also study the effect of incorporating feature-label distortion (see Definition 5) in the semantic gap formulation of Eq. (11). This is illustrated in Fig. 2, which compares source-target representation alignment under different alignment strategies. Fig. 2a and Fig. 2b show tSNE feature embeddings under exclusive feature alignment (see Definition 4) and naive fine-tuning (NFT). NFT produces no alignment with the source's representation space, while exclusive feature alignment leads to an exhaustive alignment. In contrast, Fig. 2c shows that minimizing a combination of feature alignment and feature-label distortion leads to a more selective and effective alignment: target features align only with relevant regions of the source's space. This leads to substantial performance gains over both NFT and exclusive feature alignment (Table 7, Appendix E).

### 4.5 Systematic Analysis of Existing Methods

The measurable terms **FA** and **FLD** in the theoretical bound on the target generalization error in Theorem 7 provide a principled diagnostic tool for understanding the strengths and failure modes of prior cross-modal

Table 2: Prediction errors (↓) incurred by the tested baselines across 10 diverse tasks in NAS-Bench-360 (Tu et al., 2022). The results of MoNA (Ma et al., 2024) are quoted from the corresponding paper due to unavailable code. RECRAFT achieves best performance on 8 out of 10 tasks which results in the best overall average rank of 1.3 across all tasks. The columns **#1** and **#2** report the number of times a method achieve best and second best performance, respectively. See Appendix E for additional details on the error bars.

| **Model** | **Darcy** Relative $\ell_2$ | **DeepSEA** 1- AUROC | **ECG** 1-$F_1$ score | **CIFAR100** 0-1 error (%) | **Satellite** 0-1 error (%) | **Spherical** 0-1 error(%) | **Ninapro** 0-1 error (%) | **Cosmic** 1- AUROC | **Psicov** $MAE_8$ | **FSD50K** 1-mAP | **Avg. Rank** | **#1** | **#2** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hand-designed | 8.0E-3 | 0.30 | 0.28 | 19.39 | 19.80 | 67.41 | 8.74 | 0.13 | 3.37 | 0.62 | 5.6 | 0 | 1 |
| NAS-Bench-360 | 2.6E-2 | 0.32 | 0.34 | 23.39 | 12.51 | 48.23 | 7.35 | 0.23 | 2.95 | 0.63 | 5.8 | 0 | 0 |
| DASH | 8.0E-3 | 0.28 | 0.32 | 24.37 | 12.28 | 71.38 | 6.63 | 0.20 | 3.30 | 0.60 | 4.9 | 0 | 2 |
| Perceiver IO | 2.4E-2 | 0.38 | 0.66 | 70.04 | 15.96 | 82.57 | 22.4 | 0.49 | 8.10 | 0.73 | 7.7 | 0 | 0 |
| NFT | 7.4E-3 | 0.490 | 0.44 | 9.74 | 13.82 | 55.76 | 8.35 | 0.17 | 1.92 | 0.63 | 5.6 | 0 | 0 |
| ORCA | 7.5E-3 | 0.291 | 0.30 | 7.80 | 11.63 | 29.87 | 7.74 | 0.15 | 1.91 | 0.56 | 3.8 | 0 | 2 |
| PaRE | 7.4E-3 | 0.286 | 0.28 | 6.70 | 11.21 | 27.04 | 7.12 | 0.12 | **0.99** | **0.55** | 2.2 | 2 | 4 |
| MoNA | **6.8E-3** | 0.280 | **0.27** | **6.48** | 11.13 | 27.13 | 7.28 | 0.121 | **0.99** | **0.55** | 1.9 | 5 | 2 |
| RECRAFT | 7.2E-3 | **0.278** | **0.27** | 7.30 | **11.11** | **26.41** | **6.60** | **0.11** | **0.99** | **0.55** | **1.3** | **8** | 1 |

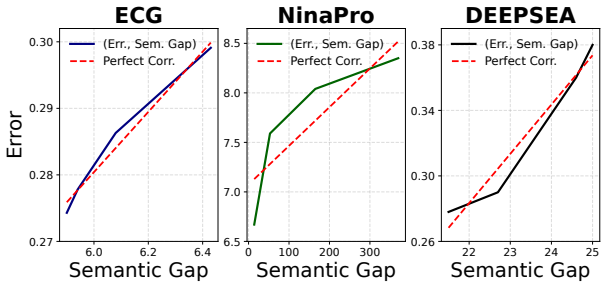


Figure 3: Target error versus semantic gap (Eq. (11)) for RECRAFT on ECG, NinaPro, and DeepSEA. All plots show a strong, consistent correlation across tasks.

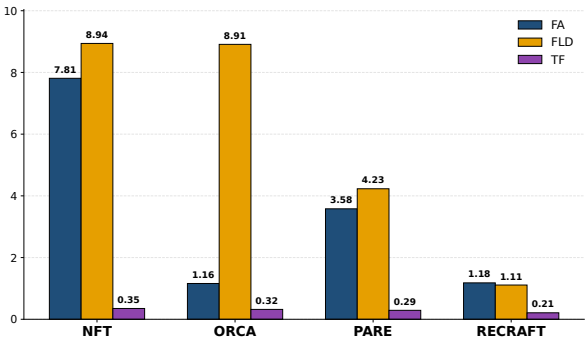


Figure 4: Detailed breakdown of the performance bound in Theorem 7 into **FA**, **FLD**, and **TF** achieved with different cross-modal fine-tuning methods on the Cosmic dataset. The results show that RECRAFT incurs much lower **FLD** than other baselines.

adaptation techniques, which remain largely empirical. This section provides additional experiments to measure the **FA** (Eq. 14), **FLD** (Eq. 7), and **TF** (Eq. 9) terms incurred by previous methods (Fig. 4).

Based on the results illustrated in Fig. 4, we can draw the following conclusions. First, NFT (Naive Fine-Tuning) incurs much larger **FA** and **FLD** losses than the other methods and consequently achieves the worst performance (see Tab. 2 and Tab. 1). Second, ORCA effectively reduces **FA** but barely reduces **FLD** by an insignificant amount. This is not surprising given that ORCA performs **FA** and **TF** in separate stages without accounting for their incurred **FLD**. This can make **TF** overcompensate for a suboptimal **FA**, which increases the risk of overfitting. Consequently, its effectiveness remains limited compared to our approach.

Third, PARE has a better balance between **FA** and **FLD** compared to ORCA. However, its significant reduction on **FLD** appears to come at a cost of increasing **FA** (compared to ORCA), which leads to an overall marginal reduction in **FA** + **FLD**. The significant reduction on **FLD** can be attributed to PARE's design which aims to minimize a linear combination of source and target losses. It uses an intermediate distributional representation that combines the most important information from both source and target modalities. Including most important information from the source modalities in the aligned representation helps constrain the feature alignment (**FA**) within the most important regions in the source's representation landscape, which in turn limits how far the aligned features can deviate from the source's feature-label structure. This can help reduce **FLD**. Nonetheless, this scheme was not optimized to prevent negative transfer, where some source-specific important information is irrelevant to the target's task but might still be included in the intermediate representation (see Fig. 2). As such, the feature alignment must account for such extra, irrelevant information and hence, increases the **FA** loss. This explains why PARE incurs larger **FA** than ORCA despite having smaller **FLD**.

In contrast, our method RECRAFT implements a principled minimization procedure for both **FA** + **FLD** (Section 3.1) and **TF** (Section 3.2). This leads to a significant reduction in both **FA** and **FLD** while also reducing **TF**, thus consistently achieving better performance than all the above baselines (Section 4). Moreover, our theoretical decomposition provides a new analytical lens that will inspire important

and synergistic research directions (Appendix K).

# 5 RELATED WORKS

Due to limited space, we provide a short review of the most relevant work on in-modal and cross-modal fine-tuning. A broader review is deferred to Appendix C.

## 5.1 In-Modal Fine-Tuning of FMs

The recent advent of large pre-trained or foundation models (FMs) has enabled flexible knowledge transfer to a wide range of downstream tasks under a unified, task-agnostic paradigm. This is commonly known as fine-tuning, which is model-agnostic and does not require shared input or output spaces. Such transfer is feasible because these FMs are trained on broad, diverse corpora (e.g., text, image, or audio) that often overlap semantically or structurally with the content of downstream datasets for in-modal tasks. As a result, fine-tuning has been successfully applied across domains such as vision (Kirillov et al., 2023; Liu et al., 2021b; Bui et al., 2024; Weng et al., 2024; Phung et al., 2026), video understanding (Bertasius et al., 2021), language (Zheng et al., 2021; Yang et al., 2022; Ma et al., 2023), and speech (Radford et al., 2022; Li et al., 2021a). There are also multi-modal FMs (Radford et al., 2021; Alayrac et al., 2022; Kim et al., 2021) which were pre-trained to learn the embeddings of multiple modalities together but existing approaches to facilitate knowledge transfer from these models are still restricted to within the pre-trained modalities.

## 5.2 Cross-Modal Fine-Tuning of FMs

Early attempts in cross-modal fine-tuning have focused on transferring language models to other modalities such as vision (Kiela et al., 2019; Tan and Bansal, 2019; Gu et al., 2022), DNA/protein sequences (Nguyen et al., 2023; Lin et al., 2023; Jumper et al., 2021). These provide initial evidence of cross-modal transferability but their designs are hand-tailored to specific target tasks and modalities rather than for general purpose (Shen et al., 2023). Recently, a few general-purpose cross-modal fine-tuning framework have emerged with remarkable successes in finetuning existing vision (Liu et al., 2021b) or language (Liu et al., 2019) FMs to solve a broad, diverse set of unseen tasks and data modalities (Shen et al., 2023; Cai et al., 2024; Ma et al., 2024).

In particular, ORCA (Shen et al., 2023) aligns source and target representation distributions using optimal transport before fine-tuning the entire network. PARE (Cai et al., 2024) alternatively introduces a gating mechanism to integrate source and target features during fine-tuning. MoNA (Ma et al., 2024) addresses modality representation alignment via a bi-level optimization framework: the inner loop learns the optimal target predictor for a given embedder, while the outer loop updates the embedder to align the source feature-label semantics with the combined representation and prediction under this predictor. This approach heuristically reduces the misalignment between source's and target's feature-label structures while fitting to the target data. However, as noted in Section 1, these approaches largely adopt heuristic methods combining distributional feature alignment and target fitting to facilitate cross-modal knowledge transfer. Overall, these approaches lack a principled means to assess the impact of their heuristic strategies on the generalized target performance.

