Transforming Vision Transformer: Towards Efficient Multi-Task Asynchronous Learning

Hanwen Zhong[1,2] **Jiaxin Chen**[1,2*] **Yutong Zhang**[1,2] **Di Huang**[2] **Yunhong Wang**[1,2]
[1]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China
[2]School of Computer Science and Engineering, Beihang University, Beijing, China
{hanwenzhong,jiaxinchen,ytzhang_mq,dhuang,yhwang}@buaa.edu.cn

Abstract

Multi-Task Learning (MTL) for Vision Transformer aims at enhancing the model capability by tackling multiple tasks simultaneously. Most recent works have predominantly focused on designing Mixture-of-Experts (MoE) structures and integrating Low-Rank Adaptation (LoRA) to efficiently perform multi-task learning. However, their rigid combination hampers both the optimization of MoE and the effectiveness of reparameterization of LoRA, leading to sub-optimal performance and low inference speed. In this work, we propose a novel approach dubbed Efficient Multi-Task Learning (EMTAL) by transforming a pre-trained Vision Transformer into an efficient multi-task learner during training, and reparameterizing the learned structure for efficient inference. Specifically, we firstly develop the MoEfied LoRA structure, which decomposes the pre-trained Transformer into a low-rank MoE structure and employ LoRA to fine-tune the parameters. Subsequently, we take into account the intrinsic asynchronous nature of multi-task learning and devise a learning Quality Retaining (QR) optimization mechanism, by leveraging the historical high-quality class logits to prevent a well-trained task from performance degradation. Finally, we design a router fading strategy to integrate the learned parameters into the original Transformer, archiving efficient inference. Extensive experiments on public benchmarks demonstrate the superiority of our method, compared to the state-of-the-art multi-task learning approaches. The project page is available at https://github.com/Yewen1486/EMTAL.

1 Introduction

Multi-task learning (MTL) [1, 2, 3] for Vision Transformer (ViT) aims at simultaneously learning multiple tasks, which has gained popularity in the computer vision community recently. It solves multiple relevant problems through sharing feature representations and forming a unified multi-task learner, thus enhancing the training and inference efficiency, and reducing the storage overhead. Moreover, by virtue of a well-designed MTL framework, the performance of each task can be further improved. Due to these merits, MTL has been used in a wide range of applications, such as the scene understanding [4, 5] and the erudite fine-grained recognition [6].

Despite both theoretical [7] and practical [8, 9] validations of their potential on enhancing the model generalizability, conventional MTL approaches [6, 10] often suffer performance degradation when compared to training tasks independently, which is primarily due to two reasons. Firstly, they adopt suboptimal learning strategies leading to conflicting task gradients and varying loss scales [11], which increase competitive interference during task optimization. Secondly, their model structures often fail to extract representative features for each task when utilizing a shared backbone network [12, 4, 6, 5].

*Corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-2579

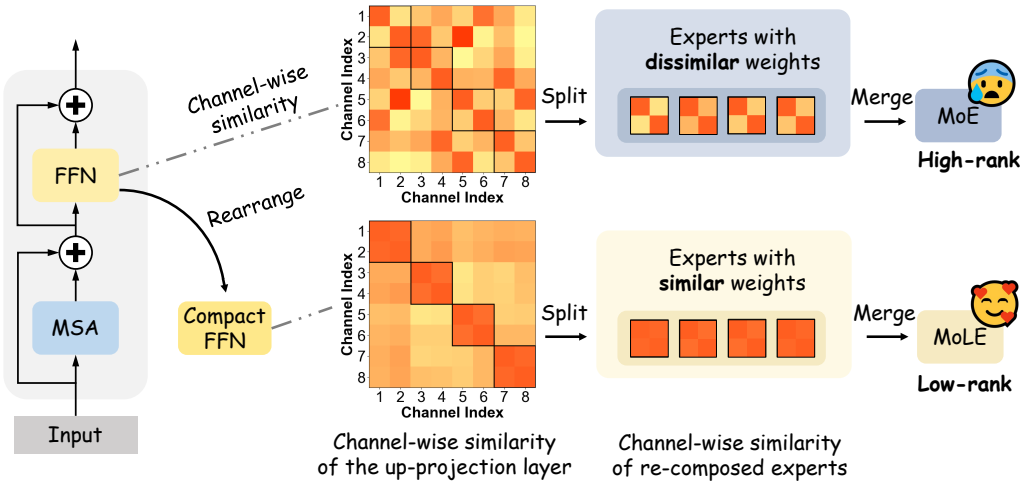
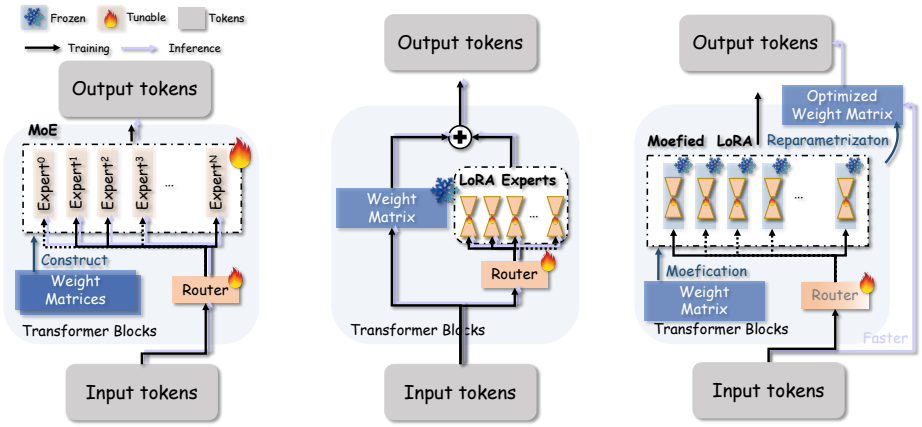


Figure 1: *FFN as Mixture of Low-rank Experts*. Given an up-projection weight matrix in FFN, a straightforward way of splitting it into MoE is to divide every K channels into separate experts, resulting in highly dissimilar experts and a high-rank MoE, which is inherently unsuitable for integration with LoRA. In contrast, our proposed MoLE approach rearranges the weight matrix into groups of similar channels as experts, creating specialized low-rank experts that are better suited for integrating with LoRA.

Several recent works have attempted to deal with the above issues, which concentrate on the following two aspects. 1) ***Efficient multi-task learners.*** As shown in Figure 2, instead of designing complex but incompact network structures [12, 4] that incur a large number of tunable parameters, recent works [13, 5, 14] such as MLoRE and MOELoRA explore the advantages of Mixture-of-Experts (MoE) in extracting task-specific features by enhancing the diversity of parameters and features [15, 16, 17], and Parameter-Efficient Fine-Tuning (PEFT) in reducing the tunable parameters and storage overhead [18, 19, 20, 21]. Nevertheless, MLoRE [5] still relies on a substantial number of additional parameters, limiting the overall efficiency and feasibility of training. MOELoRA [14] adopts a unitary LoRA structure to tune the experts, which weakens the learning capability of individual experts. Moreover, both methods utilizes task-driven routers, requiring either a static network with a fixed number of tasks or a dynamic routing network. The former results in significant storage overhead, while the latter increases the inference cost. 2) ***Multi-Task Optimization (MTO) strategies***. Existing works on MTO can be broadly categorized into the gradient-based methods and the loss-based methods. The gradient-based methods [22, 2, 23, 3, 24, 25] seek to balance the gradients across multiple tasks in the last shared layer, by decreasing differences in their magnitude or direction, and aggregating sub-gradients into a unified one. The loss-based methods [26, 27, 28] optimize the MTL process by balancing the multi-task losses. Recently, IMTL [11] is proposed to treat all tasks equally without bias, while AMTL [29] synchronizes learning progress across tasks. However, as each task has its own intrinsic optimization pace due to varying levels of training difficulty for distinct tasks, forcing synchronization in MTO disrupts these inherent properties, thus leading to suboptimal solutions. For more detailed discussion of related works, we refer to Appendix A.

