Enhancing Cross-Modal Fine-Tuning with Gradually Intermediate Modality Generation

Lincan Cai [1] **Shuang Li** [1] **Wenxuan Ma** [1] **Jingxuan Kang** [2] **Binhui Xie** [1] **Zixun Sun** [3] **Chengwei Zhu** [3]

Abstract

Large-scale pretrained models have proven immensely valuable in handling data-intensive modalities like text and image. However, fine-tuning these models for certain specialized modalities, such as protein sequence and cosmic ray, poses challenges due to the significant modality discrepancy and scarcity of labeled data. In this paper, we propose an end-to-end method, `PaRe`, to enhance cross-modal fine-tuning, aiming to transfer a large-scale pretrained model to various target modalities. `PaRe` employs a gating mechanism to select key patches from both source and target data. Through a modality-agnostic **Pa**tch **Re**placement scheme, these patches are preserved and combined to construct data-rich intermediate modalities ranging from easy to hard. By gradually intermediate modality generation, we can not only effectively bridge the modality gap to enhance stability and transferability of cross-modal fine-tuning, but also address the challenge of limited data in the target modality by leveraging enriched intermediate modality data. Compared with hand-designed, general-purpose, task-specific, and state-of-the-art cross-modal fine-tuning approaches, `PaRe` demonstrates superior performance across three challenging benchmarks, encompassing more than ten modalities.

1. Introduction

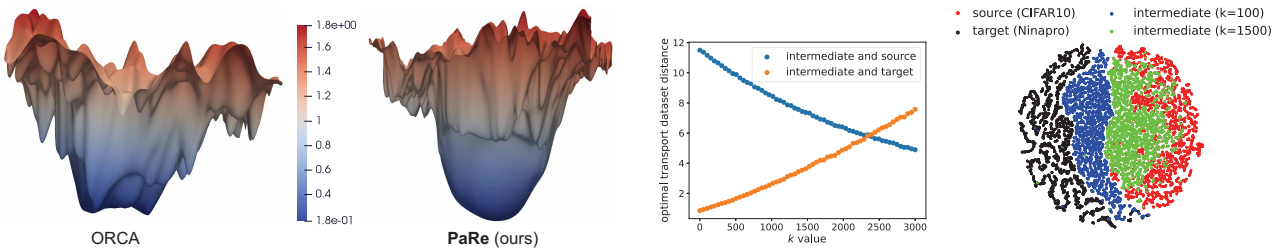
Multimodal perception, as a fundamental component of intelligence, is indispensable to realize artificial general intelligence. Recently, multimodal large language models (MLLMs) (Alayrac et al., 2022; OpenAI, 2023; Anil et al., 2023; Driess et al., 2023; Bai et al., 2023; Zhu et al., 2023a; Peng et al., 2024; Li et al., 2023b; Bao et al., 2023; Dong et al., 2024a) have effectively served as a versatile interface across various tasks encompassing vision, language, and multimodal tasks. While these models demonstrate remarkable performance and broad applicability in conventional modalities such as text, images, and videos, their development demands billions of high-quality data, substantial computational resources, and advanced training techniques. In certain professional field where data is scarce and modality-specific, the need for specialized models becomes apparent.

Fine-tuning contirbutes to this issue by transferring knowledge of large-scale pretrained models to downstream tasks, where data modality usually maintain consistent (Touvron et al., 2023; Howard & Ruder, 2018; Pan & Yang, 2010; Goyal et al., 2023; Dehghani et al., 2023; Jia et al., 2022; Bertasius et al., 2021). Recent studies have revealed the feasibility of fine-tuning pretrained models for unseen modalities (Moor et al., 2023; Yang et al., 2024; Shen et al., 2023; Lin et al., 2023; Pang et al., 2024). For instance, one could utilize a pretrained vision or language model to tackle genomics tasks. Despite the potential of fine-tuning pretrained models across modalities, cross-modal fine-tuning encounters two challenges: a) *Modality gap*: The gap may arise from task mismatch (e.g., image classification and genetic prediction) and data heterogeneity. Adapting these tasks to facilitate fine-tuning while preserving meaningful representations is arduous. b) *Data scarcity*: Some tasks in certain modalities may suffer from limited labeled data due to the necessity for additional expertise or difficulties in data collection, thus hindering effective fine-tuning.

ORCA (Shen et al., 2023) has undertaken a preliminary exploration to confront the above challenges by employing a two-stage training approach. In the first stage, it enhances model transferability by reducing the optimal transport dataset distance (OTDD) (Alvarez-Melis & Fusi, 2020) between source and target modality data, aiming to minimize the discrepancy between the two modalities. Subsequently, in the second stage, full fine-tuning is conducted to adapt the pretrained model to the target modality, yielding promising results. Despite its efforts on reducing the OTDD across distinct modalities, the significant modality gap and data scarcity remain to be addressed. As illustrated

[1]Beijing Institute of Technology [2]University of Illinois Urbana-Champaign [3]Interactive Entertainment Group, Tencent. Correspondence to: Shuang Li <shuangli@bit.edu.cn>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.



(a) Loss landscape (b) OTDD (c) t-SNE visualization

Figure 1: (a) The loss landscapes of models fine-tuned with ORCA (Shen et al., 2023) and PaRe on the Ninapro dataset. (b) The OTDD (Alvarez-Melis & Fusi, 2020) between the intermediate modality with different k values and source or target modality respectively. (c) Target embeddings (**black** dots), intermediate modality embeddings obtained by replacing target patches with different number of source patches (**blue** and **green** dots), and source embeddings (**red** dots) visualized using t-SNE. Intermediate modalities effectively bridge the modality gap and enhance the model's transferability and stability.

in Fig. 1 (a), we show the loss landscape (Li et al., 2018) of model fine-tuned with ORCA. It is evident that ORCA faces instability during training, potentially leading to suboptimal results as it is prone to getting trapped in unfavorable local optima. We conjecture that one reason is the alignment stage in ORCA, which can reduce the modality gap but not completely eliminate it. Another factor is that fine-tuning with limited data often leads to overfitting.

Driven by the above analysis, we propose an end-to-end approach to promote cross-modal fine-tuning through gradually generating intermedia modality data and bridging distinct modalities. Motivated by traditional data augmentation techniques like Mixup (Zhang et al., 2017) and CutMix (Yun et al., 2019), we achieve this by mixing source and target data to generate diverse intermediate modalities. However, due to the diversity between source and target modality data, applying mixing operations directly applied in the raw space is not feasible, and the substantial differences between modalities also increase the risk of model confusion. Here, we thoroughly analyze the characteristics of different modalities and design a modality-agnostic **Pa**tch **Re**placement (PaRe) method to construct intermediate modalities. Initially, data from source and target modalities are seperately mapped to a unified dimensional space using distinct, specific embedders, facilitating mixing operations at the embedding level. Subsequently, each patch within the source and target samples undergoes scoring through a gate network. A higher score signifies the patch's increased importance and its contribution to the model's classification of the respective sample. Consequently, we select top-k scored source patches to replace bottom-k scored target patches, thereby maximizing the preservation of crucial information in the intermediate modality data for both source and target data.

Take a step further, we facilitate the model's gradual progression from easier intermediate modalities to more challenging ones, ultimately adapting it to the target modality. This curriculum learning process (Bengio et al., 2009) can enhance the stability of the cross-modal fine-tuning. Specifically, we use the OTDD (Alvarez-Melis & Fusi, 2020) between the generated intermediate modality data and the source modality data as a metric to gauge the difficulty level. As Alvarez-Melis & Fusi (2020) analysed, a smaller OTDD between source and target dataset indicates higher transferability of the source model to the target dataset. The Fig. 1 (b)-(c) reveal that as the k value increases, the OTDD between the intermediate modality data and the source modality data decreases, enhancing the transferability of the source pretrained model. Thus, we progressively decrease the k value during training to construct intermediate modalities ranging from easy to hard. Recall that, as shown in Fig 1 (a), the loss landscape of the model fine-tuned using PaRe appears smoother and is less prone to getting stuck in unfavorable local optima compared to ORCA (Shen et al., 2023). This phenomenon is attributable to the generation of intermediate modalities and the transition from easy to hard.

In a nutshell, our contributions are summarized as follows:

- We propose an end-to-end cross-modal fine-tuning framework that is able to adapt a pretrained source model to any target modality. Leveraging the designed gradually intermediate modality generation, one can bridge the modality gap and alleviate the issue of insufficient data in the target modality, enhancing the model's transferability and stability.
- We design a modality-agnostic **Pa**tch **Re**placement (PaRe) method to construct intermediate modalities. By using a gate network for patch scoring, we extract pivotal patches from embeddings of both source and target modalities, blending them to facilitate a smoother training process with intermediate modalities.
- We validate the effectiveness of PaRe on three benchmarks comprising 48 datasets. In the most challenging NAS-360-Bench benchmark which contains 10 modalities, our PaRe significantly outperforms other approaches, including task-specific, general-purpose, and cross-modal fine-tuning, across all datasets.

2. Related Work

Mutil-modal transformers. Transformers (Vaswani et al., 2017) is first used successfully for natural language processing. With the rapid success of large language models (LLMs) (Devlin et al., 2018; Raffel et al., 2020; Zhang et al., 2022; Brown et al., 2020; Touvron et al., 2023; Chiang et al., 2023), researchers have started aligning multimodal data with LLMs (Radford et al., 2021; Alayrac et al., 2022; OpenAI, 2023; Bao et al., 2023; Wang et al., 2023a; Sun et al., 2024; Hong et al., 2023; Chen et al., 2023; Ye et al., 2023; Li et al., 2023a; Han et al., 2023; Dong et al., 2024b; Wei et al., 2023; Shukor et al., 2023). These general-purpose models excel in perceiving data-rich modalities (e.g., image, video, audio, text), following instructions, and learning in context. However, in many specialized domains where the data is scarced, a well-adapted specialized model is needed.

In-modality fine-tuning. Pretrained models are widely used in fields like vision (e.g., dense prediction (Kirillov et al., 2023; Liu et al., 2021) and 3D understanding (Bertasius et al., 2021; Luo et al., 2022a)), language (e.g., cross-lingual learning (Zheng et al., 2021; Yang et al., 2022; Ma et al., 2023) and parameter-efficient (Hu et al., 2022; Houlsby et al., 2019)), and speech (Radford et al., 2023; Li et al., 2021a). This line of methods endeavor to transfer knowledge learned during the pre-training process and data modality of downstream tasks always within seen modalities. But, whether one can transfer from one modality to another irrelevant modality is still under-explored. Imagine harnessing the power of pretrained vision transformers not just for image classification, but for unraveling the intricacies of physics puzzles.