# 6 LIMITATIONS AND FUTURE WORKS

In future work, our proposed method will be further generalized to address two existing limitations. First, while the use of Euclidean cost metric to characterize the geometric structure under which the Wasserstein distance is computed does not invalidate the bound in Theorem 7, it is unclear whether a Wasserstein distance parameterized with Euclidean cost metric would optimally tighten the upper-bound. We will investigate this aspect via developing a metric learning component which parameterizes and learns this cost metric as additional parameters of the upper-bound. Second, while upper-bounding **FLD** with the conditional entropic terms in Eq. 18 and approximating them with a well-established pseudo-label approach (Nguyen et al., 2020) results in a valid empirical upper-bound, it is unclear whether the bound gap can be made vanishingly small as we increase the no. of sampled pseudo labels (see Eq. 19). This remains an open question to be addressed in the future generalization of our work.

# 7 CONCLUSION

This paper presents a new theoretical perspective for cross-modal fine-tuning which highlights and addresses a limitation that hinders generalized knowledge transfer in existing work. This leads to a novel development of a theoretical framework that provably establishes an upper-bound on the generalized target performance. The bound reveals a precise computational representation for the interaction between feature alignment and target fitting. This inspires a practical algorithm that optimize this interaction in a principled manner, paving the way for more effective algorithm designs. The developed algorithm performs significantly better than the recent SOTA methods on two extensive cross-modal fine-tuning benchmark, thus conclusively validating our theoretical insight.

## Acknowledgement

This work utilized GPU compute resource at SDSC and ACES through allocation CIS230391 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services and Support (ACCESS) program Boerner et al. (2023), which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296.

### References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning. *ArXiv*, abs/2204.14198, 2022. URL `https://api.semanticscholar.org/CorpusID:248476411`.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding?, 2021.

Timothy J. Boerner, Stephen Deems, Thomas R. Furlani, Shelley L. Knuth, and John Towns. Access: Advancing innovation: Nsf's advanced cyberinfrastructure coordination ecosystem: Services & support. In *Practice and Experience in Advanced Research Computing 2023: Computing for the Common Good*, PEARC '23, page 173–176, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450399852. doi: 10.1145/3569951.3597559. URL `https://doi.org/10.1145/3569951.3597559`.

Long Minh Bui, Tho Tran Huu, Duy Dinh, Tan Minh Nguyen, and Trong Nghia Hoang. Revisiting kernel attention with correlated gaussian process representation. In *The 40th Conference on Uncertainty in Artificial Intelligence*, 2024. URL `https://openreview.net/forum?id=xlIK0vu3MW`.

Lincan Cai, Shuang Li, Wenxuan Ma, Jingxuan Kang, Binhui Xie, Zixun Sun, and Chengwei Zhu. Enhancing cross-modal fine-tuning with gradually intermediate modality generation, 2024. URL `https://arxiv.org/abs/2406.09003`.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, July 2006. Theorem 2.6.5 states "Conditioning reduces entropy (H(X—Y) <= H(X)), sometimes summarized as 'information never hurts.'".

Junnan Gu, Xi Chen, Yang Liu, and Ming Li. Vision-language pre-training with triple contrastive learning, 2022.

Minh Hoang and Trong Nghia Hoang. Few-shot learning via repurposing ensemble of black-box models. In *Proc. AAAI*, 2024.

Trong Nghia Hoang, Chi Thanh Lam, Kian Hsiang Low, and Patrick Jaillet. Learning Task-Agnostic Embedding of Multiple Black-box Experts for Multi-Task Model Fusion. In *Proc. ICML*, 2020.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795*, 2021.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A A Kohl, Andrew J Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold, 2021.

Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, and Davide Testuggine. Supervised multimodal bitransformers for classifying images and text. *ArXiv*, abs/1909.02950, 2019. URL `https://api.semanticscholar.org/CorpusID:202539204`.

Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, 2021. URL `https://api.semanticscholar.org/CorpusID:231839613`.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL `https://arxiv.org/abs/1412.6980`.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

Chi Thanh Lam, Trong Nghia Hoang, Kian Hsiang Low, and Patrick Jaillet. Model Fusion for Personalized Learning. In *Proc. ICML*, 2021.

Shuang Li, Binhui Xie, Jiashu Wu, Ying Zhao, Chi Harold Liu, and Zhengming Ding. Simul-

taneous semantic alignment network for heterogeneous domain adaptation. *Proceedings of the 28th ACM International Conference on Multimedia*, 2020. URL https://api.semanticscholar.org/CorpusID:220961739.

Xian Li, Changhan Wang, Yun Tang, Chau Tran, Yuqing Tang, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. Multilingual speech translation from efficient finetuning of pretrained models. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 827–838, Online, August 2021a. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.68. URL https://aclanthology.org/2021.acl-long.68/.

Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=c8P9NQVtmnO.

Yuxiang Lin, Ling Luo, Ying Chen, Xushi Zhang, Zihui Wang, Wenxian Yang, Mengsha Tong, and Rongshan Yu. St-align: A multimodal foundation model for image-gene alignment in spatial transcriptomics. *ArXiv*, abs/2411.16793, 2024. URL https://api.semanticscholar.org/CorpusID:274280682.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574. URL https://www.science.org/doi/abs/10.1126/science.ade2574. Earlier versions as preprint: bioRxiv 2022.07.20.500902.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021a.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021b.

Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5):5, 2017.

Wenxuan Ma, Shuang Li, Lincan Cai, and Jingxuan Kang. Language semantic graph guided data-efficient learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=tUyW68cRqr.

Wenxuan Ma, Shuang Li, Lincan Cai, and Jingxuan Kang. Learning modality knowledge alignment for cross-modality transfer, 2024. URL https://arxiv.org/abs/2406.18864.

Cuong V. Nguyen, Tal Hassner, Matthias Seeger, and Cedric Archambeau. Leep: A new measure to evaluate transferability of learned representations. In *Proceedings of the 37th International Conference on Machine Learning*, pages 7294–7305. PMLR, 2020.

Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Callum Birch-Sykes, Michael Wornow, Aman Patel, Clayton Rabideau, Stefano Massaroli, Yoshua Bengio, Stefano Ermon, Stephen A. Baccus, and Chris Ré. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution, 2023. URL https://arxiv.org/abs/2306.15794.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010. doi: 10.1109/TKDE.2009.191.

Gabriel Peyré and Marco Cuturi. Computational optimal transport, 2020. URL https://arxiv.org/abs/1803.00567.

Thu Hang Phung, Duong M. Nguyen, Thanh Trung Huynh, Quoc Viet Hung Nguyen, Trong Nghia Hoang, and Phi Le Nguyen. Federated prompt-tuning with heterogeneous and incomplete multimodal client data. *ArXiv*, abs/2602.07081, 2026. URL https://api.semanticscholar.org/CorpusID:284585890.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. URL https://api.semanticscholar.org/CorpusID:231591445.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Ro-

bust speech recognition via large-scale weak supervision, 2022. URL https://arxiv.org/abs/2212.04356.

M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2018.10.045. URL https://www.sciencedirect.com/science/article/pii/S0021999118307125.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation, 2018. URL https://arxiv.org/abs/1707.01217.

Junhong Shen, Mikhail Khodak, and Ameet Talwalkar. Efficient architecture search for diverse tasks, 2022. URL https://arxiv.org/abs/2204.07554.

Junhong Shen, Liam Li, Lucio M. Dery, Corey Staten, Mikhail Khodak, Graham Neubig, and Ameet Talwalkar. Cross-modal fine-tuning: Align then refine. In *Proceedings of the International Conference on Machine Learning (ICML)*. ICML, 2023. URL https://arxiv.org/abs/2302.05738.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Dan MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. Pdebench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2022.

Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers, 2019.

Renbo Tu, Nicholas Roberts, Mikhail Khodak, Junhong Shen, Frederic Sala, and Ameet Talwalkar. NAS-bench-360: Benchmarking neural architecture search on diverse tasks. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://openreview.net/forum?id=xUXTbq6gWsB.

C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2008. ISBN 9783540710509. URL https://books.google.com.vn/books?id=hV8o5R7_5tkC.

Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2018.05.083. URL https://www.sciencedirect.com/science/article/pii/S0925231218306684.

Pei-Yau Weng, Minh Hoang, Lam M. Nguyen, My T. Thai, Tsui-Wei Weng, and Trong Nghia Hoang. Probabilistic federated prompt-tuning with non-IID and imbalanced data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=nw6ANsC66G.

Huiyun Yang, Huadong Chen, Hao Zhou, and Lei Li. Enhancing cross-lingual transfer by manifold mixup. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=OjPmfr9GkVv.

Yuan Yao, Yu Zhang, Xutao Li, and Yunming Ye. Heterogeneous domain adaptation via soft transfer network. *Proceedings of the 27th ACM International Conference on Multimedia*, 2019. URL https://api.semanticscholar.org/CorpusID:201651510.

Wenzhe Yin, Shujian Yu, Yicong Lin, Jie Liu, Jan-Jakob Sonke, and Efstratios Gavves. Domain adaptation with cauchy-schwarz divergence, 2024. URL https://arxiv.org/abs/2405.19978.

Bo Zheng, Li Dong, Shaohan Huang, Wenhui Wang, Zewen Chi, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. Consistency Regularization for Cross-Lingual Fine-Tuning. In *Proceedings of ACL 2021*, 2021.

## Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes, see Appendix H]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [ Yes, see Appendix H ]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
   (d) Information about consent from data providers/curators. [Not Applicable, Public datasets]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

# Supplementary Materials

## A Proof of Theorem 7

This section provides the detailed proof of our main result stated in Theorem 7, which bounds the gap between the target and source generalized errors. The bound is characterized in terms of (i) the distributional feature alignment (**FA**) between the source and target in Eq. (5), (ii) the feature-label distortion (**FLD**) under their respective feature maps in Eq. (7), and (iii) the target model's fit (**TF**) to the fine-tuning data in Eq. (9). For clarity, we restate the result below.

$$\mathrm{err}_\tau(\phi) \;\leq\; \mathrm{err}_s(\theta) \;+\; \mathbf{FA}(\phi,\theta) \;+\; \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\Big[\mathbf{FLD}(\boldsymbol{u}) \;+\; \mathbf{TF}(\boldsymbol{u})\Big]\,, \tag{26}$$

Here, $D_\tau^\phi(\boldsymbol{u})$ denote the target's marginal feature distribution under (target) feature map $\phi$ and $D_s^\theta(z \mid \boldsymbol{u})$ denote the feature-label conditional under (source) feature map $\theta$. Our proof goes below.