To overcome the drawbacks of existing works, we propose a novel MTL framework dubbed Efficient Multi-Task Asynchronous Learning (EMTAL). Basically, EMTAL consists of the MoEfied LoRA structure, the Quality Retaining (QR) optimization mechanism and the router fading strategy, of which MoEfied LoRA decomposes a pre-trained Vision Transformer model into an efficient multi-task learner, QR accomplishes asynchronous learning of multi-task knowledge and enable establishing an efficient unified model by combining with the router fading strategy. Specifically, inspired by MoEfication [30, 31], the proposed MoEfied LoRA firstly decomposes the FFN layer of Vision Transformer into a MoE structure by clustering similar channels into experts, creating specialized low-rank experts as demonstrated in Figure 1. Considering the inherent low-rank property of each expert, LoRA is naturally employed to perform efficient training of MoE. Subsequently, in order to

achieve asynchronous learning of multi-task knowledge based on the MoEfied LoRA module, QR constrains logits of early converged tasks to retain near the optima when continuously optimizing the insufficiently converged tasks, thus avoiding severe interference between tasks and improving multi-task optimization. Finally, the router fading strategy is combined with MoEfied LoRA and QR by gradually diminishing the router's fole in the last training epochs, seamlessly integrating learned parameters into the model structure without incurring extra inference time cost and storage overhead.



(a) MoE [12, 4]. The conventional MoE leverages multiple expert networks and a gating mechanism to dynamically select the most relevant experts for each input. It focuses on designing complex but incompact network structures, incurring a large number of tunable parameters.

(b) LoRA Experts [5, 14]. LoRA Experts employ unified low-rank adaptation modules to achieve the parameter efficiency, which however limit the expert capacity, and requires either static networks with substantial storage overhead or dynamic routers with high inference cost.

(c) MoEfied LoRA (Ours). Our method groups similar weights into specialized low-rank experts, enabling seamless integration with LoRA to create an efficient multi-task learner. Besides, by combing with a router fading strategy, our method ensures both training and inference efficiency while substantially reduces storage overhead.

Figure 2: Summary of representative architectures of multi-task learning.

The main contributions of this paper are summarized as follows. 1) We propose a novel efficient multi-task learning framework dubbed EMTAL. To the best of our knowledge, our work makes the first investigation on decomposing a pre-trained Vision Transformer model for multi-task learning and reparameterizing the learned multi-task knowledge into a unified model. 2) We design a MoEfied LoRA structure, a QR multi-task optimization mechanism combined with a router fading strategy to accomplish an efficient asynchronous multi-task learner. 3) We extensively evaluate the proposed method on challenging multi-task fine-grained visual classification datasets and the VTAB benchmark, and the experimental results demonstrate that our method significantly improves the performance of single-task learning and the state-of-the-art MTL approaches.

2 The Proposed Approach

In this section, we mainly introduce the preliminary concepts of Vision Transformer, and describe the technical details of the proposed EMTAL approach.

2.1 Preliminary

Given an input image $I \in \mathbb{R}^{3\times H\times W}$, a standard Vision Transformer [32] model with L layers first divides I into m non-overlapping patches, which are further fed into a patch embedding layer, generating m D-dimensional visual tokens. After concatenating with a class token, the input tokens are finally formed as $\boldsymbol{X^0} \in \mathbb{R}^{(1+m)\times D}$. Each transformer layer contains a Multi-headed Self-Attention (MSA) [33] block, a Feed-Forward Networks (FFN) block and a Layer Normalziation (LN). The tokens of the l-th layer are generated based on those in the $(l-1)$-th layer formulated as below:

$$\boldsymbol{X}^{l'} = \mathrm{MSA}(\mathrm{LN}(\boldsymbol{X}^{l-1})) + \boldsymbol{X}^{l-1}, \quad \boldsymbol{X}^{l} = \mathrm{FFN}(\mathrm{LN}(\boldsymbol{X}^{l'})) + \boldsymbol{X}^{l'}. \tag{1}$$