Cross-modality fine-tuning. Adapting pretrained models to other modalities and tasks has been demonstrated in recent works (Tan & Bansal, 2019; Pang et al., 2024; Gu et al., 2022; Shen et al., 2023). In particular, LLMs have been employed in the life sciences to translate between text and chemistry (Edwards et al., 2021), biology (Luo et al., 2022b), medical (Moor et al., 2023), DNA-sequencing (Nguyen et al., 2023), and protein sequences (Lin et al., 2023) and folding (Jumper et al., 2021)). In contrast to the aforementioned approaches, this work aims to fine-tune pretrained models initially trained on general modalities like vision or language, on specialized modalities with scarce data. Shen et al. (2023) have taken the first step in this direction via alignment at the embedding space then fine-tuning the whole network. Different from previous works, our work progressively constructs different intermediate modalities during the training process, which enhances the model's transferability and training stability.

Curriculum learning. Curriculum Learning (Bengio et al., 2009; Zhou & Bilmes, 2018) promotes the strategy of learning from easier samples first and harder samples later. This idea has been widely explored in training neural networks (Hacohen & Weinshall, 2019; Wang et al., 2023b), reinforcement learning (Narvekar et al., 2020; Klink et al., 2022) and transfer learning (Weinshall et al., 2018; Zhang et al., 2021). In this work, we intend to narrow the substantial modality gap between source and target data. And we adopt a step-by-step approach, starting with simpler intermediate modalities and gradually moving towards more complex ones. This progressive generation ultimately enhances the stability of the model's cross-modal fine-tuning process to better align with the target modality.

Cross-modality mixing. Mixup (Zhang et al., 2017) is a commly used and effective technique for data augmentation. There are two main categories: Global Mixup, exemplified by methods like Mixup (Zhang et al., 2017), UniMix (Shen et al., 2022b), Manifold-Mixup (Verma et al., 2019) and PatchMix (Zhu et al., 2023b), and Region Mixup, represented by CutMix (Yun et al., 2019), Saliencymix (Uddin et al., 2021) and TransMix (Chen et al., 2022). All of these methods only involve Mixup for uni-modal data. As for cross-modality, VLMixer (Wang et al., 2022) employs modality-agnostic augmentation to create semantically invariant cross-modal inputs which proficiently merging visual tokens with non-grounded linguistic tokens. There is semantic correlation among cross-modal data in VLMixer. However, our work focuses on exploring a method akin to Mixup technique for unpaired cross-modal data, where no semantic correlation between modalities and significant modality gap exists. Through the patch replacement approach we designed, we construct data-rich intermediate modalities, bridging the gap between modalities to enhance the transferability of the source pretrained model.

3. Method

Problem setup. A modality $\mathcal{M}$ contains a feature space $\mathcal{X}$, a label space $\mathcal{Y}$, and a joint probability distribution $P(\mathcal{X}, \mathcal{Y})$. In this paper, we focus on a more difficult cross-modal setting that the feature space $\mathcal{X}^t$, label space $\mathcal{Y}^t$ and joint probability distribution $P(\mathcal{X}^t, \mathcal{Y}^t)$ in the target modalities $\mathcal{M}^t$ are all different from those in the source (pretrained) modality $\mathcal{M}^s$, i.e., $\mathcal{X}^t \neq \mathcal{X}^s$, $\mathcal{Y}^t \neq \mathcal{Y}^s$, and $P^t(\mathcal{X}^t, \mathcal{Y}^t) \neq P^s(\mathcal{X}^s, \mathcal{Y}^s)$ (e.g., natural images vs. PDEs).

Our goal is to adapt the model pretrained from the source modality $\mathcal{M}^s$ to the target modality $\mathcal{M}^t$ in a supervised manner. In contrast to the two-stage fine-tuning approach employed by ORCA (Shen et al., 2023), we design an end-to-end fine-tuning approach named `PaRe`. This approach involves constructing intermediate modalities through patch

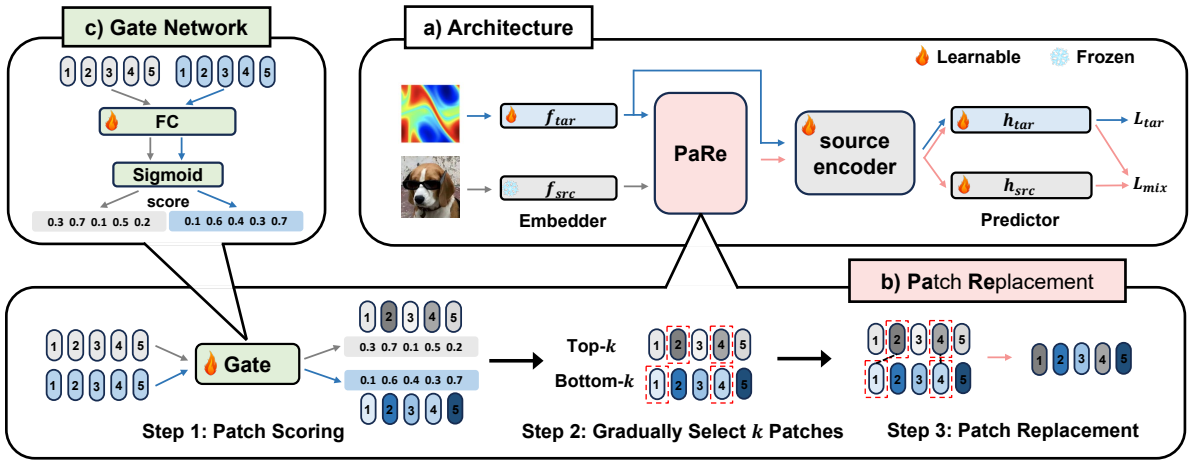


Figure 2: Framework overview. a) The overall architecture of the model and workflow of our method. b) **Pa**tch **Re**placement (`PaRe`) module contains three steps: patch scoring using the designed gate network, gradually select top-k source patches and bottom-k target patches and replace the selected target patches with the source patches one by one. c) The architecture of the gate network which contains a Full-Connected (FC) layer and a Sigmoid layer.

scoring and gradually patch replacement, addressing the gap between the source and target modalities in a progressive way. This approach effectively enhances the model transferabiltiy and training stability, while also alleviating the issue of insufficient training of model due to the limited data for target modality. Fig. 2 illustrates the workflow of `PaRe`. In the following subsections, we will provide a detailed overview of each module in `PaRe`.

3.1. Architecture design

In order to apply the source modality pretrained models to the target modality, we follow ORCA (Shen et al., 2023), which decomposes a transformer-based model into three parts: an embedder f, a feature encoder g and a predictor h, and then employs a pretrained architecture and weights to initialize the feature encoder g.

Source and target embedder. Since our approach requires patch-level replacement of source and target embeddings, serving as inputs to the feature encoder, we need to design source and target-specific embedders to map them to the same dimension. We can denote f^s as the pretrained source embedder, which transforms the source raw data $\mathcal{X}^s$ into sequence source embeddings $\tilde{\mathcal{X}}^s = \mathbb{R}^{N\times D}$ (where N denotes the embeddings length and D denotes the embedding dimension). The target embedder f^t is randomly initialized and designed to process target inputs $\mathcal{X}^t$ of arbitrary dimension and transform them to $\tilde{\mathcal{X}}^t = \mathbb{R}^{N\times D}$. Subsequently, we obtain the mixed embedding $\tilde{\mathcal{X}}^m$ through Patch Scoring and Patch Replacement, as proposed in Sec 3.3.

Custom predictor. Different modalities may correspond to different tasks, various tasks typically entail distinct types of outputs, such as classification logits in $\mathbb{R}^K$, or dense map where the spatial dimension aligns with the input, and per-index logits correspond to K classes. For classification, we follow ORCA (Shen et al., 2023) to utilize average pooling along the sequence length dimension, resulting in 1D tensors with a length of D. Subsequently, to make seperate predictions for the feature $g(\tilde{\mathcal{X}}^t)$ of target modality data and the feature $g(\tilde{\mathcal{X}}^s)$ of the intermediate modality data, we randomly initialize two classifiers h^t and h^s to map D to K such as the target prediction $p^t = h^t(g(\tilde{\mathcal{X}}^t))$. In the case of dense prediction tasks, we also follow ORCA's configuration to shape the tensor to the desired output dimension.

3.2. Gradually intermediate modality generation

Adapting a pretrained source model to various target modalities encounters two issues. The first is the substantial disparity between source and target modalities, which is quantified by the optimal transport dataset distance (OTDD) (Alvarez-Melis & Fusi, 2020) in this work. A larger OTDD signifies greater divergence between modalities, leading to diminished model transferability. The second issue relates to the inadequate training of the model due to the limited amount of data available in the target modality.

Hence, during the end-to-end training process, we progressively construct intermediate modality data, transitioning from resembling the source modality (easier) to resembling the target modality (harder). Through these intermediate modality data, we bridge the modality gap and alleviate the issue of in-sufficient target data, thereby enhancing the model transferability and the training stability. This method provides a simple yet effective solution to the aforementioned challenges.

3.3. Modality-agnostic patch replacement

Motivated by in-modality transfer learning, constructing an intermediate domain serves as an effective approach to bridge the domain gap. They often linearly combine images from different domains (such as Mixup (Zhang et al., 2017)) to generate intermediate domain data.

Nevertheless, creating intermediate modality data proves to be more challenging for diverse modalities. The varying input dimensions between target and source modality data hinder the application of methods such as Mixup (Zhang et al., 2017) or CutMix (Yun et al., 2019) in raw space. While it may be intuitive to consider these approaches in a unified embedding space, the substantial differences between modalities increase the risk of model confusion through direct Mixup. Furthermore, employing region-based replacement methods like CutMix may not produce favorable outcomes for diverse modality data. The reason is that unlike image modality data where semantic information is often concentrated in the middle region, some modalities, such as PDE data, may harbor more critical information at the edges. Consequently, we opt for a modality-agnostic patch replacement approach to construct intermediate modalities which can maximize the preservation of key information in the intermediate modality for both source and target modalities.

Patch scoring with gate network. For two modalities without semantic correlation, the most straightforward way of patch replacement is to randomly select patches for replacement. But, random patch replacement may lead to instances where non-semantic patches from source modality data replace crucial patches from target modality data, disrupting the model's training. Therefore, we design a patch selected strategy called **Patch Scoring** using a **Gate Network** such that, in the mixed embeddings after the patch replacement, crucial information from both source and target data is well preserved. This preservation is essential for effective model classification and efficient training.