First, following Definition 1, the generalized target error is

$$\mathrm{err}_\tau(\phi) \;\triangleq\; -\mathbb{E}_{(\boldsymbol{x}',z')\sim D_\tau}\Big[\log p_\tau\big(z' \mid \phi(\boldsymbol{x}')\big)\Big] \tag{27}$$

$$=\; -\mathbb{E}_{(\boldsymbol{u},z')\sim D_\tau^\phi}\Big[\log p_\tau\big(z' \mid \boldsymbol{u}\big)\Big] \;=\; -\mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_\tau^\phi(z'|\boldsymbol{u})}\Big[\log p_\tau\big(z' \mid \boldsymbol{u}\big)\Big]\,. \tag{28}$$

Likewise, the generalized source error is

$$\mathrm{err}_s(\theta) \;=\; -\mathbb{E}_{D_s^\theta(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\log p_s\big(z \mid \boldsymbol{u}\big)\Big]\,. \tag{29}$$

Combining Eqs. (28) and (29), we can rewrite

$$\mathrm{err}_\tau(\phi) - \mathrm{err}_s(\theta) \;=\; \mathbf{A} + \mathbf{B} \quad \text{where} \tag{30}$$

$$\mathbf{A} \;\triangleq\; \mathrm{err}_\tau(\phi) + \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\log D_s^\theta(z \mid \boldsymbol{u})\,, \tag{31}$$

$$\mathbf{B} \;\triangleq\; -\mathrm{err}_s(\theta) - \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\log D_s^\theta(z \mid \boldsymbol{u})\,. \tag{32}$$

We will bound **A** and **B** next.

**1. Bounding A.** Plugging Eq. (28) into Eq. (31), we have

$$\mathbf{A} \;=\; \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\log D_s^\theta(z \mid \boldsymbol{u})\Big] \;-\; \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_\tau^\phi(z'|\boldsymbol{u})}\Big[\log p_\tau(z' \mid \boldsymbol{u})\Big]\,. \tag{33}$$

Following Definition 5, let $\Lambda_{\boldsymbol{u}}^*(z' \mid z) \in C_{\boldsymbol{u}}^*$ denote a valid transport map from $D_s^\theta(z \mid \boldsymbol{u})$ to $D_\tau^\phi(z' \mid \boldsymbol{u})$. That is,

$$D_\tau^\phi(z' \mid \boldsymbol{u}) \;=\; \mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\Lambda_{\boldsymbol{u}}^*(z' \mid z)\Big]\,. \tag{34}$$

Plugging Eq. (34) into Eq. (33), we can rewrite

$$\mathbf{A} \;=\; \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\log D_s^\theta(z \mid \boldsymbol{u})\Big] \;-\; \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\left[\sum_{z'}\Lambda_{\boldsymbol{u}}^*(z' \mid z)\log p_\tau(z' \mid \boldsymbol{u})\right]\,. \tag{35}$$

Furthermore, we also have

$$\begin{aligned} -\log p_\tau(z' \mid u) \;&=\; -\log\left(\sum_a \Lambda_{\boldsymbol{u}}(z' \mid a) D_s^\theta(a \mid \boldsymbol{u})\right) \;\leq\; -\log\Big(\Lambda_{\boldsymbol{u}}(z' \mid z) D_s^\theta(z \mid \boldsymbol{u})\Big) \\ &=\; -\log\Lambda_{\boldsymbol{u}}(z' \mid z) - \log D_s^\theta(z \mid \boldsymbol{u})\,. \end{aligned} \tag{36}$$

for any $z$ and valid transport map $\Lambda_{\boldsymbol{u}}(z' \mid .) \in C_{\boldsymbol{u}}$ from $D_s^\theta(z \mid \boldsymbol{u})$ to $p_\tau(z' \mid \boldsymbol{u})$ as defined in Definition 6. Plugging Eq. (36) into Eq. (35), we obtain the following upper-bound,

$$\begin{aligned}
\mathbf{A} &\leq \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\log D_s^\theta(z \mid \boldsymbol{u})\Big] \\
&- \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\left[\sum_{z'} \Lambda_{\boldsymbol{u}}^*(z' \mid z) \log \Lambda_{\boldsymbol{u}}(z' \mid z)\right] - \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\left[\sum_{z'} \Lambda_{\boldsymbol{u}}^*(z' \mid z) \log D_s^\theta(z \mid \boldsymbol{u})\right] \quad (37) \\
&= \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\log D_s^\theta(z \mid \boldsymbol{u})\Big] \\
&- \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\left[\sum_{z'} \Lambda_{\boldsymbol{u}}^*(z' \mid z) \log \Lambda_{\boldsymbol{u}}(z' \mid z)\right] - \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\log D_s^\theta(z \mid \boldsymbol{u})\Big] \\
&= \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\left[-\sum_{z'} \Lambda_{\boldsymbol{u}}^*(z' \mid z) \log \Lambda_{\boldsymbol{u}}(z' \mid z)\right] = \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\ \mathbb{H}\Big[\Lambda_{\boldsymbol{u}}^*(. \mid z), \Lambda_{\boldsymbol{u}}(. \mid z)\Big] \\
&= \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\ \mathbb{H}\Big[\Lambda_{\boldsymbol{u}}^*(. \mid z)\Big] + \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\ \mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\ \mathbb{KL}\Big[\Lambda_{\boldsymbol{u}}^*(. \mid z) \parallel \Lambda_{\boldsymbol{u}}(. \mid z)\Big] . \quad (38)
\end{aligned}$$

As Eq. (38) holds for all choices of $\Lambda_{\boldsymbol{u}}^*(z' \mid z) \in C_{\boldsymbol{u}}^*$ and $\Lambda_{\boldsymbol{u}}(z' \mid z) \in C_{\boldsymbol{u}}$, we can tighten its right-hand side (RHS) by choosing for each $(\boldsymbol{u})$, $\Lambda_{\boldsymbol{u}}^*(\cdot \mid \cdot) \in C_{\boldsymbol{u}}^*$ that minimizes $\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\mathbb{H}[\Lambda_{\boldsymbol{u}}^*(. \mid z)]\Big]$ and taking minimum over the remaining choice of $\Lambda_{\boldsymbol{u}}(\cdot \mid \cdot) \in C_{\boldsymbol{u}}$. This results in a tighten bound below:

$$\mathbf{A} \leq \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\left[\min_{\Lambda_{\boldsymbol{u}}^* \in C_{\boldsymbol{u}}^*} \mathbb{E}_{D_s^\phi(z|\boldsymbol{u})}\Big[\mathbb{H}\Big[\Lambda_{\boldsymbol{u}}^*(. \mid z)\Big]\Big]\right] + \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\left[\min_{\Lambda_{\boldsymbol{u}} \in C_{\boldsymbol{u}}} \mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\mathbb{KL}\Big(\Lambda_{\boldsymbol{u}}^+(. \mid z) \| \Lambda_{\boldsymbol{u}}(. \mid z)\Big)\Big]\right] . \quad (39)$$

where $\Lambda_{\boldsymbol{u}}^+(. \mid .) = \text{argmin}_{\Lambda_{\boldsymbol{u}}^* \in C_{\boldsymbol{u}}^*}\ \mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\Big[\mathbb{H}[\Lambda_{\boldsymbol{u}}^*(. \mid z)]\Big]$. Following the definition of **FLD** and **TF** in Eqs. (7)-(9),

$$\mathbf{A} \leq \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\Big[\mathbf{FLD}(\boldsymbol{u}) + \mathbf{TF}(\boldsymbol{u})\Big] , \quad (40)$$

**2. Bounding B.** We will now show that **B** in Eq. (32) is upper-bounded by $\mathbf{FA}(\phi, \theta)$ in Eq. (5) to complete the proof. To see this, note that in practice, $p_s(z \mid \boldsymbol{u})$ often approximates $D_s^\theta(z \mid \boldsymbol{u})$ faithfully. Exploiting this practical property, we can rewrite **B** in Eq. (32) as

$$\begin{aligned}
\mathbf{B} &= \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\Big[-\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\big[\log p_s(z \mid \boldsymbol{u})\big]\Big] - \mathbb{E}_{D_s^\theta(\boldsymbol{u})}\Big[-\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}\big[\log p_s(z \mid \boldsymbol{u})\big]\Big] \quad (41) \\
&= \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\big[\ell_s(\boldsymbol{u})\big] - \mathbb{E}_{D_s^\theta(\boldsymbol{u})}\big[\ell_s(\boldsymbol{u})\big] . \quad (42)
\end{aligned}$$

where $\ell_s(\boldsymbol{u}) \triangleq \mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}[-\log p_s(z \mid \boldsymbol{u})]$ as previously defined in Definition 4. For any cost metric $\delta \in \Delta$ such that $|\ell_s(\boldsymbol{u}_1) - \ell_s(\boldsymbol{u}_2)| \leq \tau_\delta \cdot \delta(\boldsymbol{u}_1, \boldsymbol{u}_2)$, the Kantorovich-Rubinstein duality ascertains that

$$\mathbf{B} \leq \tau_\delta \cdot W_\delta\Big(D_\tau^\phi(\boldsymbol{u}), D_s^\theta(\boldsymbol{u})\Big) . \quad (43)$$

As this is true for any $\delta \in \Delta$, we can again tighten the right-hand side (RHS) of Eq. (43) via taking minimum over the choice of $\delta$. That is,

$$\mathbf{B} \leq \min_{\delta \in \Delta}\Big\{\tau_\delta \cdot W_\delta\Big(D_\tau^\phi(\boldsymbol{u}), D_s^\theta(\boldsymbol{u})\Big)\Big\} = \mathbf{FA}(\phi, \theta) . \quad (44)$$

Plugging Eqs. (40) and (44) in Eq. (30) completes our proof.