https://doi.org/10.52202/079017-2579

Similar to [30, 31], our work mainly focuses on FFN, which usually consists of two linear layers $\{\boldsymbol{W}_{up} \in \mathbb{R}^{D\times(r\cdot D)}, \boldsymbol{b}_{up} \in \mathbb{R}^{r\cdot D}\}$, $\{\boldsymbol{W}_{down} \in \mathbb{R}^{(r\cdot D)\times D}, \boldsymbol{b}_{down} \in \mathbb{R}^{D}\}$ and a GELU activation operation, where r represents the scaling factor. Accordingly, FNN processes the normalized input $\boldsymbol{X}_n^{l\,\prime} = \mathrm{LN}(\boldsymbol{X}^{l\prime})$ as follows:

$$\boldsymbol{X}_{\mathrm{FFN}}^{l} = \mathrm{GELU}(\boldsymbol{X}_n^{l\,\prime}\boldsymbol{W}_{up} + \boldsymbol{b}_{up})\boldsymbol{W}_{down} + \boldsymbol{b}_{down}. \tag{2}$$

This procedure ensures the effective transformation and projection of the input through the encoder layers, enabling the prediction of the class probability distribution y for downstream tasks.

2.2 Framework Overview

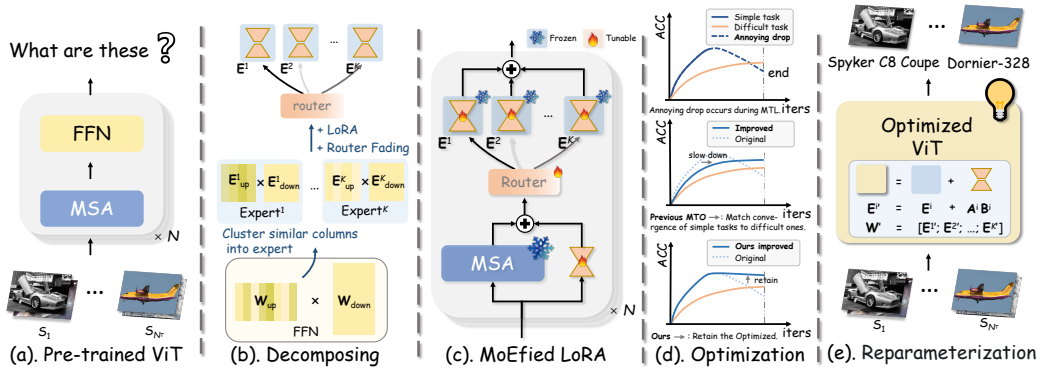


Figure 3: Illustration of the proposed EMTAL framework. Given a pre-trained ViT, we firstly decompose it into a MoE-based multi-task learner by using the balanced k-means. LoRA is then applied to the low-rank experts, creating an efficient multi-task learner dubbed MoEfied LoRA. During multi-task optimization, the Quality Retaining is employed to maintain the high-quality knowledge for tasks that have already converged. Finally, with the aid of the router fading strategy, the learned knowledge is reparameterized back into the pre-trained ViT, eliminating the extra inference cost.

As shown in Figure 3 (a), in order to establish a unified model for N_T tasks, we follow [6] by unifying the label spaces for multiple tasks into an overall one with N_{class} classes, and merging the training samples as $S = \bigcup_{t=1}^{N_T} S_t$, where S_t denotes the set of training samples for the i-th task. Supposing a pre-trained Vision Transformer with an embedding backbone $\Phi(\cdot;\theta_\phi) : \mathcal{X} \to \mathcal{F}$, where θ_ϕ represents the parameters to be frozen in the network, it maps an input $\boldsymbol{x} \in \mathcal{X}$ to the feature space $\mathcal{F}$. We decompose it into an efficient multi-task learner dubbed **MoEfied-LoRA** denoted by $\Phi'(\cdot;[\theta_\phi;\theta_t])$, where θ_t indicates the set of newly employed tunable parameters and $[;]$ refers to the concatenation operation. Subsequently, the **Quality Retaining** multi-task optimization mechanism as well as the router fading strategy are applied to asynchronously learning the tunable parameters θ_t. Finally, we reparameterize θ_t into the original backbone, achieving an efficient unified model $\Phi(\cdot;\theta'_\phi)$.