Formally, we denote the source embeddings $\tilde{\mathcal{X}}^s = \mathbb{R}^{N\times D}$ contains N patches that $\tilde{\mathcal{X}}^s = \{\tilde{x}_1^s, \tilde{x}_2^s, ..., \tilde{x}_N^s\}$ and the target embeddings $\tilde{\mathcal{X}}^t = \mathbb{R}^{N\times D}$ contains N patches that $\tilde{\mathcal{X}}^t = \{\tilde{x}_1^t, \tilde{x}_2^t, ..., \tilde{x}_N^t\}$. We score each patch $\tilde{x}_i^s$ from source and $\tilde{x}_i^t$ from target using a gate network with a fully-connected (FC) layer and a sigmoid (σ) layer $\mathcal{S}^s = \sigma(FC(\tilde{\mathcal{X}}^s))$, $\mathcal{S}^t = \sigma(FC(\tilde{\mathcal{X}}^t))$, the higher the score, the more critical information the patch contains that contributes to the model's classification (e.g., in an image of a dog, a patch containing the dog's eyes would score higher than a patch containing the background). Then, we keep the positions of the top(N-k) target patches with the highest scores fixed and replace the bottom(k) target patches with the lowest scores with the top(k) source patches with the highest scores. To enable the gradient backward properly to update the gate network, we opt for using Gumble Softmax (Jang et al., 2016) approach to achieve the selection. Hence, we can obtain the mixed embeddings $\tilde{\mathcal{X}}^m$ of the intermediate modality which contains the key information of both source and target data.

Moreover, the parameter k linearly decreases with the number of training epochs to facilitate the transition of the intermediate modality from the source to the target. For the labels of intermediate modality data, the calculation involves taking the weighted sum of the labels y^s from the source data and y^t target data. However, due to the disparate distributions of y^s and y^t, this process is transformed into a weighted sum for the loss. Therefore, we can denote that the predictions of the intermediate modality data as $p^{ms} = h^s(g(\tilde{\mathcal{X}}^m))$ for source and $p^{mt} = h^t(g(\tilde{\mathcal{X}}^m))$ for target. The weight λ can be calculated by $\lambda = \frac{k}{N}$. Finally, we can calculate the mixed loss $\mathcal{L}_{mix}$ using mixed embeddings $\tilde{\mathcal{X}}^m$ as the inputs:

$$\mathcal{L}_{mix} = (1-\lambda)\mathcal{L}_{tar}(p^{mt}, y^t) + \lambda\mathcal{L}_{src}(p^{ms}, y^s), \quad (1)$$

where $\mathcal{L}_{tar}$ is the task-specific loss for different target modalities and $\mathcal{L}_{src}$ is the CrossEntropyLoss for the source modality. The total loss of our method can be defined as:

$$\mathcal{L}_{total} = \beta_1\mathcal{L}_{tar}(p_t, y_t) + \beta_2\mathcal{L}_{mix}, \quad (2)$$

where β_1, β_2 are trade-off parameters. We summarize our `PaRe` in Alg. 1 in the Appendix A.1.

4. Experiments

In this section, we first validate the effectiveness of `PaRe` for cross-modal fine-tuning on three benchmarks: NAS-Bench-360, PDEBench and OpenML-CC18, comprising a total of 48 datasets. Subsequently, through a series of analytical experiments, we showcase the superiority of each module within `PaRe` when compared to alternative approaches. Finally, by presenting intuitive visualization results, we illustrate the effectiveness of our gate network and the successful preservation of source knowledge in `PaRe`.

Implementation details. We follow ORCA (Shen et al., 2023) in using RoBERTa (Liu et al., 2019) and Swin Transformers (Liu et al., 2021) as pretrained source models for 1D/2D modalities respectively, treating language and vision as the source modalities. For 2D classification tasks, CIFAR10 (Krizhevsky et al., 2009) and Tiny-ImageNet (Le & Yang, 2015) serve as proxy datasets. For 2D dense prediction tasks, we use VOC (Everingham et al., 2015) as a proxy dataset, modifying its labels to create a simpler foreground-background segmentation task. For 1D tasks, CoNLL-2003 is employed as a proxy dataset. For other experimental settings such as learning rates, number of epochs, optimizers, we adhere to the configurations specified by ORCA. Our experiments are conducted in a single NVIDIA RTX 4090.

Table 1: Prediction errors (↓) across 10 diverse tasks on NAS-Bench-360. "FPT" and "NFT" respectively represent fine-tuning only the layer normalization of the model and performing one-stage full fine-tuning of the model.

	CIFAR-100 0-1 error (%)	Spherical 0-1 error (%)	Darcy Flow relative ℓ_2	PSICOV MAE_8	Cosmic 1-AUROC	NinaPro 0-1 error (%)	FSD50K 1-mAP	ECG 1-F1 score	Satellite 0-1 error (%)	DeepSEA 1-AUROC
Hand-designed	19.39	67.41	8.00E-03	3.35	0.127	8.73	0.62	0.28	19.80	0.30
NAS-Bench-360	23.39	48.23	2.60E-03	2.94	0.229	7.34	0.60	0.34	12.51	0.32
DASH	24.37	71.28	7.90E-03	3.30	0.190	6.60	0.60	0.32	12.28	0.28
Perceiver IO	70.04	82.57	2.40E-02	8.06	0.485	22.22	0.72	0.66	15.93	0.38
FPT	10.11	76.38	2.10E-02	4.66	0.233	15.69	0.67	0.50	20.83	0.37
NFT	7.67	55.26	7.34E-03	1.92	0.170	8.35	0.63	0.44	13.86	0.51
ORCA	6.53	29.85	7.28E-03	1.91	0.152	7.54	0.56	**0.28**	11.59	0.29
PaRe	**6.25**	**25.55**	**7.00E-03**	**0.99**	**0.121**	**6.53**	**0.55**	**0.28**	**11.18**	**0.28**

Table 2: Normalized Root Mean Squared Errors (nRMSEs, ↓) across 8 tasks of PDEBench. PaRe surpasses U-Net and PINN in all tasks, outperforms ORCA in 6 out of 8 tasks, and exhibits performance comparable to FNO.

	Advection 1D	Burgers 1D	Diffusion-Reaction 1D	Diffusion-Sorption 1D	Navier-Stokes 1D	Darcy-Flow 2D	Shallow-Water 2D	Diffusion-Reaction 2D
PINN	6.70E-01	3.60E-01	6.00E-03	1.50E-01	7.20E-01	1.80E-01	8.30E-02	8.40E-01
FNO	1.10E-02	**3.10E-03**	**1.40E-03**	1.70E-03	6.80E-02	2.20E-01	**4.40E-03**	**1.20E-01**
U-Net	1.10E+00	9.90E-01	8.00E-02	2.20E-01	-	-	1.70E-02	1.60E+00
ORCA	9.80E-03	1.20E-02	3.00E-03	**1.60E-03**	**6.20E-02**	8.10E-02	6.00E-03	8.20E-01
PaRe	**2.70E-03**	8.30E-03	2.60E-03	**1.60E-03**	6.62E-02	**8.06E-02**	5.70E-03	8.18E-01

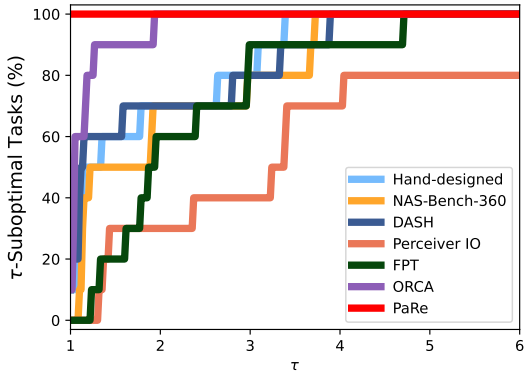


Figure 3: Aggregating Table 1 results using performance profiles (Dolan & Moré, 2002). The ordinate represents the cumulative distribution of problems solved by the method within a factor τ of the best performance. Therefore, the closer a curve approaches the top-left corner of the graph, the more capable the method is of solving more problems with minimal performance degradation. PaRe being as a horizontal line means it is always the best.

4.1. Overall results

NAS-Bench-360 comprises four 2D classification tasks, three 2D dense prediction tasks, and three 1D tasks. Here, we compare four types of baselines: (1) task-specific models designed by (Tu et al., 2022); (2) general-purpose models exemplified by Perceiver IO (Jaegle et al., 2022); (3) AutoML methods featuring the top-performing algorithm on NAS-Bench-360, DASH (Shen et al., 2022a); (4) cross-modal fine-tuning methods including naive fine-tuning and ORCA.

Table 3: Average classification results across 30 datasets on OpenML-CC18. "Diff. from XGBoost" is the acrosstask average of per-task difference from XGBoost. On 15/30 datasets, PaRe ranks the first among all compared methods.

OpenML-CC18	LightGBM	CatBoost	XGBoost	AutoGluon	TabPFN	ORCA	PaRe
# Wins/Ties	1/30	1/30	2/30	7/30	5/30	7/30	**15/30**
Avg. AUROC (↑)	0.8840	0.8898	0.8909	0.8947	0.8943	0.8946	**0.9030**
Diff. from XGBoost	-0.0069	-0.0011	0	+0.0038	+0.0034	+0.0036	**+0.0121**

As shown in Table 1, PaRe achieves the best performance across all tasks. Whether hand-designed or AutoML method, we demontrate significant preformance gains compares to them across multiple tasks. Particularly, in the comparison of cross-modal fine-tuning methods, we outperform ORCA on nearly all tasks. Moreover, we make substantial progress on tasks where ORCA couldn't surpass hand-designed or AutoML methods, establishing a new state-of-the-art results.

In addition, we employ performance profiles to comprehensively compare multiple methods across a suite of datasets. Performance profiles are statistical tools used to assess and demonstrate the efficacy of optimization algorithms across a multitude of test cases. Each curve represents a different method and shows the proportion of problems it solves within varying thresholds of a performance factor τ. The performance factor τ is a normalized measure of how each method's performance compares to the best performance. As shown in Fig. 3, PaRe is always the best, which means PaRe outperforms other methods across all tasks.

Table 4: Comparison prediction errors (↓) of traditional mixing strategies and PaRe variants across 10 diverse tasks, and the impact of varying strategies for different values of k, where "non-gradual" indicates a constant k, while the other three represent different strategies for decreasing k.

	Method	CIFAR-100	Spherical	Darcy Flow	PSICOV	Cosmic	NinaPro	FSD50K	ECG	Satellite	DeepSEA
	Mixup	6.59	26.60	7.70E-03	**0.99**	0.500	7.74	0.56	0.29	11.51	0.29
	CutMix	**6.11**	27.76	7.20E-03	**0.99**	0.135	8.41	0.56	0.29	11.58	**0.28**
PaRe	w/ non-gradual	6.59	27.68	7.20E-03	**0.99**	0.138	7.59	0.57	0.28	11.61	0.29
	w/ piecewise	6.22	26.88	**6.90E-03**	**0.99**	0.132	6.98	0.56	0.29	**10.89**	0.29
	w/ exponential	6.38	26.35	7.00E-03	**0.99**	**0.119**	7.13	**0.55**	**0.28**	11.56	**0.28**
	w/ linear (default)	6.25	**25.55**	7.00E-03	**0.99**	0.121	**6.53**	**0.55**	**0.28**	11.18	**0.28**

PDEBench comprises multiple scientific ML-related datasets, with a focus on the physics domain. Following ORCA, we validate PaRe on eight of these datasets. We compare our method with different SOTA task-specific models, including the physics-informed neural network PINN (Raissi et al., 2019), Fourier neural operator (FNO) (Li et al., 2021b), the generic image-to-image regression model U-Net (Ronneberger et al., 2015) and ORCA (Shen et al., 2023). As shown in Table 2, PaRe demonstrate further improvement over ORCA across multiple datasets, achieving results to be SOTA on nearly half of the datasets among all methods.