## B Tightness of the Generalization Bound in Theorem 7

This section evaluates the empirical tightness of the bound in Theorem 7 which is quoted below for convenience:

$$\text{err}_\tau(\phi) \leq \text{err}_s(\theta) + \mathbf{FA}(\phi, \theta) + \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}\Big[\mathbf{FLD}(\boldsymbol{u}) + \mathbf{TF}(\boldsymbol{u})\Big] . \quad (45)$$

Table 3: Comparison of RECRAFT and Baselines on Multiple PDE Tasks (↓ indicates lower is better). U-Net results for Navier-Stokes and Darcy-Flow are unavailable (due to memory constraints) in the benchmark paper. RECRAFT achieves an average rank of **1.5** and attains the best performance on 4 out of 8 tasks. Columns **#1** and **#2** indicate the number of times each method achieves the best and second-best performance, respectively.

| Model | Darcy (2D) nRMSE | Advection (1D) nRMSE | Burgers (1D) nRMSE | Diffusion-Sorption (1D) nRMSE | Shallow Water (2D) nRMSE | Diffusion-Reaction (2D) nRMSE | Diffusion-Reaction (1D) nRMSE | Navier-Stokes (1D) nRMSE | Avr. Rank | #1 | #2 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PINN | 0.18 | 0.67 | 0.36 | 0.15 | 0.085 | 0.84 | 0.84 | 0.720 | 3.125 | 0 | 1 |
| FNO | 0.22 | 0.011 | **0.0031** | 1.8E-3 | **4.4E-3** | **0.12** | **1.4E-3** | 0.068 | 1.625 | 4 | 3 |
| U-Net | - | 1.1 | 0.99 | 0.22 | 0.017 | 1.6 | 0.08 | - | - | 0 | 0 |
| RECRAFT | **0.079** | **0.0078** | 0.0108 | **1.6E-3** | 5.4E-3 | 0.817 | 2.8E-3 | **0.050** | **1.500** | 4 | 4 |

Table 4: Prediction errors (↓) incurred by the tested methods (with standard deviation) across 10 diverse tasks on NAS-Bench-360. The reported results of MoNA (Ma et al., 2024) are quoted from the corresponding paper since its source code is not released. Our method RECRAFT achieves best performance in 8 out 10 tasks.

| Model | Darcy Relative $\ell_2$ | DeepSEA 1- AUROC | ECG 1-$F_1$ score | CIFAR100 0-1 error (%) | Satellite 0-1 error (%) | Spherical 0-1 error(%) | Ninapro 0-1 error (%) | Cosmic 1- AUROC | Psicov $MAE_8$ | FSD50K 1-mAP |
|---|---|---|---|---|---|---|---|---|---|---|
| Hand-designed | 8.0E-3 ± 1E-3 | 0.300 ± 2E-2 | 0.28 ± 0.01 | 19.39 ± 0.2 | 19.80 ± 0.01 | 67.41 ± 0.8 | 8.74 ± 0.9 | 0.13 ± 1E-2 | 3.37 ± 0.15 | 0.62 ± 4E-3 |
| NAS-Bench-360 | 2.6E-2 ± 1E-3 | 0.320 ± 1E-2 | 0.34 ± 0.01 | 23.39 ± 0.03 | 12.51 ± 0.25 | 48.23 ± 2.5 | 7.35 ± 0.8 | 0.23 ± 4E-3 | 2.95 ± 0.14 | 0.60 ± 0.03 |
| DASH | 8.0E-3 ± 2E-3 | 0.280 ± 1E-2 | 0.32 ± 6E-3 | 24.37 ± 0.83 | 12.28 ± 0.50 | 71.38 ± 0.7 | 6.63 ± 0.4 | 0.20 ± 5E-3 | 3.30 ± 0.17 | 0.60 ± 0.02 |
| Perceiver IO | 2.4E-2 ± 1E-2 | 0.380 ± 4E-3 | 0.66 ± 0.01 | 70.04 ± 0.4 | 15.96 ± 0.01 | 82.57 ± 0.2 | 22.4 ± 1.5 | 0.49 ± 1E-2 | 8.10 ± 0.05 | 0.73 ± 0.02 |
| NFT | 7.4E-3 ± 1E-4 | 0.490 ± 2E-2 | 0.44 ± 0.03 | 9.74 ± 1.21 | 13.82 ± 0.24 | 55.76 ± 2.3 | 8.35 ± 0.4 | 0.17 ± 2E-2 | 1.92 ± 0.06 | 0.63 ± 0.01 |
| ORCA | 7.5E-3 ± 8E-5 | 0.291 ± 2E-3 | 0.30 ± 6E-3 | 7.80 ± 0.41 | 11.63 ± 0.20 | 29.87 ± 0.8 | 7.74 ± 0.4 | 0.15 ± 5E-3 | 1.91 ± 0.04 | 0.56 ± 0.01 |
| PaRE | 7.4E-3 ± 1E-4 | 0.286 ± 4E-3 | 0.28 ± 7E-3 | 6.70 ± 0.3 | 11.21 ± 0.07 | 27.04 ± 0.7 | 7.12 ± 0.3 | 0.12 ± 4E-3 | **0.99 ± 0.03** | **0.55 ± 0.01** |
| MoNA | **6.8E-3** | 0.280 | **0.27** | **6.48** | 11.13 | 27.13 | 7.28 | 0.121 | **0.99** | **0.55** |
| RECRAFT | 7.2E-3 ± 7E-5 | **0.278 ± 2E-3** | **0.27 ± 6E-3** | 7.30 ± 0.4 | **11.11 ± 0.04** | **26.41 ± 0.7** | **6.60 ± 0.3** | **0.11 ± 3E-3** | **0.99 ± 0.02** | **0.55 ± 0.01** |

This evaluation is conducted with respect to the pre-trained source encoder $\theta$, the target encoder $\phi$ as well as the corresponding target prediction heads $p_\tau(z' \mid \boldsymbol{u})$ which are learned via optimizing the above bound using our proposed algorithm in the main text. We will show that the bound is sufficiently tight for the learned target encoder $\phi$ which consequently demonstrates that the bound in Theorem 7 is a sufficiently good surrogate to optimize for the (inaccessible) target generalization loss.

To achieve this, we use the source's and target's *test sets*, which were not used during the pre-training and fine-tuning of the source and target models, to compute the corresponding target loss $\text{err}_\tau(\phi)$ and source loss $\text{err}_s(\theta)$. To compute the bound on the target generalization loss, we calculate the feature alignment (**FA**) using Eq. (14) and approximate the feature-label distortion (**FLD**) using the upper bound in Eq. (21). We then compute the optimal target fitting term **TF** at the learned target feature map $\phi$ and prediction head $p_\tau(z' \mid \boldsymbol{u})$ on the target test set via (1) solving for $\Lambda^+_{\boldsymbol{u}}(. \mid .)$ that minimizes the feature-label distortion (**FLD**),

$$\Lambda^+_{\boldsymbol{u}}(\cdot \mid \cdot) \quad = \quad \text{argmin}_{\Lambda^*_{\boldsymbol{u}} \in C^*_{\boldsymbol{u}}} \mathbb{E}_{D^\theta_s(z|\boldsymbol{u})} \Big[ \mathbb{H}\big[\Lambda^*_{\boldsymbol{u}}(. \mid z)\big] \Big] \ , \tag{46}$$

with respect to the linear constraint in Eq. (6); and (2) leveraging it to equivalently rewrite the **TF** formulation in Definition 6 as the optimal solution for a convex optimization task with linear constraints:

$$\begin{aligned} \mathbf{TF}(\boldsymbol{u}) \quad &= \quad \min_{\Lambda_{\boldsymbol{u}}(.|.)} \ \mathbb{E}_{D^\theta_s(z|\boldsymbol{u})} \Big[ \mathbb{KL}\big(\Lambda^+_{\boldsymbol{u}}(\cdot|z) || \Lambda_{\boldsymbol{u}}(\cdot|z)\big) \Big] \\ \text{subject to} \ \ p_\tau(z'|\boldsymbol{u}) \quad &= \quad \mathbb{E}_{D^\theta_s(z|\boldsymbol{u})} \Big[ \Lambda_{\boldsymbol{u}}(z'|z) \Big] \quad \text{for each} \quad z' \in \mathcal{Z}' \ . \end{aligned} \tag{47}$$

The above is a direct consequence of the formulation in Definition 6 when we fix a particular choice of the target's feature map $\phi$ and its corresponding prediction head $p_\tau(z' \mid \boldsymbol{u})$.

Importantly, we note that such formulation assumes knowledge of $\phi$ and $p_\tau(z' \mid \boldsymbol{u})$ as well as the test set and therefore cannot be used as an alternative to our proposed algorithm for learning $\phi$ and $p_\tau(z' \mid \boldsymbol{u})$. Instead, this formulation is an effective probing tool for post-training inspection/evaluation of the tightness of the theoretical bound in Theorem 7 as it admits a closed-form solution for $\Lambda_{\boldsymbol{u}}$ in terms of the (learned) target feature map $\phi$, prediction head $p_\tau(z' \mid \boldsymbol{u})$, and the corresponding target test data's induced conditional distribution $D^\phi_\tau(z' \mid \boldsymbol{u})$:

$$\Lambda_{\boldsymbol{u}}(z'|z) \quad = \quad \Lambda^+_{\boldsymbol{u}}(z'|z) \cdot p_\tau(z'|\boldsymbol{u}) \ / \ D^\phi_\tau(z'|\boldsymbol{u}) \quad \text{and} \quad \mathbf{TF}(\boldsymbol{u}) \quad = \quad \mathbb{KL}\Big(D^\phi_\tau(z'|\boldsymbol{u}) \parallel p_\tau(z'|\boldsymbol{u})\Big) \ . \tag{48}$$

**Remark.** Eq. (48) also provides direct support for the claim in Section 3.2 that minimizing the target loss also minimizes the target fitting error (**TF**). To elaborate, given a feature map $\phi$ and a specified target task, the oracle conditional distribution $D_\tau^\phi(z' \mid \boldsymbol{u})$ is fixed. During training phase in Section 3.2, the target predictor $p_\tau(z' \mid \boldsymbol{u})$ is trained to minimize the target loss, which, in turn, reduces the Kullback-Leibler (KL) divergence between $p_\tau(z' \mid \boldsymbol{u})$ and $D_\tau^\phi(z' \mid \boldsymbol{u})$. This corresponds exactly to Eq. (48).

Using the above calculations, we now have access to both the target's generalization loss and its upper-bound based on the source's generalization loss and other key quantities of cross-modal fine-tuning such as feature alignment(**FA**), feature-label distortion(**FLD**) and target fitting(**TF**). The gap between the target's generalization loss and its upper-bound is visualized in Fig. 5 which shows that our error bound is sufficiently tight with small different gap observed consistently across a variety of target datasets. The averaged gap over all these target tasks is less than 29%.