2.3 MoEfied-LoRA

Basically, MoE-based learners benefits MTL in the following two ways. First, they enable dynamic encoding of different samples across tasks via a router and multiple experts, significantly enhancing feature diversity. Second, they reduce the number of parameters and computational cost, remarkably promoting the training and inference efficiency. However, early attempts include delicately designed MMoE [12], M^{3}ViT [4] and Mod-Squad [13] fail to establish a unified structure and inevitably introduce additional inference overhead, considering that LoRA increases inference latency by 20-30% without reparameterization [34]. Recently, MoEfication methods [30, 31] aimed at "*group together the neurons that are often activated simultaneously*" have demonstrated promising performance in constructing effective MoE structures from pre-trained ViT models. Inspired by this, in this work we attempt to "*group together the neurons with similar weights*" to construct the MoE structure. As the corresponding experts naturally meet the low-rank conditions in LoRA [34], we can therefore obtain an excellent efficient multi-task learner by combining with LoRA. By employing a router fading strategy to preserve the reparameterization property of LoRA, we can further eliminate additional inference latency.

https://doi.org/10.52202/079017-2579

Specifically, as shown in Figure 3 (b), we fistly decompose the FFN into a MoLE (Mixture of Low-rank Experts) structure. To convert the FFN in the l-th layer into MoE, we draw on insights from [35, 36], which view the FFN as a memory bank that retains knowledge from pre-trained models. Each column of the up-projection matrix $\boldsymbol{W}_{up}$ serves as a key to be matched, which each row of the down-projection matrix $\boldsymbol{W}_{down}$ act as the corresponding value. Since similar keys in the $\boldsymbol{W}_{up}$ tend to serve similar functions, they should be grouped within the same expert. Formally, $\boldsymbol{W}_{up}$ is clustered column-wise into K clusters by using the balanced k-means [37], with a mapping function $C(\cdot)$ that assigns the idx-th column to cluster $C(idx)$. Therefore, columns within the same cluster constitute an expert. Additionally, $\boldsymbol{b}_{up}, \boldsymbol{W}_{down}$ should match $\boldsymbol{W}_{up}$ channel by channel, and also adhere to the clustering results of $\boldsymbol{W}_{up}$ to construct the experts $\{\boldsymbol{E}^i\}_{i=1}^K$. Consequently, they are concatenated for the MoEfication process and then split to obtain the corresponding experts, formulated as below:

$$\boldsymbol{W} = [\boldsymbol{W}_{up}; \boldsymbol{b}_{up}; \boldsymbol{W}_{down}^{\mathsf{T}}], \boldsymbol{W} \in \mathbb{R}^{(2D+1)\times(rD)}, \tag{3}$$

$$\boldsymbol{E}^i = \boldsymbol{W}_{\{idx|C(idx)=i\}}, i \in 1, 2, \cdots, K, \boldsymbol{E}^i \in \mathbb{R}^{(2D+1)\times(\frac{rD}{K})}, \tag{4}$$

$$\boldsymbol{E}^i_{up}, \boldsymbol{E}^i_b, \boldsymbol{E}^i_{down} = \boldsymbol{E}^i_{1:D}, \boldsymbol{E}^i_D, \boldsymbol{E}^i_{D+1:2D+1}. \tag{5}$$