OpenML-CC18 benchmark is for tabular classification. We assess the performance of PaRe across 30 datasets on OpenNL-CC18 (Vanschoren et al., 2014). Our evaluation includes comparisons against the classical boosting methods XGBoost (Chen & Guestrin, 2016), CatBoost (Ostroumova et al., 2017) and LightGBM (Ke et al., 2017), deep learning approaches like AutoGluon (Erickson et al., 2020) and TabPFN (Hollmann et al., 2022) and cross-modal fine-tuning method ORCA. As shown in Table 3, the average accuracy of PaRe across 30 datasets is the highest among all methods, with the highest results observed in 15 out of the 30 datasets. Compared to ORCA, PaRe is better on 22 datasets. The detailed results can be found in the Appendix A.3.

In short, these results emphasize that our end-to-end strategy of constructing intermediate modalities for cross-modal fine-tuning, compared to two-stage alignment and fine-tuning of ORCA, can more efficiently close the modality gap, which enhances model transferability. In the following, we will systematically analyze the strengths of each module in PaRe.

4.2. Why using gradually patch replacement?

In this section, we conduct ablation studies to analyze the superiority of PaRe compared to other mixing techniques. Additionally, we investigate the impact of different gradual choices for the value of k.

Comparison of the mixing ways. There are various ways to perform mixing at the embedding space. The question is, which method can achieve modality-agnostic behavior and perform well on various target modalities. In Table 4, we compare our patch replacement method with patch Mixup and CutMix, which are the two most commonly used and robust methods. From the results, it is evident that our method outperforms Mixup and CutMix on almost all datasets. Particularly, on the Cosmic dataset, mixup fails to train and produces a random outcome. We attribute the superiority our method to thorough consideration of modality differences. When there is a significant modality gap, direct application of mixup way overly confuse the model, making training difficult. On the other hand, CutMix, which replaces patches in a block-wise manner, can potentially cover the critical information in the data, disrupting model

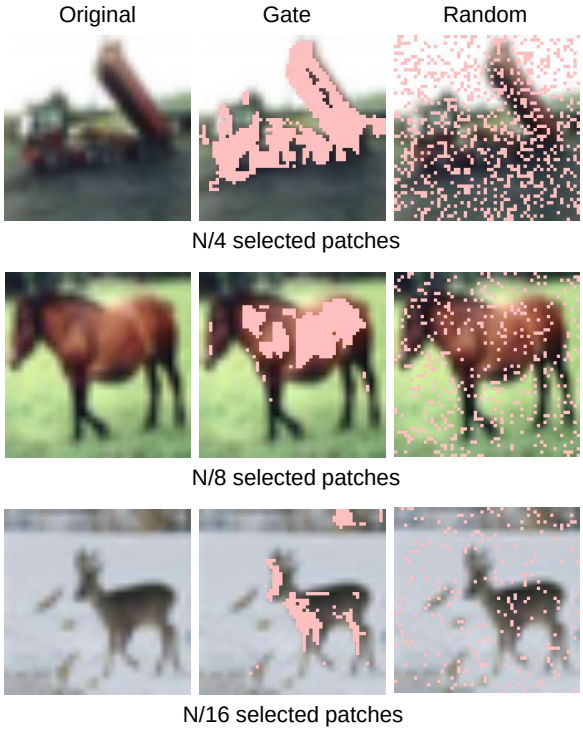


Figure 4: The visualization of the different numbers of patches selected by random strategy and our gate strategy. Additional visualizations can be found in the Appendix A.4.

Table 5: Comparison prediction errors (↓) between different strategies (Random vs. Gate) to select patches for replacement.

	CIFAR-100	Spherical	Darcy Flow	PSICOV	Cosmic	NinaPro	FSD50K	ECG	Satellite	DeepSEA
Random	6.52	28.06	7.10E-03	**0.99**	0.146	6.98	0.56	0.29	11.32	**0.28**
Gate	**6.25**	**25.55**	**7.00E-03**	**0.99**	**0.121**	**6.53**	**0.55**	**0.28**	**11.18**	**0.28**

training and yielding suboptimal results. Our patch replacement method ensures that the model is not overly confused. Additionally, our gating mechanism maximally preserves essential information from both modality data.

Comparison of different k-value choices. `PaRe` selects k patches from the source embedding to replace k patches in the target embedding. Next we analyze the strategy for choosing the value of k. We categorize the overall strategy into two types: non-gradual and gradual. For the non-gradual approach, the selection of k is random, while the gradual approach involves choosing k values that decrease in some form as the training progresses. As shown in Table 4, we observe that the use of gradual strategies generally outperforms the non-gradual strategy across various datasets. The model's progression from the source to the target modality through the intermediate modality constructed by `PaRe`, can be considered a form of curriculum learning that starts from easy and progresses to more challenging data. Thus, gradual strategies enhance the model's transferability compared to not gradual strategy. In addition, we compare different ways of decreasing the k value, including piecewise, exponential and linear decreasing. We find that there may exist an optimal decreasing strategy for each task. Ultimately, `PaRe` adopt the simplest linear decreasing strategy as our default setup across all datasets.

4.3. How to select patches for replacement?

When considering patch selection from source embeddings to replace patches in target embeddings, a straightforward approach is to randomly choose patches for replacement. While random replacement indeed leads to the generation of intermediate modalities, as illustrated in Fig. 4, the right column represents patches selected through random patch selection, highlighted for visualization. From this column, we observe that due to the randomness of replacement, there is a possibility to select too many background patches. Additionally, the critical patches are not continuous, which might hinder the model from capturing essential information in the data and, instead, disrupt the model's training.

Therefore, we employ a gate network for patch scoring. Based on the scores, we select the top-k scored patches from the source to replace the bottom-k scored patches in the target. This approach allows us to preserve critical information from both source and target data as much as possible. The middle column in Fig. 4 represents patches selected through patch scoring. We observe that patches chosen through this strategy focus more on the main parts of the data, enabling more thorough training of the model on the intermediate modality.

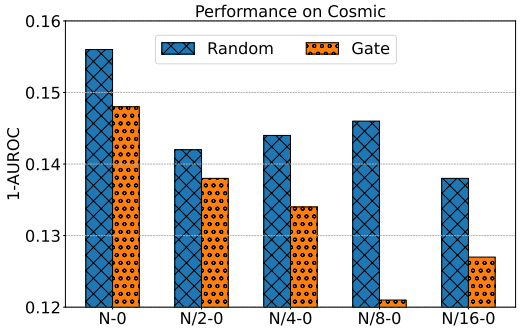


Figure 5: The impact on the results of random and gate strategy on Cosmic dataset with different initial k value. The smaller the initial k value, the larger performance percentage difference between random and gate strategy.

In Table 5, we present a comparison between the results of selecting patches randomly and using the gate network for patch selection. We observe that, across all datasets, the approach using the gate network consistently outperforms the random selection method. This demonstrates the effectiveness of the gate network in patch scoring. Furthermore, in Fig. 5, we compare the results on the Cosmic dataset when choosing different initial values for k and gradually decreasing it to 0 during the training process for both random and patch scoring methods. From the results, we observe that as the initial k value decreases (e.g., N-0 and N/8-0), the performance gap between the patch scoring method and the random method widens. This suggests that the advantage of our gate network becomes more pronounced when a smaller number of patches are selected.

To conduct a more comprehensive analysis, we configure the gate network with an fully connected (FC) layer followed by a sigmoid function, an MLP followed by a sigmoid function and an MLP followed by Dropout and a sigmoid function. In Table 6, we found that increasing the complexity of the gate network may not necessarily lead to improvement. This could be due to the limited amount of target modality data, and the increased complexity of the network may result in insufficient training. Therefore, we use the simplest gate network structure (FC layer followed by a sigmoid function) as the version of our method.

Table 6: Comparison prediction errors (↓) between different configuration of gate network.

	CIFAR-100	Spherical	Darcy Flow	PSICOV	Cosmic	NinaPro	FSD50K
FC + sigmoid (**Ours**)	6.25	**25.55**	7.00E-03	**0.99**	**0.121**	**6.53**	**0.55**
MLP + sigmoid	6.40	26.08	**6.90E-03**	**0.99**	0.125	6.74	**0.55**
MLP + dropout + sigmoid	**6.24**	26.52	**6.90E-03**	**0.99**	0.131	6.74	**0.55**

4.4. Influence of proxy source datasets

By deploying the patch replacement algorithm in the embedding space, PaRe facilitates the alignment of modalities, allowing for the utilization of a wide range of proxy source datasets. This promotes generalization across intermediate modalities, thus enhancing the model's performance. To demonstrate that the primary advancement of PaRe lies in its ability to gradually bridge the modality gap, rather than simply resulting from the substitution of different proxy source datasets. We conduct an ablation study in Table 7 using different proxy source datasets. Specifically, we utilize CIFAR-10, Caltech101 and Tiny-ImageNet as the proxy source datasets for the 2D classification task. As shown in Table 7, significant improvements of PaRe is over ORCA (Shen et al., 2023) across all scenarios, indicating that the efficacy of PaRe stems from its ability to bridge the modality gap rather than simply swapping proxy source datasets. Besides, PaRe achieves good results with multiple proxy source datasets with relatively robust performance.

Table 7: The influence of using CIFAR-10, Caltech101 and Tiny-ImageNet as the proxy source datasets for 2D classification task.

2D Classification		CIFAR-100	Spherical	NinaPro	FSD50K
Method	proxy source dataset	0-1error(%)	0-1error(%)	0-1error(%)	1-mAP
ORCA	CIFAR10	6.53	29.85	7.54	0.56
PaRe		**6.25**	**26.47**	**6.53**	**0.55**
ORCA	Caltech101	6.48	29.64	8.04	0.57
PaRe		**6.39**	**26.22**	**7.19**	**0.56**
ORCA	Tiny-ImageNet	6.44	28.21	8.35	0.56
PaRe		**6.25**	**25.55**	**7.59**	**0.55**

4.5. Visualization of source knowledge preservation

The goal of cross-modal fine-tuning is to adapt a pretrained model with rich knowledge in the source modality to a specific target modality. Intuitively, if the pretrained model can effectively retain the knowledge from the source modality during cross-modal fine-tuning, it can better leverage the abundant knowledge for transfer. Therefore, we obtain two models fine-tuned on Ninapro with PaRe and ORCA separately, and we compare the t-SNE visualization of the CIFAR10 features they output in Fig. 6.