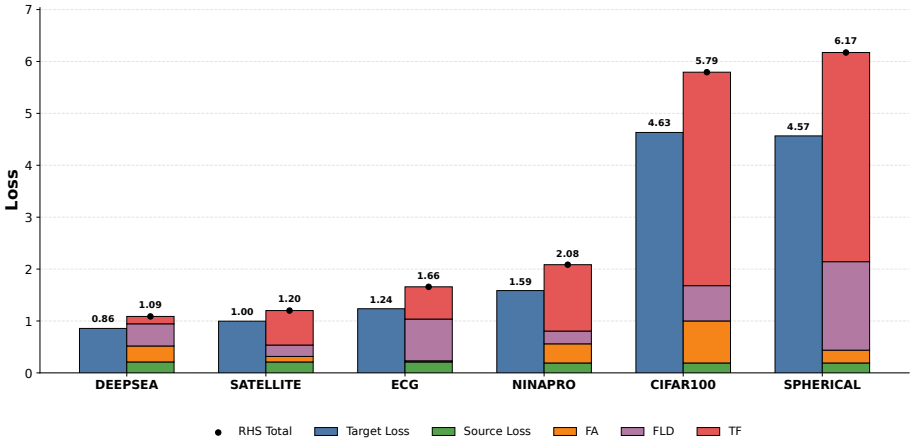


Figure 5: Bar charts illustrating the gap between the target's generalization loss and its upper-bound established in Theorem 7. For each target task, there are two bars representing the target's generalization loss and its upper-bound. The bar representing the upper-bound is further demarcated into the source's generalization loss (fine-tuning overhead), feature alignment(**FA**), feature Label distortion(**FLD**), and target fitting(**TF**). The bound evaluation is conducted at the learned target's feature map $\phi$ and its corresponding prediction head $p_\tau(z' \mid \boldsymbol{u})$.

## C Related Works

This section summarizes the existing literature on domain adaptation (Appendix C.1), in-modal fine-tuning (Appendix C.2), and cross-modal fine-tuning (Appendix C.3).

### C.1 Domain Adaptation

Domain adaptation is a form of transductive transfer learning in which the inputs and outputs of the source and target tasks are identical, but their corresponding data distributions are different (Pan and Yang, 2010; Wang and Deng, 2018). Its main focus is on transferring knowledge within the same task across different data distributions (i.e., domains) assuming shared input/output spaces. It also assumes that target inputs are accessible during training (often in the absence of target labels) to enable distribution alignment in a transductive setting. Furthermore, prior to the advent of foundation models (FMs), algorithmic designs for domain adaptation were tailored to specific task (i.e., input/output spaces) and source model's architecture, limiting their applicability to other scenarios. In contrast, fine-tuning in the FM paradigm enables model-agnostic adaptation procedures that operate without assuming specific forms of input/output or requiring access to target data during training. While some recent efforts in domain adaptation have relaxed assumptions on shared input/output spaces (Hoang et al., 2020; Lam et al., 2021; Hoang and Hoang, 2024; Yin et al., 2024), they still require source and target inputs to originate from the same modality (e.g., images, text), which restricts their use in more general cross-modal transfer settings. Other recent approaches have considered text-to-image transfer but require access to the (unlabeled) target data during training, are restricted to the same task, use a specific model architecture catering to text and image modalities (Yao et al., 2019; Li et al., 2020). This is different from the broader

scheme of cross-modal fine-tuning which aims to enable transfer across different tasks and unseen modalities via model-agnostic procedures.

### C.2 In-Modal Fine-Tuning of FMs

The recent advent of large pre-trained or foundation models (FMs) has enabled flexible knowledge transfer to a wide range of downstream tasks under a unified, task-agnostic paradigm. This is commonly known as fine-tuning, which is model-agnostic and does not require shared input or output spaces. Such transfer is feasible because these FMs are trained on broad, diverse corpora (e.g., text, image, or audio) that often overlap semantically or structurally with the content of downstream datasets for in-modal tasks. As a result, fine-tuning has been successfully applied across domains such as vision (Kirillov et al., 2023; Liu et al., 2021b), video understanding (Bertasius et al., 2021)), language (Zheng et al., 2021; Yang et al., 2022; Ma et al., 2023), and speech (Radford et al., 2022; Li et al., 2021a). There are also multi-modal FMs (Radford et al., 2021; Alayrac et al., 2022; Kim et al., 2021) which were pre-trained to learn the embeddings of multiple modalities together but existing approaches to facilitate knowledge transfer from these models are still restricted to within the pre-trained modalities. Extending fine-tuning to cross-modal scenarios with unseen data and downstream tasks (e.g., fine-tuning vision/language FMs to solve complex physics problems) are however less explored as discussed next.

### C.3 Cross-Modal Fine-Tuning of FMs

Early attempts in cross-modal fine-tuning have focused on transferring language models to other modalities such as vision (Kiela et al., 2019; Tan and Bansal, 2019; Gu et al., 2022), DNA/protein sequences (Nguyen et al., 2023; Lin et al., 2023; Jumper et al., 2021). These provide initial evidence of the cross-modal transferability of FMs but their designs are nonetheless hand-tailored to specific target tasks and modalities rather than for general-purpose (Shen et al., 2023). More recently, a few general-purpose cross-modal fine-tuning frameworks have emerged with remarkable successes in finetuning existing vision (Liu et al., 2021b) or language (Liu et al., 2019) FMs to solve a broad and diverse set of unseen tasks and data modalities (Shen et al., 2023; Cai et al., 2024; Ma et al., 2024). In particular, ORCA (Shen et al., 2023) aligns source and target representation distributions using optimal transport before fine-tuning the entire network. PARE (Cai et al., 2024) alternatively introduces a gating mechanism to integrate source and target features during fine-tuning. MoNA (Ma et al., 2024) addresses modality representation alignment via a bi-level optimization framework: the inner loop learns the optimal target predictor for a given embedder, while the outer loop updates the embedder to align the source feature-label semantics with the combined representation and prediction under this predictor. This help reduces the misalignment between source's and target's feature-label structures while fitting to the target data. However, as noted in Section 1, these approaches largely adopt heuristic methods combining distributional feature alignment and target fitting to facilitate cross-modal knowledge transfer. Overall, these approaches lack a principled means to assess the impact of their heuristic strategies on the generalized target performance.

## D Technical Background on Wasserstein Distance

As the core technical block of our theoretical analysis is built on the Wasserstein distance, we provide a succinct definition of this distance (Appendix D.1) and its computation (Appendix D.2) below.

### D.1 Wasserstein Distance

The Wasserstein $p$-distance is a fundamental metric in the theory of optimal transport. It is used to measure the distance between probability measures defined on a metric space. Intuitively, it is the cost of transporting mass from one distribution to another with respect to a cost metric $\delta(\boldsymbol{u}_1, \boldsymbol{u}_2)$ measuring the distance between two locations. This concept is widely applied in fields such as probability theory, machine learning, and partial differential equations (Villani, 2008). Let $\big(\mathcal{U}, \delta\big)$ be a metric space that is a Polish space. For $p \in [1, +\infty]$, the Wasserstein-$p$ distance between two probability measures $\mu$ and $\nu$ on space $\mathcal{U}$ is:

$$W_\delta^p\big(\mu, \nu\big) \quad = \quad \inf_{\pi \in \Pi(\mu,\nu)} \Big(\mathbb{E}_{(\boldsymbol{u}_1, \boldsymbol{u}_2)\sim\pi}\big[\delta(\boldsymbol{u}_1, \boldsymbol{u}_2)^p\big]\Big)^{\frac{1}{p}} \tag{49}$$

For example, we denote the Wasserstein-1 distance between two probability measures $\mu$ and $\nu$ on $\big(\mathcal{U}, \delta\big)$ as:

$$W_\delta(\mu,\nu) \;=\; \inf_{\pi\in\Pi(\mu,\nu)} \Big(\mathbb{E}_{(\boldsymbol{u}_1,\boldsymbol{u}_2)\sim\pi}\big[\delta(\boldsymbol{u}_1,\boldsymbol{u}_2)\big]\Big) \tag{50}$$

$$\;=\; \inf_{\pi\in\Pi(\mu,\nu)} \int_{\mathcal{U}}\int_{\mathcal{U}} \delta(\boldsymbol{u}_1,\boldsymbol{u}_2)\pi(\boldsymbol{u}_1,\boldsymbol{u}_2)d\boldsymbol{u}_1 d\boldsymbol{u}_2 \;. \tag{51}$$

A key property of the Wasserstein-1 distance is the Kantorovich duality theorem, which provides a dual formulation of the distance in terms of Lipschitz functions. This duality is particularly useful in applications ranging from machine learning to functional analysis (Villani, 2008) and is stated below. For any $\tau_\delta > 0$, let $\mathrm{Lip}_{\tau_\delta}$ denote the set of functions $g : \mathcal{U} \to \mathbb{R}$ such that

$$\big|g(\boldsymbol{u}_1) - g(\boldsymbol{u}_2)\big| \;\leq\; \tau_\delta \delta\big(\boldsymbol{u}_1,\boldsymbol{u}_2\big)$$

for all $\boldsymbol{u}_1, \boldsymbol{u}_2 \in \mathcal{U}$. Then, the Wasserstein-1 distance is given by

$$W_\delta(\mu,\nu) \;=\; \frac{1}{\tau_\delta} \sup_{f\in \mathrm{Lip}_{\tau_\delta}} \Big(\mathbb{E}_{\boldsymbol{u}_1\sim\mu}\Big[g(\boldsymbol{u}_1)\Big] \;-\; \mathbb{E}_{\boldsymbol{u}_2\sim\nu}\Big[g(\boldsymbol{u}_2)\Big]\Big) \;. \tag{52}$$

### D.2 Computation of Wasserstein Distance

Computing the Wasserstein distance is achieved via solving a linear program with a computational complexity of $O(n^3)$ (Peyré and Cuturi, 2020). This is however impractical for large datasets. To address this, we adopt entropic regularization to introduce a penalty term $\epsilon H(\pi) = -\epsilon \mathbb{E}_\pi[\log(\pi)]$ with $\epsilon > 0$ which leads to the following augmented optimization:

$$\pi_\epsilon \;=\; \arg\min_{\pi\in\Pi(\mu,\nu)} \Big(\mathbb{E}_{(\boldsymbol{u}_1,\boldsymbol{u}_2)\sim\pi}\big[\delta(\boldsymbol{u}_1,\boldsymbol{u}_2)\big] \;+\; \epsilon H(\pi)\Big) \;. \tag{53}$$

This regularized problem is solved efficiently using the Sinkhorn algorithm, which iteratively scales the transport plan $\pi_\epsilon = \mathrm{diag}(u) K \mathrm{diag}(v)$, where $u, v$ are scaling vectors updated to satisfy the marginal constraints.