Since the column vectors within these experts are relatively similar, exhibiting low-rank characteristics, they naturally satisfy the low-rank conditions in LoRA [34]. Therefore, we apply a set of LoRA parameters $\{\boldsymbol{A}^i_{up}, \boldsymbol{B}^i_{up}\}$ and $\{\boldsymbol{A}^i_{down}, \boldsymbol{B}^i_{down}\}$ for $\boldsymbol{E}^i_{up}$ and $\boldsymbol{E}^i_{down}$ to enable more efficient learning and improved performance, which is referred as MoLE and formally described as below:

$$\boldsymbol{E}^{i\ \prime}_{up} = \boldsymbol{E}^i_{up} + \boldsymbol{A}^i_{up}\boldsymbol{B}^i_{up}, \tag{6}$$

$$\boldsymbol{E}^{i\ \prime}_{down} = \boldsymbol{E}^i_{down} + \boldsymbol{A}^i_{down}\boldsymbol{B}^i_{down}. \tag{7}$$

Moreover, to leverage the benefits of dynamic routing for extracting diverse features during training, we initially establish a sample-driven soft router for each MoLE, denote as $\boldsymbol{W}_r \in \mathbb{R}^{D\times K}$, to reweight the experts. For the l-th layer of MoLE, we calculate weights for each experts as ω^l by adopting the following formulation:

$$\omega^l = K \cdot \mathrm{softmax}\left(\frac{\mathrm{LN}(\boldsymbol{X}^{l\prime})\,\boldsymbol{W}^l_r}{\tau}\right), \tag{8}$$

During training, the router is employed to fully optimize the MoLE. The output $\boldsymbol{X}^l_{\text{FFN-de}}$ of the decomposed FFN is calculated as follows:

$$\boldsymbol{X}^l_{\text{FFN-de}} = \sum_{i=1}^{K} \mathrm{GELU}([\omega^l_1 \cdot (\boldsymbol{X}^{l\ \prime}_n \boldsymbol{E}^{1\ \prime}_{up} + \boldsymbol{E}^1_b); \cdots; \omega^l_K \cdot (\boldsymbol{X}^{l\ \prime}_n \boldsymbol{E}^{K\prime}_{up} + \boldsymbol{E}^K_b)])\boldsymbol{E}^{i\ \prime}_{down} + \boldsymbol{b}_{down}. \tag{9}$$

2.4 Quality Retaining Optimization

The goal of multi-task learning (MTL) is to ensure that the final model performs well across all tasks. However, due to distinct task difficulties and individual optimization schedules, achieving optimal performance on all tasks simultaneously with a single model can be challenging. As displayed in Figure 3 (d), despite introducing strong priors to synchronize the optimization schedules across tasks, existing methods [26, 29, 25] can disrupt the inherent schedules of tasks with varying difficulties, creating challenges for MTL optimization. Therefore, we propose a different perspective: maintaining the inherent optimization pace of each task is crucial. Specifically, we allow asynchronous convergence of tasks and introduce the Quality Retaining (QR) MTO strategy to preserve high-quality knowledge from already converged tasks during subsequent optimization.

Specifically, at iteration $iter$, we maintain an optimal knowledge bank $\boldsymbol{Z} \in \mathbb{R}^{N_{class}\times N_{class}}$, which records the Exponential Moving Average (EMA) logits of each class learned during the optimization process from iteration 0 to $iter-1$. This knowledge is distilled into the currently optimized model using a distillation loss. Formally, we maintain the knowledge bank $\boldsymbol{Z}$ using EMA. For a sample s in the current training batch with label $label_s$, we update $\boldsymbol{Z}^{iter}_{label_s}$ as follows:

$$\boldsymbol{Z}^{iter}_{label_s} = m \cdot \boldsymbol{Z}^{iter-1}_{label_s} + (1-m) \cdot \boldsymbol{z}_s, \tag{10}$$

https://doi.org/10.52202/079017-2579

where $\boldsymbol{z}_s$ indicates the logits of sample s and $m \in (0, 1)$ is a momentum coefficient. This results in a real-time updated knowledge repository $\boldsymbol{Z}$.