As shown in Fig. 6, although ORCA narrows the OTDD between source and target modality data in the first stage, it still fails to preserve source modality knowledge due to significant modality gap. In contrast, PaRe utilizes intermediate modality data for training, requiring accurate feature extraction from source patches for correct classification. Therefore, knowledge from source modality can be well-preserved, leading to better transferability of the model.

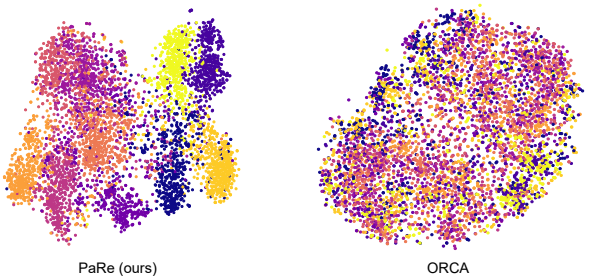


Figure 6: The t-SNE of the feature of PaRe and ORCA on source proxy dataset CIFAR10, which indicates the ability of source knowledge preservation.

5. Conclusion

In this paper, we propose an end-to-end cross-modal fine-tuning method, PaRe, employing patch scoring for patch replacement between the source and target modality data. PaRe facilitates the generation of intermediate modalities that progress from easy to hard during training, bridging the modality gap to enhance training stability and model tranferability, while also mitigating the challenge of limited data in the target modality. PaRe achieve good performance on three benchmarks consisting of 48 datasets, presenting a novel transfer methodology for cross-modal fine-tuning.

Limitation and furture work. During the course of this research, we identify some limitations in the current version and directions for future improvement: Firstly, since our method does not require training with computationally expensive OTDD (Alvarez-Melis & Fusi, 2020), our approach exhibits high scalability with respect to the source modality proxy dataset. However, determining the most suitable source modality proxy dataset based on the target modality dataset remains a challenge. Secondly, we observe that solely augmenting the target modality data can yield better results in certain modalities. However, these data augmentation methods are not universal. Therefore, a modality-agnostic data augmentation method is necessary to prevent model overfitting and enhance cross-modal fine-tuning. Lastly, leveraging unlabeled data from the target modality is also a promising direction for research, as it can better alleviate the issue of insufficient target modality data.

Acknowledgements

This paper was supported by the National Natural Science Foundation of China (No. 62376026), Beijing Nova Program (No. 20230484296) and CCF-Tencent Rhino-Bird Open Research Fund.

Impact Statement

The application of cross-modal transfer learning in this paper demonstrates significant performance improvements by transferring pretrained vision or language models to other modalities such as PDEs, protein structures, cosmic rays, gestures, and so on. This approach can drive advancements in diverse fields, promote interdisciplinary research, and enhance the robustness and versatility of models. By improving efficiency and accessibility, our work has the potential to foster novel discoveries, improve human-computer interaction, and contribute to societal and economic benefits.

References

Adhikari, B. DEEPCON: protein contact prediction using dilated convolutional neural networks with dropout. *Bioinformatics*, 36(2):470–477, 07 2019.

Alayrac, J., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J. L., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., and Simonyan, K. Flamingo: a visual language model for few-shot learning. pp. 23716–23736, 2022.

Alvarez-Melis, D. and Fusi, N. Geometric dataset distances via optimal transport. *NeurIPS*, 33:21428–21439, 2020.

Anil, R., Borgeaud, S., Wu, Y., Alayrac, J., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., Silver, D., Petrov, S., Johnson, M., Antonoglou, I., Schrittwieser, J., Glaese, A., Chen, J., Pitler, E., Lillicrap, T. P., Lazaridou, A., Firat, O., Molloy, J., Isard, M., Barham, P. R., Hennigan, T., Lee, B., Viola, F., Reynolds, M., Xu, Y., Doherty, R., Collins, E., Meyer, C., Rutherford, E., Moreira, E., Ayoub, K., Goel, M., Tucker, G., Piqueras, E., Krikun, M., Barr, I., Savinov, N., Danihelka, I., Roelofs, B., White, A., Andreassen, A., von Glehn, T., Yagati, L., Kazemi, M., Gonzalez, L., Khalman, M., Sygnowski, J., et al. Gemini: a family of highly capable multimodal models. *CoRR, abs/2312.11805*, 2023.

Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., Hui, B., Ji, L., Li, M., Lin, J., Lin, R., Liu, D., Liu, G., Lu, C., Lu, K., Ma, J., Men, R., Ren, X., Ren, X., Tan, C., Tan, S., Tu, J., Wang, P., Wang, S., Wang, W., Wu, S., Xu, B., Xu, J., Yang, A., Yang, H., Yang, J., Yang, S., Yao, Y., Yu, B., Yuan, H., Yuan, Z., Zhang, J., Zhang, X., Zhang, Y., Zhang, Z., Zhou, C., ingren Zhou, and Xiaohuan Zhou and, T. Z. Qwen technical report. *CoRR, abs/2309.16609*, 2023.

Bao, F., Nie, S., Xue, K., Li, C., Pu, S., Wang, Y., Yue, G., Cao, Y., Su, H., and Zhu, J. One transformer fits all distributions in multi-modal diffusion at scale. In *ICML*, pp. 1692–1717, 2023.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. Curriculum learning. In *ICML*, pp. 41–48, 2009.

Bertasius, G., Wang, H., and Torresani, L. Is space-time attention all you need for video understanding? In *ICML*, pp. 813–824, 2021.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In *NeurIPS*, 2020.

Chen, J.-N., Sun, S., He, J., Torr, P. H., Yuille, A., and Bai, S. Transmix: Attend to mix for vision transformers. In *CVPR*, pp. 12135–12144, 2022.

Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. *SIGKDD*, 2016.

Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., Li, B., Luo, P., Lu, T., Qiao, Y., and Dai, J. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *CoRR, abs/2312.14238*, 2023.

Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., and Xing, E. P. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023.

Cohen, T., Geiger, M., Köhler, J., and Welling, M. Spherical cnns. In *ICML*, 2018.

Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A. P., Caron, M., Geirhos, R., Alabdulmohsin, I., Jenatton, R., Beyer, L., Tschannen, M., Arnab, A., Wang, X., Ruiz, C. R., Minderer, M., Puigcerver, J., Evci, U., Kumar, M., van Steenkiste, S., Elsayed, G. F., Mahendran, A., Yu, F., Oliver, A., Huot, F., Bastings, J., Collier, M., Gritsenko, A. A., Birodkar, V., Vasconcelos, C. N., Tay, Y., Mensink, T., Kolesnikov, A., Pavetic, F., Tran, D., Kipf, T., Lucic, M., Zhai, X., Keysers, D., Harmsen, J. J., and Houlsby, N. Scaling

vision transformers to 22 billion parameters. In *ICML*, pp. 7480–7512, 2023.

Dempster, A., Petitjean, F., and Webb, G. I. ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Min. Knowl. Discov.*, 34(5):1454–1495, 2020.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR, abs/1810.04805*, 2018.

Dolan, E. D. and Moré, J. J. Benchmarking optimization software with performance profiles. *Math. Program.*, 91 (2):201–213, 2002.

Dong, R., Han, C., Peng, Y., Qi, Z., Ge, Z., Yang, J., Zhao, L., Sun, J., Zhou, H., Wei, H., Kong, X., Zhang, X., Ma, K., and Yi, L. Dreamllm: Synergistic multimodal comprehension and creation. In *ICLR*, 2024a.

Dong, X., Zhang, P., Zang, Y., Cao, Y., Wang, B., Ouyang, L., Wei, X., Zhang, S., Duan, H., Cao, M., et al. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *CoRR, abs/2401.16420*, 2024b.

Driess, D., Xia, F., Sajjadi, M. S. M., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., Huang, W., Chebotar, Y., Sermanet, P., Duckworth, D., Levine, S., Vanhoucke, V., Hausman, K., Toussaint, M., Greff, K., Zeng, A., Mordatch, I., and Florence, P. Palm-e: An embodied multimodal language model. In *ICML*, pp. 8469–8488, 2023.

Edwards, C., Zhai, C., and Ji, H. Text2mol: Cross-modal molecule retrieval with natural language queries. In *EMNLP*, pp. 595–607, 2021.

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., and Smola, A. Autogluon-tabular: Robust and accurate automl for structured data. *CoRR, abs/2003.06505*, 2020.

Everingham, M., Eslami, S. M. A., Gool, L. V., Williams, C. K. I., Winn, J. M., and Zisserman, A. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.*, 111(1):98–136, 2015.

Fonseca, E., Favory, X., Pons, J., Font, F., and Serra, X. Fsd50k: an open dataset of human-labeled sound events. *CoRR, abs/2010.00475*, 2021.

Goyal, S., Kumar, A., Garg, S., Kolter, Z., and Raghunathan, A. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *CVPR*, pp. 19338–19347, 2023.

Gu, X., Yang, Y., Zeng, W., Sun, J., and Xu, Z. Keypoint-guided optimal transport with applications in heterogeneous domain adaptation. In *NeurIPS*, pp. 14972–14985, 2022.

Hacohen, G. and Weinshall, D. On the power of curriculum learning in training deep networks. In *ICML*, pp. 2535–2544, 2019.

Han, J., Gong, K., Zhang, Y., Wang, J., Zhang, K., Lin, D., Qiao, Y., Gao, P., and Yue, X. Onellm: One framework to align all modalities with language. *CoRR, abs/2312.03700*, 2023.

Hollmann, N., Muller, S., Eggensperger, K., and Hutter, F. Tabpfn: A transformer that solves small tabular classification problems in a second. 2022.

Hong, S., Xu, Y., Khare, A., Priambada, S., Maher, K. O., Aljiffry, A., Sun, J., and Tumanov, A. Holmes: Health online model ensemble serving for deep learning models in intensive care units. *SIGKDD*, 2020.

Hong, W., Wang, W., Lv, Q., Xu, J., Yu, W., Ji, J., Wang, Y., Wang, Z., Zhang, Y., Li, J., Xu, B., Dong, Y., Ding, M., and Tang, J. Cogagent: A visual language model for gui agents. *CoRR, abs/2312.08914*, 2023.

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., de Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. Parameter-efficient transfer learning for NLP. In *ICML*, pp. 2790–2799, 2019.

Howard, J. and Ruder, S. Universal language model fine-tuning for text classification. In *ACL*, pp. 328–339, 2018.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

Huang, G., Liu, Z., and Weinberger, K. Q. Densely connected convolutional networks. *CVPR*, pp. 2261–2269, 2017.