The algorithm has an overall complexity of $O(n^2)$ which offers significant speedup for large-scale problems (Peyré and Cuturi, 2020). For a practical implementation, we utilize the Sinkhorn algorithm as provided by the Python Optimal Transport (POT) library and use its default regularization parameter $\epsilon = 0.1$ to ensure a balance between computational efficiency and accuracy.

## E Additional Experimental Results and Ablation Studies

This section provides additional experimental results to supplement the main text results. These results include: (i) the performance of RECRAFT compared to specialized physic-informed methods such as PINN (Raissi et al., 2019) and FNO (Li et al., 2021b), as well as a baseline U-Net method (Ronneberger et al., 2015) on PDEBench as shown in Table 3; (ii) experiment results on NAS-Bench-360 (Tu et al., 2022) with additional details on error bars for each task, as shown in Table 4; (iii) experiment results on PDEBench with additional details on error bars for each task, as shown in Table 5 and Table 6.

We also provide the ablations across several tasks on NAS-Bench-360 (Tu et al., 2022) isolating contributions: NFT vs. FA-only vs. RECRAFT, as shown in Table 7. RECRAFT achieves the best performance across all tasks and highlights the contribution of feature-label distortion(FLD) loss.

To assess the sensitivity of $\omega$, which ensures the Lipschitz continuity of the loss function $\ell_s$ on the source dataset and balances the minimization of feature alignment (**FA**) and feature-label distortion (**FLD**), we evaluate the model's performance across different target datasets using varying values of $\omega$. Fig. 6 illustrates the sensitivity of $\omega$ across various target datasets. The results suggest that setting $\omega$ within the range of approximately 0.3 to 0.5 achieves an optimal balance between **FA** and **FLD** during optimization.

## F Practical Source Re-calibration to Ensure Lipschitz Constraint with $\ell_2$ Metric

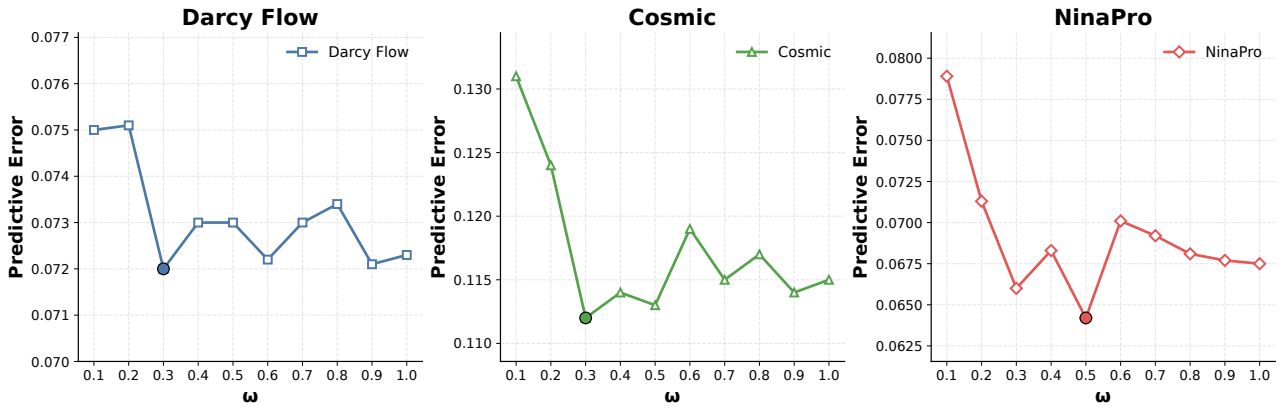


Figure 6: Predictive error (↓) of Darcy Flow, Cosmic and Ninapro across various values of $\omega$, which achieves the balance between minimizing Feature Alignment(**FA**) and Feature Label Distortion(**FLD**).

Table 5: Predictive error (↓) (with standard deviation) achieved by our proposed method RECRAFT and other specialized physic-informed baselines across multiple PDE tasks. U-Net results for Naiver-Stokes and Darcy Flow are missing because the benchmark paper (Takamoto et al., 2022) does not evaluate them (due to memory issues).

| **Model** | **Darcy** nRMSE | **Advection** nRMSE | **Burgers** nRMSE | **Diff.Sorp** nRMSE | **S.Water** nRMSE | **Diff.Reac**(2D) nRMSE | **Diff.Reac**(1D) nRMSE | **Navier-Stokes** nRMSE |
|---|---|---|---|---|---|---|---|---|
| PINN | $0.180 \pm 2\text{E-}3$ | $0.67 \pm 0.03$ | $0.36 \pm 0.03$ | $0.15 \pm 0.03$ | $0.085 \pm 3\text{E-}3$ | $0.840 \pm 0.01$ | $0.840 \pm 0.01$ | $0.720 \pm 5\text{E-}3$ |
| FNO | $0.220 \pm 3\text{E-}3$ | $0.011 \pm 1\text{E-}3$ | $\mathbf{0.003}1 \pm \mathbf{5E\text{-}5}$ | $1.8\text{E-}3 \pm 4\text{E-}5$ | $\mathbf{4.4E\text{-}3 \pm 3E\text{-}5}$ | $\mathbf{0.120} \pm \mathbf{4E\text{-}4}$ | $\mathbf{1.4E\text{-}3 \pm 1E\text{-}4}$ | $0.068 \pm 2\text{E-}3$ |
| U-Net | - | $1.1 \pm 0.21$ | $0.99 \pm 0.02$ | $0.22 \pm 0.02$ | $0.017 \pm 2\text{E-}3$ | $1.6 \pm 7\text{E-}3$ | $0.08 \pm 4\text{E-}3$ | - |
| RECRAFT | $\mathbf{0.079 \pm 1E\text{-}3}$ | $\mathbf{0.0078} \pm 4\text{E-}4$ | $0.0108 \pm 3\text{E-}4$ | $\mathbf{1.6E\text{-}3 \pm 3E\text{-}4}$ | $5.4\text{E-}3 \pm 5\text{E-}5$ | $0.817 \pm 3\text{E-}3$ | $2.8\text{E-}3 \pm 2\text{E-}4$ | $\mathbf{0.050 \pm 3E\text{-}3}$ |

This section provides a concrete example of how the source's prediction map can be recalibrated to ensure Lipschitz constraint with a practical choice of $\delta$. Note that the theoretical bound in Theorem 7 holds with any metric, we can then choose $\delta$ to be a Euclidean distance, i.e., $\delta \equiv \ell_2$. Given the choice, we can now aim to minimize $\tau_\delta = \omega$ via recalibrating the source's prediction map. This is achieved via finding a smallest value of $\omega$ for which the source's prediction map can be refitted so that $\ell_s(\boldsymbol{u}) := -\mathbb{E}_{D_s^\theta(z|\boldsymbol{u})}[\log(p_s(z \mid \boldsymbol{u}))]$ is $O(\omega)$-Lipschitz with respect to the Euclidean metric $\delta \equiv \ell_2$ while preserving performance on the proxy dataset. To do this, we view $\omega$ as a hyper-parameter and run ablation experiments on the proxy dataset with different choices of $\omega$ over a pre-defined range $[0.1, 1.0]$ to determine its optimal choice. For each $\omega$, we recalibrate the source's prediction map via

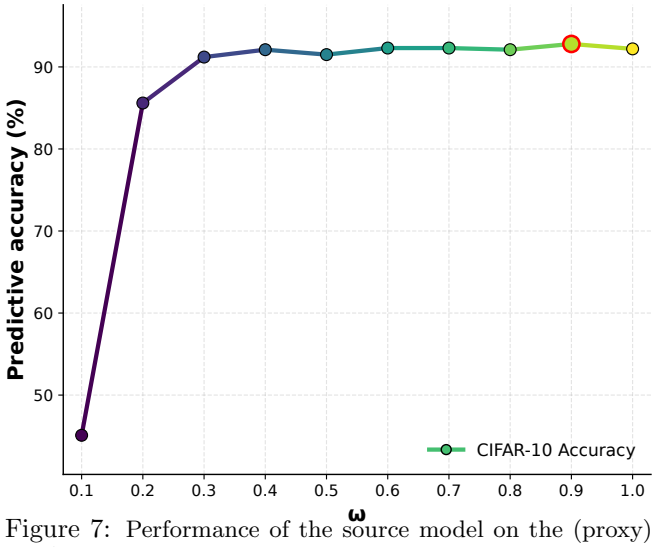


Figure 7: Performance of the source model on the (proxy) CIFAR-10 dataset with re-calibrated prediction head to satisfy the Lipschitz constraint in Definition 4 with constant $\tau_\delta = \omega$ where $\omega$ varies in $[0.1, 1.0]$.

$$\gamma(\omega) = \operatorname*{argmin}_{\gamma} \mathop{\mathbb{E}}_{\boldsymbol{x} \sim P_s} \max\left(0, \|\nabla_{\boldsymbol{u}} \ell_s(\theta(\boldsymbol{x}); \gamma)\|_2 - \omega\right)$$

with $\gamma$ denotes the last layer of $p_s(z \mid \boldsymbol{u})$. Once the re-calibrated $\gamma(\omega)$ has been computed, we re-evaluate the source performance to determine whether $\gamma(\omega)$ preserves performance and $\omega$ can be further reduced.

Our experiments show that smaller values of $\omega$ indeed lead to larger source loss values as they impose more constraints on the prediction map. Based on these ablation results, we selected $\omega = 0.3$ for 2D tasks; and $\omega = 0.5$ for 1D tasks. These are the minimum values of $\omega$ for which the re-calibrated prediction map preserves the source's model on the proxy dataset. To visualize this, we present Fig. 7 which plots the re-calibrated (source) model's performance on the proxy CIFAR-10 dataset versus Lipschitz threshold $\tau_\delta \triangleq \omega \in [0.1, 1.0]$ (for 2D tasks).