To ensure the retention of high-quality knowledge for already optimized tasks, we employ a straightforward method, *i.e.*, weighting the distillation process by the reciprocal of the loss from each task. This procedure implies that tasks with lower loss (already optimized) should rely more on the learned knowledge, while tasks with higher loss (still being optimized) will depend more on the ground truth. Therefore, the Quality Retaining loss L_{QR} for samples within a mini-batch S_{batch} is defined as below:

$$\mathcal{L}_{QR} = \sum_{t=1}^{N_T} \frac{1}{\mathcal{L}_{CE,t}} \cdot \sum_{s \in S_{batch}} \mathrm{KL}\left(\mathrm{softmax}(\boldsymbol{z}_s), \mathrm{softmax}(\boldsymbol{Z}_{label_s})\right) \cdot \mathbb{1}(s \in S_t), \tag{11}$$

where $\mathrm{KL}(\cdot)$ and $\mathbb{1}$ indicates the Kullback-Leibler divergence [38] and the indicator function, respectively. $\mathcal{L}_{CE,t}$ is the Cross-Entropy loss for the t-th task.

Based on Eq. (11), the overall training loss is formulated as $\mathcal{L} = \sum_{t=1}^{N_T} \mathcal{L}_{CE,t} + \mathcal{L}_{QR}$. This strategy ensures that the model maintains optimal performance for already converged tasks while allowing other tasks to continue their optimization at their inherent pace.

2.5 Router Fading and Insights on the Unified Model Structure

MTL aims to develop a universal model capable of executing multiple tasks simultaneously. To achieve this goal, we transform the unified pre-trained model into an MoEfied LoRA and develop the quality retaining mechanism to preserve multi-task knowledge as discussed in the previous sections. However, the dynamic routing in MoEfied LoRA limits LoRA's capability of reparameterizing the learned parameters into a unified, static pre-trained structure. To address this issue, we design a router fading strategy that gradually diminishes the router's role in the later stages of training. This approach allows the knowledge embedded within the optimized router to be implicitly absorbed as follows:

$$\omega^l = \alpha * \omega^l + (1 - \alpha), \tag{12}$$

where $\alpha \in [0, 1]$ is the trade-off hyper-parameter. Finally, as shown in Figure 3 (e), we completely remove the router after training by setting $\alpha = 0$. In the mean time, we reparametere knowledge learned by LoRA back using Eq. (6) and Eq. (7), and concatenate them to replace the original parameters $\boldsymbol{W}_{up}, \boldsymbol{b}_{up}, \boldsymbol{W}_{down}$ of the l-th layer with $\boldsymbol{W}_{up}{}', \boldsymbol{b}'_{up}, \boldsymbol{W}_{down}{}'$ as below:

$$\boldsymbol{W}_{up}{}' = [\boldsymbol{E}_{up}^{1}{}'; \boldsymbol{E}_{up}^{2}{}'; \cdots; \boldsymbol{E}_{up}^{K}{}'], \tag{13}$$

$$\boldsymbol{b}_{up}{}' = [\boldsymbol{E}_{b}^{1\prime}; \boldsymbol{E}_{b}^{2\prime}; \cdots; \boldsymbol{E}_{b}^{K\prime}], \tag{14}$$

$$\boldsymbol{W}_{down}{}' = [\boldsymbol{E}_{down}^{1}{}'; \boldsymbol{E}_{down}^{2}{}'; \cdots; \boldsymbol{E}_{down}^{K}{}']. \tag{15}$$

The above technique avoids extra computational costs and maintains a unified model structure, delivering a new perspective for multi-task learning.

3 Experimental Results and Analysis

3.1 Datasets and Evaluation Metric

By following [6], we mainly evaluate the performance of our proposed EMTAL method on the challenging *Multi-task FGVC* benchmark. In addition, we conduct experiments on the *Specialized VTAB-1k* dataset to validate the effectiveness over previous solutions. *Multi-task FGVC* is a collection of public datasets specifically for multi-task fine-grained visual classification, including CUB-200-2011 [39], Stanford Cars [40], FGVC-Aircraft [41] and Oxford Flowers [42]. In order to make fair comparisons, we adopt the standard training/testing split as depicted in [6]. *Specialized VTAB-1k* [43] consists of specialist images from specialized equipment, where we employ multi-task learning to fully leverage these expensively annotated data. We follow the standard training/validation splits used in [43] for fair comparisons. The top-1 accuracy is utilized as the evaluation metric. To further demonstrate the effectiveness of our method on the tasks of pixel-to-pixel dense prediction, we also conduct experiments on the NYUv2 dataset [44]. Specifically, we integrate our method