Jaegle, A., Borgeaud, S., Alayrac, J.-B., Doersch, C., Ionescu, C., Ding, D., Koppula, S., Zoran, D., Brock, A., Shelhamer, E., Henaff, O. J., Botvinick, M., Zisserman, A., Vinyals, O., and Carreira, J. Perceiver IO: A general architecture for structured inputs & outputs. In *ICLR*, 2022.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *CoRR, abs/1611.01144*, 2016.

Jia, M., Tang, L., Chen, B.-C., Cardie, C., Belongie, S., Hariharan, B., and Lim, S.-N. Visual prompt tuning. In *ECCV*, pp. 709–727, 2022.

Josephs, D., Drake, C., Heroy, A., and Santerre, J. semg gesture recognition with a simple model of attention. *CoRR*, abs/2006.03645, 2020.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. In *NeurIPS*, 2017.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W., Dollár, P., and Girshick, R. B. Segment anything. In *ICCV*, pp. 3992–4003, 2023.

Klink, P., Yang, H., D'Eramo, C., Peters, J., and Pajarinen, J. Curriculum reinforcement learning via constrained optimal transport. In *ICML*, pp. 11341–11358, 2022.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Le, Y. and Yang, X. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., and Shan, Y. Seed-bench: Benchmarking multimodal llms with generative comprehension. *CoRR, abs/2307.16125*, 2023a.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. In *NeurIPS*, pp. 6391–6401, 2018.

Li, J., Li, D., Savarese, S., and Hoi, S. C. H. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pp. 19730–19742, 2023b.

Li, X., Wang, C., Tang, Y., Tran, C., Tang, Y., Pino, J., Baevski, A., Conneau, A., and Auli, M. Multilingual speech translation from efficient finetuning of pretrained models. In *ACL*, pp. 827–838, 2021a.

Li, Z., Kovachki, N. B., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. Fourier neural operator for parametric partial differential equations. In *ICLR*, 2021b.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., dos Santos Costa, A., Fazel-Zarandi, M., Sercu, T., Candido, S., and Rives, A. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *CoRR, abs/1907.11692*, 2019.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pp. 10012–10022, 2021.

Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., and Li, T. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022a.

Luo, R., Sun, L., Xia, Y., Qin, T., Zhang, S., Poon, H., and Liu, T. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings Bioinform.*, 23(6), 2022b.

Ma, W., Li, S., Cai, L., and Kang, J. Language semantic graph guided data-efficient learning. In *NeurIPS*, 2023.

Moor, M., Banerjee, O., Abad, Z. S. H., Krumholz, H. M., Leskovec, J., Topol, E. J., and Rajpurkar, P. Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956):259–265, 2023.

Narvekar, S., Peng, B., Leonetti, M., Sinapov, J., Taylor, M. E., and Stone, P. Curriculum learning for reinforcement learning domains: A framework and survey. *J. Mach. Learn. Res.*, 21:181:1–181:50, 2020.

Nguyen, E., Poli, M., Faizi, M., Thomas, A. W., Birch-Sykes, C., Wornow, M., Patel, A., Rabideau, C. M., Massaroli, S., Bengio, Y., Ermon, S., Baccus, S. A., and Ré, C. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. *CoRR, abs/2306.15794*, 2023.

OpenAI. GPT-4 technical report. 2023.

Ostroumova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. Catboost: unbiased boosting with categorical features. In *NeurIPS*, 2017.

Pan, S. J. and Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 36(10):1345–1359, 2010.

Pang, Z., Xie, Z., Man, Y., and Wang, Y.-X. Frozen transformers in language models are effective visual encoder layers. In *ICLR*, 2024.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. Moment matching for multi-source domain adaptation. In *ICCV*, pp. 1406–1415, 2019.

Peng, Z., Wang, W., Dong, L., Hao, Y., Huang, S., Ma, S., and Wei, F. Kosmos-2: Grounding multimodal large language models to the world. In *ICLR*, 2024.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In *ICML*, pp. 8748–8763, 2021.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. Robust speech recognition via large-scale weak supervision. In *ICML*, pp. 28492–28518, 2023.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.*, 378:686–707, 2019.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pp. 234–241, 2015.

Shen, J., Khodak, M., and Talwalkar, A. Efficient architecture search for diverse tasks. In *NeurIPS*, pp. 16151–16164, 2022a.

Shen, J., Li, L., Dery, L. M., Staten, C., Khodak, M., Neubig, G., and Talwalkar, A. Cross-modal fine-tuning: Align then refine. In *ICML*, pp. 31030–31056, 2023.

Shen, Z., Liu, Z., Liu, Z., Savvides, M., Darrell, T., and Xing, E. Un-mix: Rethinking image mixtures for unsupervised visual representation learning. In *AAAI*, pp. 2216–2224, 2022b.

Shukor, M., Dancette, C., Rame, A., and Cord, M. UnIVAL: Unified model for image, video, audio and language tasks. *Trans. Mach. Learn. Res.*, 2023.

Sun, Q., Yu, Q., Cui, Y., Zhang, F., Zhang, X., Wang, Y., Gao, H., Liu, J., Huang, T., and Wang, X. Generative pretraining in multimodality. In *ICLR*, 2024.

Tan, H. and Bansal, M. LXMERT: learning cross-modality encoder representations from transformers. In *EMNLP*, pp. 5099–5110, 2019.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Canton-Ferrer, C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. Llama 2: Open foundation and fine-tuned chat models. *CoRR, abs/2307.09288*, 2023.

Tu, R., Roberts, N., Khodak, M., Shen, J., Sala, F., and Talwalkar, A. Nas-bench-360: Benchmarking neural architecture search on diverse tasks. In *NeurIPS*, 2022.

Uddin, A. F. M. S., Monira, M. S., Shin, W., Chung, T., and Bae, S. Saliencymix: A saliency guided data augmentation strategy for better regularization. In *ICLR*, 2021.

Vanschoren, J., Van Rijn, J. N., Bischl, B., and Torgo, L. Openml: networked science in machine learning. *SIGKDD*, 15(2):49–60, 2014.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, pp. 5998–6008, 2017.

Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., and Bengio, Y. Manifold mixup: Better representations by interpolating hidden states. In *ICML*, pp. 6438–6447, 2019.

Wang, T., Jiang, W., Lu, Z., Zheng, F., Cheng, R., Yin, C., and Luo, P. Vlmixer: Unpaired vision-language pre-training via cross-modal cutmix. In *ICML*, pp. 22680–22690, 2022.

Wang, W., Bao, H., Dong, L., Bjorck, J., Peng, Z., Liu, Q., Aggarwal, K., Mohammed, O. K., Singhal, S., Som, S., and Wei, F. Image as a foreign language: Beit pretraining for vision and vision-language tasks. In *CVPR*, pp. 19175–19186, 2023a.

Wang, Y., Yue, Y., Lu, R., Liu, T., Zhong, Z., Song, S., and Huang, G. Efficienttrain: Exploring generalized curriculum learning for training visual backbones. In *ICCV*, pp. 5852–5864, 2023b.

Wei, H., Kong, L., Chen, J., Zhao, L., Ge, Z., Yang, J., Sun, J., Han, C., and Zhang, X. Vary: Scaling up the vision vocabulary for large vision-language models. *CoRR, abs/2312.06109*, 2023.

Weinshall, D., Cohen, G., and Amir, D. Curriculum learning by transfer learning: Theory and experiments with deep networks. In *ICML*, pp. 5238–5246, 2018.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. Huggingface's transformers: State-of-the-art natural language processing. *CoRR, abs/1910.03771*, 2019.

Yang, F., Feng, C., Chen, Z., Park, H., Wang, D., Dou, Y., Zeng, Z., Chen, X., Gangopadhyay, R., Owens, A., and Wong1, A. Binding touch to everything: Learning unified multimodal tactile representations. *CoRR, abs/2401.18084*, 2024.

Yang, H., Chen, H., Zhou, H., and Li, L. Enhancing cross-lingual transfer by manifold mixup. In *ICLR*, 2022.

Ye, Q., Xu, H., Xu, G., Ye, J., Yan, M., Zhou, Y., Wang, J., Hu, A., Shi, P., Shi, Y., Li, C., Xu, Y., Chen, H., Tian, J., Qi, Q., Zhang, J., and Huang, F. mplug-owl: Modularization empowers large language models with multimodality. *CoRR, abs/2304.14178*, 2023.

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, pp. 6023–6032, 2019.

Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., and Shinozaki, T. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In *NeurIPS*, pp. 18408–18419, 2021.

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

Zhang, K. and Bloom, J. S. deepcr: Cosmic ray rejection with deep learning. *J. Open Source Softw.*, 4(41):1651, 2019.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., et al. Opt: Open pre-trained transformer language models. *CoRR, abs/2205.01068*, 2022.

Zheng, B., Dong, L., Huang, S., Wang, W., Chi, Z., Singhal, S., Che, W., Liu, T., Song, X., and Wei, F. Consistency regularization for cross-lingual fine-tuning. In *ACL*, pp. 3403–3417, 2021.

Zhou, J. and Troyanskaya, O. G. Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods*, 12:931–934, 2015.

Zhou, K., Yang, Y., Qiao, Y., and Xiang, T. Domain adaptive ensemble learning. *IEEE Trans. Image Process.*, 30:8008–8018, 2021.

Zhou, T. and Bilmes, J. Minimax curriculum learning: Machine teaching with desirable difficulties and scheduled diversity. In *ICLR*, 2018.

Zhu, D., Chen, J., Shen, X., Li, X., and Elhoseiny, M. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *CoRR, abs/2304.10592*, 2023a.

Zhu, J., Bai, H., and Wang, L. Patch-mix transformer for unsupervised domain adaptation: A game perspective. In *CVPR*, pp. 3561–3571, 2023b.

A. Appendix

A.1. Algorithm of PaRe

The algorithm of PaRe is illustrated in Alg. 1. Firstly, we score the patches from both the source and target using a gate network to obtain scores for each patch. Next, we select the top-k scored patches to replace the bottom-k scored target patches one by one. To ensure the backpropagation of gradients, we employ Gumble softmax operation to select the top-k and bottom-k patches. Finally, we update the model by separately calculating the source and target loss.