Table 6: Predictive error (↓) (with standard deviation) achieved by our proposed method RECRAFT and other state-of-the-art (SOTA) cross-modal fine-tuning baselines across multiple PDE tasks in PDEBench (Takamoto et al., 2022). The reported results of MoNA (Ma et al., 2024) are sourced from the corresponding paper due to its unavailable code.

| **Model** | **Darcy** nRMSE | **Advection** nRMSE | **Burgers** nRMSE | **Diff.Sorp** nRMSE | **S.Water** nRMSE | **Diff.Reac**(2D) nRMSE | **Diff.Reac**(1D) nRMSE | **Navier-Stokes** nRMSE |
|---|---|---|---|---|---|---|---|---|
| NFT | 0.085 ± 4E-3 | 0.0140 ± 2E-3 | 0.0130 ± 4E-4 | 3.1E-3 ± 7E-5 | 6.1E-3 ± 4E-5 | 0.830 ± 4E-3 | 9.2E-3 ± 2E-3 | 0.863 ± 2E-2 |
| ORCA | 0.081 ± 1E-3 | 0.0098 ± 2E-4 | 0.0120 ± 4E-4 | 1.8E-3 ± 2E-4 | 6.0E-3 ± 5E-5 | 0.820 ± 3E-3 | 3.2E-3 ± 2E-4 | 0.066 ± 2E-3 |
| PARE | 0.081 ± 1E-3 | **0.0032** ± 4E-4 | 0.0114 ± 3E-4 | 1.9E-3 ± 3E-4 | 5.9E-3 ± 5E-5 | 0.820 ± 5E-3 | 2.9E-3 ± 3E-4 | 0.068 ± 3E-3 |
| MoNA | **0.079** | 0.0088 | 0.0114 | **1.6E-3** | 5.7E-3 | 0.818 | **2.8**E-3 | 0.054 |
| RECRAFT | **0.079** ± **1E-3** | 0.0078 ± 4E-4 | **0.0108** ± **3E-4** | **1.6E-3** ± **3E-4** | **5.4**E-3 ± **5E-5** | **0.817** ± **3E-3** | **2.8**E-3 ± **2E-4** | **0.050** ± **3E-3** |

Table 7: Prediction errors (↓) incurred by NFT(naive fine-tuning),FA(only optimize feature alignment) and RECRAFT (with standard deviation) across 10 diverse tasks on NAS-Bench-360. Our method RECRAFT achieves best performance all tasks, highlight the effectiveness of minimizing FLD.

| **Model** | **Darcy** Relative $\ell_2$ | **DeepSEA** 1- AUROC | **ECG** 1-$F_1$ score | **CIFAR100** 0-1 error (%) | **Satellite** 0-1 error (%) | **Spherical** 0-1 error(%) | **Ninapro** 0-1 error (%) | **Cosmic** 1- AUROC | **Psicov** $MAE_8$ | **FSD50K** 1-mAP |
|---|---|---|---|---|---|---|---|---|---|---|
| NFT | 7.4E-3 ± 1E-4 | 0.490 ± 2E-2 | 0.44 ± 0.03 | 9.74 ± 1.21 | 13.82 ± 0.24 | 55.76 ± 2.3 | 8.35 ± 0.4 | 0.17 ± 2E-2 | 1.92 ± 0.06 | 0.63 ± 0.01 |
| FA | 7.3E-3 ± 8E-5 | 0.290 ± 2E-3 | 0.29 ± 6E-3 | 7.80 ± 0.41 | 11.61 ± 0.20 | 29.85 ± 0.8 | 7.73 ± 0.4 | 0.16 ± 5E-3 | 1.92 ± 0.04 | 0.57 ± 0.01 |
| Ours | **7.2**E-3 ± **7**E-5 | **0.278** ± **2E-3** | **0.27** ± **6E-3** | **7.30** ± **0.4** | **11.11** ± **0.04** | **26.41** ± **0.7** | **6.60** ± **0.3** | **0.11** ± **3E-3** | **0.99** ± **0.02** | **0.55** ± **0.01** |

## G RECRAFT algorithm

For better clarity, this section provides the pseudocode for our algorithm RECRAFT previously described in Section 3. To summarize, RECRAFT adopts a two-stage approach that decomposes the bound minimization into (1) finding a feature map $\phi$ that minimizes the semantic gap $\mathbf{FA}(\phi, \theta) + \mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}[\mathbf{FLD}(\boldsymbol{u})]$ between the source and target tasks, thus maximizing transferability (Section 3.1); and (2) learning a predictor $p_\tau(z' \mid \boldsymbol{u})$ based on the learned $\phi$ (Section 3.2) via minimizing $\mathbb{E}_{D_\tau^\phi(\boldsymbol{u})}[\mathbf{TF}(\boldsymbol{u})]$. A pseudocode of RECRAFT is detailed in Algorithm 1.

**Algorithm 1 RECRAFT** Algorithm

**Input:** Pre-trained model $M_s \triangleq (\theta, p_s(z \mid \theta(\boldsymbol{x})))$, proxy dataset $(\boldsymbol{X}^s, \boldsymbol{z}^s) \sim D_s(\boldsymbol{x}, z)$, target dataset $(\boldsymbol{X}^\tau, \boldsymbol{z}^\tau) \sim D_\tau(\boldsymbol{x}', z')$, number of epochs $n_0$, $n_1$, $n_2$.
**Output:** The target model $M_\tau \triangleq (\phi, p_\tau(z' \mid \phi(\boldsymbol{x}')))$.
***Stage 1: Learning Feature Map*** *(Section 3.1)*
Initialize the target embedder $\phi$.
**for** *epoch* $= 1$ to $n_1$ **do**
  Compute $L_{\mathbf{FA}}(\phi)$ using Eq. 14.
  Update $\phi$ by minimizing $L_{\mathbf{FA}}(\phi)$.
**end for**
**for** *epoch* $= 1$ to $n_2$ **do**
  Compute $L_{\mathbf{FLD}}(\phi)$ using Eq. 21.
  Update $\phi$ by minimizing $L_{\mathbf{FLD}}(\phi)$.
**end for**
***Stage 2: Learning Target Predictor*** *(Section 3.2)*
Frozen $\phi$, initialize $p_\tau(z' \mid \boldsymbol{u})$ from $p_s(z \mid \boldsymbol{u})$.
**for** *epoch* $= 1$ to $n_0$ **do**
  Compute $L_{\mathbf{TF}}$ using Eq. 24.
  Update $p_\tau$ by minimizing $L_{\mathbf{TF}}$.
**end for**
**Return** $(\phi, p_\tau(z' \mid \boldsymbol{u}))$

## H Hyperparameters & Implementation Details

To ensure fair comparisons, we adopt the same hyperparameters as those used in **ORCA** (Shen et al., 2023) for model fine-tuning. These specific parameter settings are shown in Table 8 for Nas-Bench-360 (Tu et al., 2022)

and Table 9 for PDEBench (Takamoto et al., 2022). For detailed implementation, we construct the target model $M_\tau \triangleq (\phi, p_\tau(z' \mid \phi(\boldsymbol{x}')))$ by adapting a pre-trained foundation model $M_s \triangleq (\theta, p_s(z \mid \theta(\boldsymbol{x})))$ which is selected to be a Swin Transformer (Liu et al., 2021b) for 2D tasks; and a RoBERTa (Liu et al., 2019) for 1D tasks. To compute the FLD for a regression task, we discretize the continuous label space into 10 distinct classes, consistent with the hyperparameter settings used in ORCA (Shen et al., 2023).

All experiments are run on an NVIDIA RTX 2080 GPU and results are averaged over 5 independent runs. Our code repository can be found at `https://github.com/khiembk/RECRAFT`.

Table 8: Hyperparameter configurations used for the 10 tasks on NAS-Bench-360 (Tu et al., 2022). *FA epoch* and *FLD epoch* denote the number of training epochs used to minimize the feature alignment (**FA**) and feature-label distortion (**FLD**). The optimizers used in our experiments include SGD, Adam (Kingma and Ba, 2017) and AdamW (Loshchilov et al., 2017).

| **Hyperparameter** | **CIFAR100** | **Spherical** | **NinaPro** | **FSD50K** | **Darcy Flow** | **PSICOV** | **Cosmic** | **ECG** | **Satellite** | **DeepSEA** |
|---|---|---|---|---|---|---|---|---|---|---|
| Backbone | Swin | Swin | Swin | Swin | Swin | RoBERTa | Swin | RoBERTa | RoBERTa | RoBERTa |
| Batch size | 32 | 32 | 32 | 32 | 4 | 1 | 4 | 4 | 16 | 16 |
| Epoch | 60 | 60 | 60 | 100 | 100 | 10 | 60 | 15 | 60 | 13 |
| Accumulation | 32 | 4 | 1 | 1 | 1 | 32 | 1 | 16 | 4 | 1 |
| Optimizer | SGD | AdamW | Adam | Adam | AdamW | Adam | AdamW | SGD | AdamW | Adam |
| Learning rate | 1.00E-04 | 1.00E-04 | 1.00E-04 | 1.00E-04 | 1.00E-03 | 5.00E-06 | 1.00E-03 | 1.00E-06 | 3.00E-05 | 1.00E-05 |
| Weight decay | 1.00E-03 | 1.00E-01 | 1.00E-05 | 5.00E-05 | 5.00E-03 | 1.00E-05 | 0.00E+00 | 1.00E-01 | 3.00E-06 | 0.00E+00 |
| Label discretization | - | - | - | - | 10 | 10 | 10 | 10 | - | - |
| $\omega$ | 0.3 | 0.3 | 0.3 | 0.3 | 0.3 | 0.5 | 0.3 | 0.5 | 0.5 | 0.5 |
| FA epoch | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 |
| FLD epoch | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 2 | 4 | 2 |

Table 9: Hyperparameter configurations used for the 8 tasks on PDEBench (Takamoto et al., 2022). *FA epoch* and *FLD epoch* denote the number of training epochs used to minimize the feature alignment (**FA**) and feature-label distortion (**FLD**). The optimizers used in our experiments include SGD, Adam (Kingma and Ba, 2017) and AdamW (Loshchilov et al., 2017).

| **Hyperparameter** | **Advection** | **Burgers** | **RD1D** | **Diff-Sor** | **Navier-Stokes** | **Darcy-Flow** | **Shallow-Water** | **RD2D** |
|---|---|---|---|---|---|---|---|---|
| Backbone | RoBERTa | RoBERTa | RoBERTa | Swin | RoBERTa | Swin | Swin | Swin |
| Batch size | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Epoch | 200 | 200 | 200 | 200 | 200 | 100 | 200 | 200 |
| Accumulation | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Optimizer | Adam | Adam | SGD | AdamW | AdamW | AdamW | AdamW | Adam |
| Learning rate | 1.00E-04 | 1.00E-05 | 1.00E-03 | 1.00E-04 | 1.00E-04 | 1.00E-04 | 1.00E-04 | 1.00E-04 |
| Weight decay | 1.00E-05 | 1.00E-05 | 1.00E-05 | 0 | 1.00E-03 | 1.00E-05 | 0 | 1.00E-03 |
| Label discretization | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| $\omega$ | 0.5 | 0.5 | 0.5 | 0.3 | 0.5 | 0.3 | 0.3 | 0.3 |
| FA epoch | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 |
| FLD epoch | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |

# I Complexity Analysis

We provide the complexity analysis of our proposed method as follows:

**Stage 1: Learning Feature Map:** Computing the loss $L_{\mathbf{FA}}(\phi)$ in Eq. (14) incurs $O(f_\phi * |D_\tau| + f_\theta * |D_s| + |W_\delta|)$ processing cost where $f_\phi$ and $f_\theta$ are the forward costs of the source and the target embedder, respectively; $O(|W_\delta|)$ is the cost of computing the Wasserstein distance $W_\delta$. Therefore, learning the target embedder within $n_1$ epochs with loss $L_{\mathbf{FA}}(\phi)$ requires $O(f_\theta * |D_s| + n_1(f_\phi * |D_\tau| + |W_\delta| + b_\phi))$ where $b_\phi$ is the cost of backpropagation of the target embedder. Likewise, updating the target embedder within $n_2$ epochs with loss $L_{\mathbf{FLD}}(\phi)$ in Eq. (21) incurs $O(n_2(f_{p_s}|D_\tau| + t_{FLD} + b_{p_s,\phi}))$ where $f_{p_s}$ is the cost of attaining $z \sim p_s(z|\phi(\mathbf{x}'))$, $t_{FLD}$ is the cost of computing the conditional entropy in Eq. (21), and $b_{p_s,\phi}$ is the cost of backpropagating through the frozen source predictor and target embedder.