Algorithm 1 Pseudocode of PaRe.

```
# N: the embeddings length
# x_t, x_s: target and source embeddings
# y_t, y_s: target and source labels
# model.gate: the gate netowrk of the model
# SubsetOperator: using gumble softmax to do selection

k = int(N - N * (current_epoch / totle_epoch))
for (x_t, y_t, x_s, y_s) in loader:
   # Patch Scoring using Gate Network
   tar_score = model.gate(x_t)
   src_score = model.gate(x_s)

   # select bottom-k and top-k
   bk_tar_mask = 1 - SubsetOperator(tar_score, N-k)
   tk_src_mask = SubsetOperator(src_score, k)
   bk_tar_indices = torch.nonzero(bk_tar_mask)
   tk_tar_indices = torch.nonzero(tk_tar_mask)

   # Patch Replacement
   x_mix = x_t.clone()
   x_mix = x_mix * (1 - bk_tar_mask)
   x_s = x_s * tk_src_mask
   x_mix[bk_tar_indices] = x_s[tk_src_indices]

   # Obtain Logits
   logits_mix_tar, logits_mix_src = model(x_mix)

   # Calculate Loss
   lam = k / N
   L_tar = L_tar(logits_mix_tar, y_t)
   L_src = L_src(logits_mix_src, y_s)
   L_mix = (1-lam) * L_tar + lam * L_src
```

A.2. Implementation details

A.2.1. Pretrained Models

We evaluate PaRe with two pretrained models in our experiments. For all 2D tasks we use Swin-base (Liu et al., 2021) pretrained on ImageNet-22K, and for all 1D tasks, we use RoBERTa-base (Liu et al., 2019) pretrained on Five English-language corpora. We follow ORCA (Shen et al., 2023) use the Hugging Face transformers library (Wolf et al., 2019) to implement the pretrained models.

A.2.2. NAS-Bench-360

NAS-Bench-360 encompasses 10 tasks across various modalities, such as image classification, hand gesture recognition, and solving PDEs, among others. These tasks have diverse objectives, including 2D classification, 2D dense prediction, and 1D classification. They involve different types of data and are trained using distinct loss functions. Additionally, each task is associated with its unique hand-designed expert model. The Table 8 provides an overview of NAS-Bench-360.

A.2.3. PDEBench

PDEBench comprises multiple datasets of partial differential equations (PDEs) with varying parameters and initial conditions. Following ORCA, we utilize the eight datasets listed in the Table 9, which provides detailed information on dataset parameters, utilized loss functions, and other relevant details.

Table 8: The introduction to the 10 tasks in NAS-Bench-360.

Dateset	DATA_NUM	DATA_DIM	TYPE	CLASS_NUM	Loss	Expert arch.
CIFAR100	60K	2D	Point	100	CE	DenseNet-BC (Huang et al., 2017)
Spherical	60K	2D	Point	100	CE	S2CN (Cohen et al., 2018)
NinaPro	3,956	2D	Point	18	LpLoss	Attention Model (Josephs et al., 2020)
FSD50K	51K	2D	Point	200	MSELoss	VGG (Fonseca et al., 2021)
Darcy Flow	1.1K	2D	Dense	1	BCE	FNO (Li et al., 2021b)
PSICOV	3,606	2D	Dense	1	FocalLoss	DEEPCON (Adhikari, 2019)
Cosmic	5,250	2D	Dense	1	BCE	deepCR-mask (Zhang & Bloom, 2019)
ECG	330K	1D	Point	4	CE	ResNet-1D (Hong et al., 2020)
Satellite	1M	1D	Point	24	CE	ROCKET (Dempster et al., 2020)
DeepSEA	250K	1D	Point	36	BCE	DeepSEA (Zhou & Troyanskaya, 2015)

Table 9: The introduction to the 8 tasks in PDEBench.

	Advection	Burgers	Diffusion-Reaction	Diffusion-Sorption	Navier-Stokes	Darcy-Flow	Shallow-Water	Diffusion-Reaction
DATA_DIM	1D	1D	1D	1D	1D	2D	2D	2D
TYPE	Dense Prediction							
Resolution	1024	1024	1024	1024	1024	128×128	128×128	128×128
Parameters	$\beta = 0.4$	$\nu = 1.0$	$\nu = 0.5, \rho = 1.0$	-	$\eta = \zeta = 0.1$	$\beta = 0.1$	-	-
Loss	Normalized Root Mean Squared Errors (nRMSEs)							

A.2.4. OpenML-CC18 Benchmark

We follow ORCA to evaluate `PaRe` on 30 datasets as shown in Table 10 from OpenML-CC18 benchmark, and we follow TabPFN to use the same evaluation protocol and use the one-vs-one AUROC as the score metric. The train-test split ratio is 0.5:0.5 to account for the limited context length of TabPFN. As for training, we employ the cross-entropy loss, with the class weights set to $1/(num\_of\_samples)$.

Table 10: The introduction of the 30 datasets in OpenML-CC18 benchmark.

OpenML ID	Name	#Class.	OpenML ID	Name	#Class.
11	balance-scale	3	1049	pc4	2
14	mfeat-fourier	10	1050	pc3	2
15	breast-w	2	1063	kc2	2
16	mfeat-karhunen	10	1068	pc1	2
18	mfeat-morphological	10	1462	banknote-authenti...	2
22	mfeat-zernike	10	1464	blood-transfusion-...	2
23	cmc	3	1480	ilpd	2
29	credit-approval	2	1494	qsar-biodeg	2
31	credit-g	2	1510	wdbc	2
37	diabetes	2	6332	cylinder-bands	2
50	tic-tac-toe	2	23381	dresses-sales	2
54	vehicle	4	40966	MiceProtein	8
188	eucalyptus	5	40975	car	4
458	analcatdata auth...	4	40982	steel-plates-fault	7
469	analcatdata dmft	6	40994	climate-model-simu...	2

A.2.5. Proxy Source Datasets

Although we don't need to follow the same approach as ORCA, which involves reducing the distance between the target and source modality at the embedding level in the first stage, we still require a proxy dataset for the source modality as we construct intermediate modalities through source and target data during end-to-end cross-modal fine-tuning. It's worth

mentioning that, since we don't need to calculate the loss using the computationally expensive OTDD, our source proxy dataset is more scalable compared to ORCA, with a relatively minor increase in computational complexity when using additional samples. Therefore, we use CIFAR10 and Tiny-imagenet as proxy datasets for 2D classification tasks, reset the labels of PASCAL VOC to 0 and 1, effectively treating it as a foreground-background segmentation task for 2D dense prediction tasks, and employ CONLL-2003 as the proxy dataset for both 1D classification and dense prediction tasks.

Our experiments in Tabel 11 revealed that choosing different source proxy datasets may have varied effects on different modalities. Therefore, determining how to select appropriate source proxy datasets based on different modalities is also a direction for future research in our method.

Table 11: Varied effects on different modalities while choosing different source proxy datasets.

Proxy Dataset	num_sample	Spherical	Darcy Flow	Ninapro
CIFAR10	5000	26.69	7.70E-03	7.13
	all	26.47	7.90E-03	**6.53**
Tiny_ImageNet	5000	27.18	-	7.44
	all	**25.55**	-	7.59
PASCAL_VOC	all	-	**7.00E-03**	-

A.2.6. HYPERPARAMETERS

Due to the multitude of tasks across different modalities, it's challenging to define a single set of fine-tuning hyperparameters for all models or tasks. Therefore, we adopt the exact same hyperparameters as ORCA (Shen et al., 2023) for model fine-tuning to facilitate comparison. The specific parameter settings are shown in the Tabel 12 and Table 13. For our method's hyperparameters, one concerns the initial and final values of k, while the other relates to the loss trade-off β_1 and β_2.

For the setting of k, we set the initial value to 3000 and the final value to 0 for all tasks except for the Cosmic dataset. Since Cosmic involves a binary classification dense prediction task, excessively large k values may overly interfere with the model. Therefore, for Cosmic, we set the initial value to 200 and the final value to 0. For all tasks, we uniformly set both β_1 and β_2 to 1.0.

Table 12: The training hyperparameters of the 10 tasks on NAS-Bench-360.

	CIFAR100	Spherical	NinaPro	FSD50K	Darcy Flow	PSICOV	Cosmic	ECG	Satellite	DeepSEA
Batch Size	32	32	32	32	4	1	4	4	16	16
Epoch	60	60	60	100	100	10	60	15	60	13
Accum.	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

Table 13: The training hyperparameters of the 8 tasks on PDEBench.

	Advection	Burgers	Diffusion-Reaction	Diffusion-Sorption	Navier-Stokes	Darcy-Flow	Shallow-Water	Diffusion-Reaction
Batch Size	4	4	4	4	4	4	4	4
Epoch	200	200	200	200	200	100	200	200
Accum.	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

A.3. Detailed results on OpenML-CC18 benchmark

The detailed results of PaRe and other compared methods are shown in Table 14, the average accuracy of PaRe across 30 datasets is the highest among all methods, with the highest results observed in 15 out of the 30 datasets. Compared to ORCA, the cross-modal fine-tuning approach like us, PaRe is better on 22 datasets.

Table 14: One-vs-one AUROC (↑) on 30 OpenML-CC18 datasets. "Diff. from XGBoost" is the acrosstask average of per-task difference from XGBoost. On 15/30 datasets, PaRe ranks the first among all compared methods.

	LightGBM	CatBoost	XGBoost	AutoGluon	TabPFN	ORCA	PaRe
balance-scale	0.9938	0.9245	0.9939	0.9919	**0.9973**	0.9949	0.9964
mfeat-fourier	0.9786	0.9816	0.9803	**0.9843**	0.9811	0.9729	0.9783
breast-w	0.991	0.9931	0.9896	0.9933	0.9934	**0.9939**	0.9909
mfeat-karhunen	0.9979	0.9986	0.9983	0.9987	0.9978	0.9968	**0.9988**
mfeat-morphologica..	0.9601	0.9629	0.9612	**0.9698**	0.9669	0.9647	0.9680
mfeat-zernike	0.9716	0.9759	0.9735	**0.9908**	0.9823	0.9829	0.9849
cmc	0.7288	0.7256	0.7299	0.7331	0.7276	0.7237	**0.7770**
credit-approval	0.9415	0.9389	0.9422	0.9415	0.9322	0.9340	**0.9601**
credit-g	0.7684	0.7852	0.7853	0.7941	0.7894	0.7748	**0.8200**
diabetes	0.8247	0.8383	0.8378	0.8391	0.8410	0.8239	**0.8570**
tic-tac-toe	0.9988	0.9992	**1**	**1**	0.9759	0.9973	0.9952
vehicle	0.9232	0.9302	0.9282	0.9416	0.9589	**0.9591**	0.9582
eucalyptus	0.8931	0.8979	0.9004	0.9204	0.9245	0.9084	**0.9510**
analcatdata_author..	0.9999	0.9999	0.9997	0.9993	**1**	0.9996	**1**
analcatdata_dmft	0.5461	0.5589	0.5743	0.5657	**0.579**	0.5627	0.5509
pc4	0.9301	0.9413	0.9291	**0.9428**	0.9383	0.9226	0.9301
pc3	0.8178	0.8247	0.8288	0.8282	0.8373	0.8411	**0.8493**
kc2	0.8141	0.8323	0.8227	0.8242	0.8346	**0.8431**	0.8398
pc1	0.8321	0.86	0.8489	0.8578	0.8761	0.8767	**0.9266**
banknote-authentic..	**1**	**1**	**1**	**1**	**1**	**1**	**1**
blood-transfusion-..	0.7144	0.7403	0.7312	0.7364	0.7549	**0.7565**	0.6287
ilpd	0.6917	0.7279	0.7171	0.723	0.7379	0.7419	**0.7927**
qsar-biodeg	0.9126	0.9217	0.9191	0.9276	0.9336	**0.9349**	0.9167
wdbc	0.9904	0.9931	0.9904	0.9956	**0.9964**	0.9929	0.9947
cylinder-bands	0.8556	0.8757	0.8782	0.8878	0.8336	0.844	**0.9243**
dresses-sales	0.5593	0.5696	0.5823	0.5507	0.5376	**0.6025**	0.5747
MiceProtein	0.9997	0.9999	0.9998	**1**	0.9999	0.9969	0.9997
car	0.9925	0.9955	0.9948	0.998	0.995	0.9983	**1**
steel-plates-fault..	0.9626	0.9655	0.9656	0.9666	0.9655	0.9543	**0.9696**
climate-model-simu..	0.9286	0.9344	0.9255	0.9391	0.9415	0.9416	**0.9551**
# Wins	1	1	2	7	5	7	**15**
Avg. AUROC	0.8840	0.8898	0.8909	0.8947	0.8943	0.8946	**0.9030**
Avg. Diff. from XGBoost	-0.0069	-0.0011	0	+0.0038	+0.0034	+0.0036	**+0.0121**

A.4. Visualization of patch selection

We visualize the patches selected by random strategy and our gate strategy in Fig. 7. The right column in Fig. 7 highlights patches selected randomly, visualized in red, we notice certain drawbacks. The random selection may result in an overabundance of background patches and a lack of continuity in the critical patches, potentially impeding the model's ability to capture essential data information and disrupting its training process. To address this issue, we introduce a gate network for patch scoring. As depicted in Fig. 7, the middle column illustrates patches selected through patch scoring. Notably, patches chosen through this method demonstrate a heightened focus on the primary components of the data, facilitating more comprehensive training of the model on the intermediate modality.

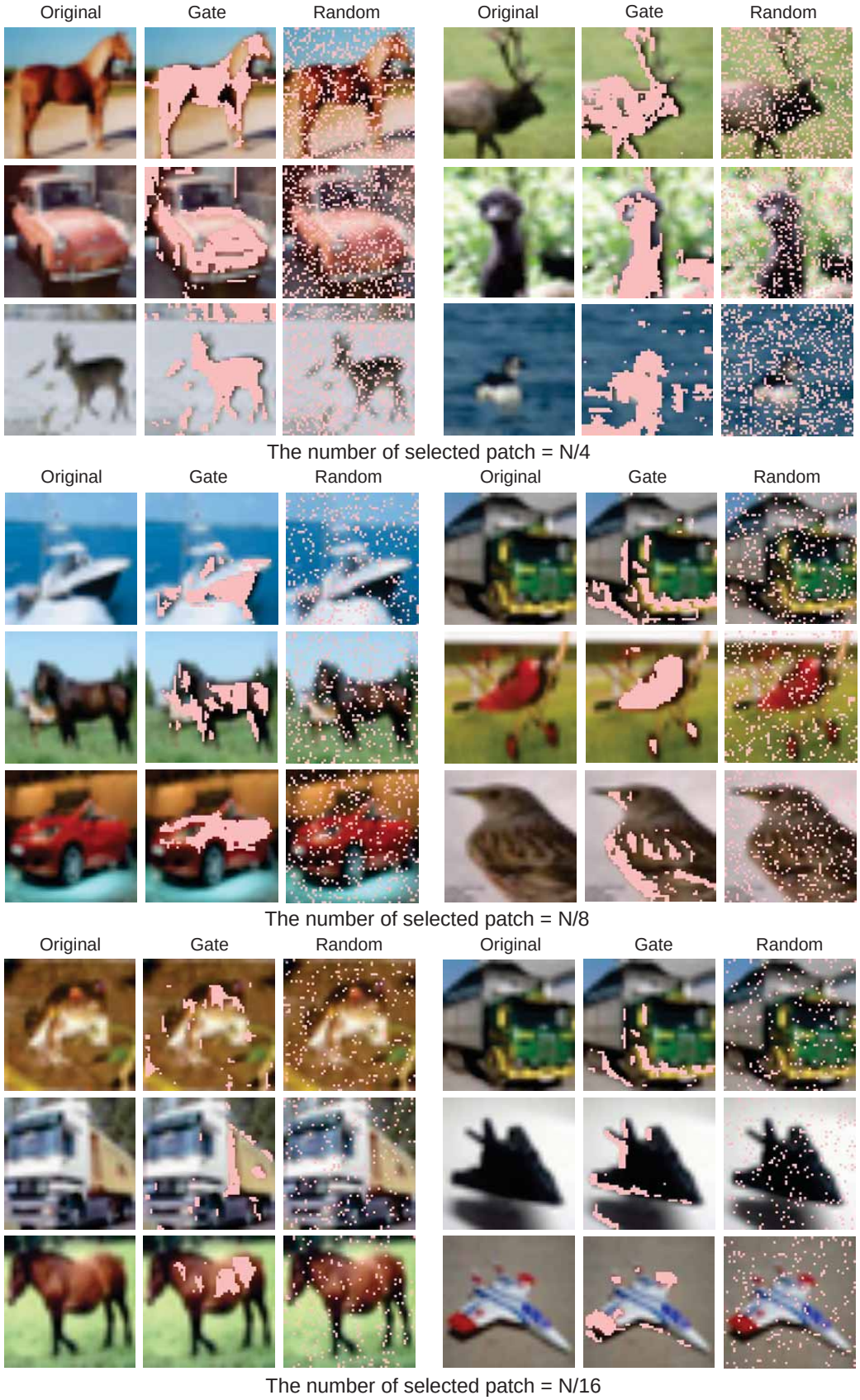


Figure 7: The visualization of the different numbers of patches selected by random strategy and our gate strategy.

Figure 8: The visualization of the different numbers of patches selected by random strategy and our gate strategy.

Furthermore, we illustrate the evolution of the gate network's predictions as it learns to integrate information from both the source modality (CIFAR10) and target modality (CIFAR100) in Fig. 8. As depicted in the visualization, we observe that initially, the gate network tends to produce random predictions. However, as training progresses, typically around the 10th epoch, we notice a significant transition as the gate network approaches convergence. At this stage, the gate network demonstrates a clear ability to select critical patches of the source modality data and the unimportant patches of the target modality data. After patch replacement, we can effectively maximize the retention of information from both modalities.

A.5. Additional evaluation on NAS-Bench-360

A.5.1. Comparison with different cross-modal fine-tuning method

In Table 15, we compare our method, `PaRe`, with various cross-modal fine-tuning approaches, including Train_from_scratch, FPT (layernorm), NFT (naive full fine-tuning), and ORCA. The results indicate that our method outperforms all other cross-modal fine-tuning methods across all tasks.

Table 15: Comparison with different cross-modal fine-tuning method including: Train_from_scratch: training SwinTransformer/RoBERTa from scratch; FPT: fine-tuning only the layernorm; NFT: fine-tuning all parameters, ORCA and `PaRe`.

	CIFAR-100	Spherical	Darcy Flow	PSICOV	Cosmic	NinaPro	FSD50K	ECG	Satellite	DeepSEA
Train_from_scratch	50.87	76.67	8.00E-02	5.09	0.5	9.96	0.75	0.42	12.38	0.39
FPT	10.11	76.38	2.10E-02	4.66	0.233	15.69	0.67	0.5	20.83	0.37
NFT	7.67	55.26	7.34E-03	1.92	0.17	8.35	0.63	0.44	13.86	0.51
ORCA	6.53	29.85	7.28E-03	1.91	0.152	7.54	0.56	**0.28**	11.59	0.29
`PaRe`	**6.25**	**25.55**	**7.00E-03**	**0.99**	**0.121**	**6.53**	**0.55**	**0.28**	**11.18**	**0.28**

A.5.2. In-modality transfer

In the preceding sections, we have thoroughly demonstrated the superiority of our method in the context of cross-modal fine-tuning. Next, we will validate the effectiveness of our approach in in-modality transfer in Table 16. Taking the miniDomainNet (Zhou et al., 2021) (a reduced version of DomainNet (Peng et al., 2019)) dataset with 126 classes as an example, we will assess the effectiveness of our method for in-modality transfer on four significantly different domains (Clipart, Painting, Real and Sketch)

Table 16: Prediction errors (↓) on four domains on miniDomainNet.

	Clipart	Painting	Real	Sketch
ORCA	10.16	12.86	**5.08**	14.29
`PaRe`	**9.21**	**11.75**	**5.08**	**14.13**

A.5.3. More data-limited scenarios

What performance differences arise when our method is applied to target modalities with scarcer data? To investigate this question, we further reduce the data in the target modality and compare the results with ORCA on three datasets: 2D classification Ninapro and 1D classification DeepSEA. We compare the results in Table 17 using 10%, 30%, 50%, 70%, and 90% of the training data. We found that even with less training data, our method still achieves better performance compared to ORCA.

A.5.4. Comparison with Data-augmentation

Data augmentation is an essential approach to enrich dataset diversity and prevent model overfitting. However, due to the diverse nature of target modalities, it's challenging to apply uniform data augmentation techniques across all modalities. For instance, common image augmentation techniques like random crop or grayscale may not be intuitively applicable to other modalities such as PDEs or cosmic rays and might even mislead the model.

Therefore, in Table 18, we further explore techniques like mixing up target data or utilizing masking operations for data augmentation. From the results, we observe that this augmentation approach is effective for datasets closer to the source

Table 17: The results of more limited data in target modality.

Ninapro	10%	30%	50%	70%	90%
ORCA	27.31	16.39	12.75	10.02	9.41
PaRe	**18.82**	**15.17**	**12.14**	**8.04**	**7.59**
DeepSEA	10%	30%	50%	70%	90%
ORCA	0.369	0.355	0.349	0.345	0.339
PaRe	**0.363**	**0.353**	**0.348**	**0.327**	**0.286**

modality, such as CIFAR100 and Spherical, yielding promising results. However, for tasks with significant differences from the source modality, like Darcy Flow or Ninapro, traditional techniques are ineffective and might even have negative impacts. We also find that if we only classify the mix embedding into the target label without incorporating the source loss, it essentially resembles the mask operation and cannot achieve modality-agnosticism. Hence, we ultimately adopt this intermediate modality generation approach for cross-modal transfer. However, designing a modality-agnostic data augmentation approach remains a future research direction.

Table 18: Comparison with in-modality data augmentation strategy.

	CIFAR100	Spherical	Darcy Flow	Ninapro
target_mixup	6.30	**24.12**	8.80E-03	9.71
target_mask	6.48	27.37	7.40E-03	7.89
PaRe w/o src_loss	6.54	26.65	7.40E-03	8.04
PaRe	**6.25**	25.55	**7.00E-03**	**6.53**