**Stage 2: Learning Target Predictor:** Updating the target predictor with the loss $L_{\mathbf{TF}}$ in $n_0$ epochs requires $O(n_0(f_{p_\tau}|D_\tau| + |D_\tau| + b_{p_\tau}))$ where $f_{p_\tau}$ and $b_{p_\tau}$ are forward and backpropagation cost of the target predictor.

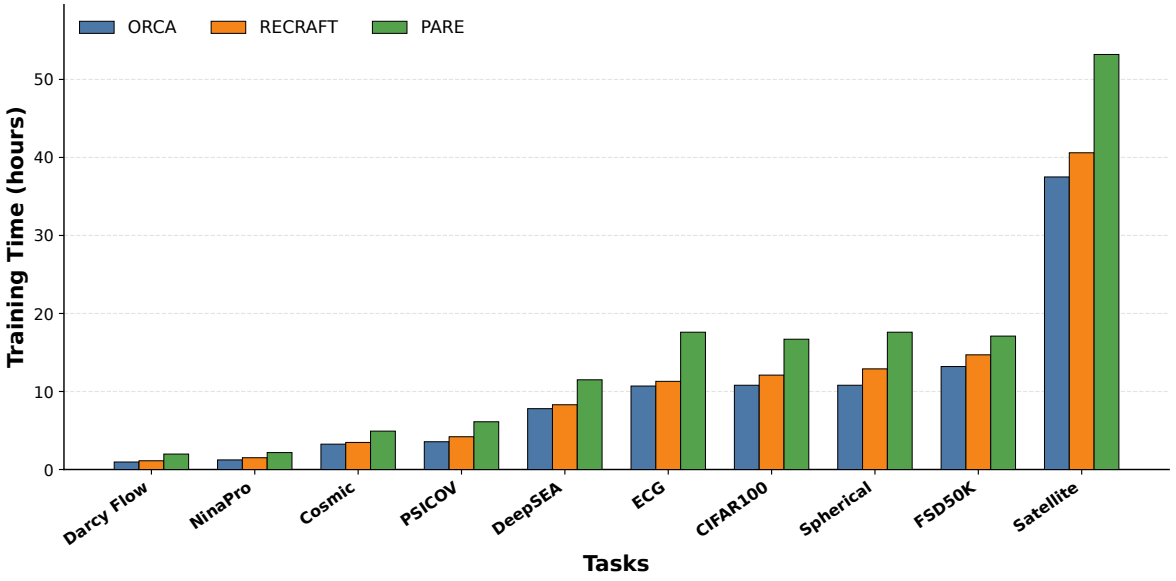


Figure 8: Plots of training time comparison on NAS-Bench-360 across a variety of baselines. RECRAFT achieves training times comparable to ORCA (Shen et al., 2023), while PARE (Cai et al., 2024) incurs significantly higher training times across all tasks.

Therefore, the total computational cost of our proposed algorithm, RECRAFT, is:

$$O(f_\theta * |D_s| + n_1(f_\phi * |D_\tau| + |W_\delta| + b_\phi)) \;+\; O(n_2(f_{p_s}|D_\tau| + t_{FLD} + b_{p_s,\phi})) \;+\; O(n_0(f_{p_\tau}|D_\tau| + |D_\tau| + b_{p_\tau}))$$

Compared to ORCA, RECRAFT introduces an additional computational cost of $O(n_2(f_{p_s}|D_\tau| + t_{FLD} + b_{p_s,\phi}))$ under identical settings for hyperparameters, number of training epochs, and Wasserstein distance $W_\delta$. However, this additional cost is empirically small, as demonstrated in Figure 5. Specifically, we compare the training efficiency (measured via training time) of RECRAFT against other state-of-the-art cross-modal fine-tuning baselines, including ORCA (Shen et al., 2023) and PARE (Cai et al., 2024), using the NAS-Bench-360 benchmark (Tu et al., 2022). MoNA (Ma et al., 2024) is excluded from the comparison due to the unavailability of its implementation. As shown in Figure 8, RECRAFT achieves training times comparable to ORCA while delivering superior performance. In contrast, PARE incurs significantly higher training times across all evaluated tasks. Notably, RECRAFT outperforms PARE on 7 out of 10 tasks in NAS-Bench-360 (Tu et al., 2022). These results suggest that RECRAFT offers the best trade-off between computational efficiency and fine-tuning performance.

## J Benchmark Information

NAS-Bench-360 (Tu et al., 2022) encompasses ten tasks categorized into three groups: 2D classification, 2D dense prediction, and 1D classification. These tasks span specialized modalities, including protein sequences (PSICOV), PDE solving (Darcy-Flow), audio processing (FSD50K), genetic data analysis (DeepSEA), and electrocardiogram signals (ECG), among others. Table 11 presents the details of each task, with a more comprehensive description available in the original paper (Tu et al., 2022).

Table 10: Data statistics of the 8 tasks in PDEBench (Takamoto et al., 2022).

| Task | Advection | Burgers | Diffusion-Reaction | Diffusion-Sorption | Navier-Stokes | Darcy-Flow | Shallow-Water | Diffusion-Reaction |
|---|---|---|---|---|---|---|---|---|
| Input Shape | 1D | 1D | 1D | 1D | 1D | 2D | 2D | 2D |
| Output Type | Dense | Dense | Dense | Dense | Dense | Dense | Dense | Dense |
| Resolution | 1024 | 1024 | 1024 | 1024 | 1024 | $128 \times 128$ | $128 \times 128$ | $128 \times 128$ |
| Parameters | $\beta = 0.4$ | $\nu = 1.0$ | $\nu = 0.5$, $\rho = 1.0$ | – | $\eta = 1.0$, $\zeta = 1.0$ | $\beta = 0.1$ | – | – |
| Loss | nRMSE | nRMSE | nRMSE | nRMSE | nRMSE | nRMSE | nRMSE | nRMSE |

Table 11: Data statistics of the 10 tasks in NAS-Bench-360 (Tu et al., 2022).

| | **CIFAR100** | **Spherical** | **NinaPro** | **FSD50K** | **Darcy Flow** | **PSICOV** | **Cosmic** | **ECG** | **Satellite** | **DeepSEA** |
|---|---|---|---|---|---|---|---|---|---|---|
| **# Training data** | 60K | 60K | 43956 | 51K | 1.1K | 3606 | 5250 | 330K | 1M | 250K |
| Input shape | 2D | 2D | 2D | 2D | 2D | 1D | 2D | 1D | 1D | 1D |
| Output type | Point | Point | Point | Point | Dense | Dense | Dense | Point | Point | Point |
| **# Classes** | 100 | 100 | 18 | 200 | – | – | – | 4 | 24 | 36 |
| Loss | CE | CE | LpLoss | MSELoss | BCE | FocalLoss | BCE | CE | CE | BCE |
| Expert Network | DenseNet-BC | S2CN | Attention Model | VGG | FNODE | EPCON | deepCR-mask | ResNet-1D | ROCKET | DeepSEA |

PDEBench (Takamoto et al., 2022) comprises multiple scientific datasets with simulated data from a wide variety of partial differential equations (PDEs) in physics. These include the 1D Advection equation, modeling linear advection with a constant speed parameter $\beta$; the 1D Burgers' equation, capturing non-linear fluid dynamics with a constant diffusion coefficient $\nu$; the 1D Diffusion-Reaction equation, combining diffusion and a source term governed by parameters $\nu$ and $\rho$; and the 1D Diffusion-Sorption equation, representing diffusion retarded by sorption, applicable to real-world scenarios. Additionally, the 1D Navier-Stokes equation describes compressible fluid dynamics with shear and bulk viscosities $\eta$ and $\zeta$. In two dimensions, the Darcy-Flow equation models steady-state flow over a unit square, scaled by a constant force term $\beta$; the Shallow-Water equations, derived from Navier-Stokes, address free-surface flow problems; and the 2D Diffusion-Reaction equation extends its 1D counterpart with two non-linearly coupled variables, presenting a complex challenge with significant real-world applications. Table 10 presents the details of each task, with a more comprehensive description available in the original paper (Takamoto et al., 2022).

## K Boarder Impacts

Our research provides a new analytical lens that opens several promising research directions:

**(i) Knowledge Distillation (KD).** The bound naturally extends to KD by decomposing the teacher–student gap into feature alignment (FA) and softened-label distribution mismatch (FLD). This perspective suggests a new class of KD objectives that jointly minimize both components, potentially yielding more effective and robust distillation compared to standard temperature-scaled KL or feature-mimicry losses.

**(ii) Retrieval-Augmented Generation (RAG) and Multimodal Retrieval.** The same decomposition can guide the alignment of retrieved documents or heterogeneous data modalities (e.g., audio, video, or scientific figures) within a shared representation space. This offers a theoretically grounded alternative to CLIP-style contrastive learning, with the potential for improved computational efficiency and robustness under label shift.

**(iii) Scaling to Foundation Models and LLMs.** Our framework highlights that explicitly accounting for FLD, beyond standard feature alignment, is critical for effective transfer. This insight provides a principled foundation for efficient cross-modal fine-tuning in large-scale models. Preliminary results on pre-trained Swin Transformer (90M parameters) and RoBERTa (88M+ parameters) demonstrate the feasibility of this approach and suggest a clear path toward scaling to modern foundation models.