Table 1: Comparison of the top-1 accuracy (%) on the Multi-task FGVC benchmark, by using ViT-B/16 supervised pre-trained on ImageNet-21K. 'FT' denotes 'Full Fine-tuning'. The best results are highlighted in **bold** and the second best ones are underlined.

Method	Reference	Unified Model	CUB-200 -2011	Stanford Cars	FGVC-Aircraft	Oxford Flowers	Mean	Tunable Params. (M)	Inference Time (ms)
MTL baselines									
Separate FT	Baseline	✗	86.4	87.6	77.2	98.8	87.49	343.92	14.30
Union FT [6]	Baseline	✓	83.1	90.7	78.3	97.5	87.39	85.98	**7.15**
MTO Gradient-based									
Nash-MTL [24]	ICLR' 22	✓	88.3	90.2	80.6	99.5	89.65	2.82	**7.15**
Aligned-MTL [25]	CVPR' 23	✓	88.9	90.6	81.6	**99.7**	90.17	2.82	**7.15**
MTO Loss-based									
GLS [26]	CVPR' 19	✓	88.4	90.1	80.0	99.6	89.55	2.82	**7.15**
AMTL [29]	ICCV' 23	✓	88.2	90.7	81.5	**99.7**	90.04	2.82	**7.15**
QR	**Ours**	✓	88.1	**92.3**	85.0	99.6	91.25	2.82	**7.15**
Efficient multi-task learners									
Dual-Prompt [49]	ECCV' 22	✗	87.8	73.5	53.1	99.4	78.4	1.1	15.48
Erudite [6]	CVPR' 23	✗	79.7	81.4	70.2	98.1	82.35	101.34	9.74
MLoRE [5]	CVPR' 24	✗	74.8	59.5	49.9	99.2	70.76	188.01	42.02
MOELoRA [14]	SIGIR' 24	✗	88.4	88.2	75,0	**99.7**	88.04	2.82	38.71
MoEfied LoRA	**Ours**	✓	88.5	91.3	81.5	**99.7**	90.27	1.20	**7.15**
EMTAL-1	**Ours**	✓	**90.5**	91.9	81.8	**99.7**	90.96	**0.75**	**7.15**
EMTAL-2	**Ours**	✓	90.0	92.2	83.5	**99.7**	91.35	0.90	**7.15**
EMTAL-4	**Ours**	✓	89.8	**92.3**	**85.2**	**99.7**	**91.73**	1.20	**7.15**

with TaskPrompter [45] by applying MoEfied LoRA and QR to the FFN layers and the semantic segmentation task head, respectively.

In addition, we evaluate on *Multi-task FGVC* for few-shot learning under 1, 2, 4, 8, and 16 shots, by following existing works [46, 47].

3.2 Implementation Details

We utilize ViT-B/16 [2] pre-trained on ImageNet-21K [32] as the base model. We use the AdamW optimizer [48] to fine-tune our models for 100 epochs and adopt the cosine learning rate decay with a linear warm-up for 10 epochs in all experiments. We fix the hyper-parameters τ in Eq. (8) to 5, since it exhibits stable performance with distinct values. As for data augmentation, we employ random resize cropping to 224 × 224 pixels and a random horizontal flip during training and resize to 248 × 248 pixels with a center crop to 224 × 224 pixels. All experiments are conducted on a single Nvidia GeForce RTX 3090 GPU.

3.3 Comparison with the State-of-the-Art Approaches

To comprehensively evaluate the performance of our approach, we compare with the MTL full fine-tuning baseline and the following categories of state-of-the-art approaches: 1) MTO Loss-based methods, including GLS [26] and AMTL [29]; 2) MTO Gradient-based methods, including Nash-MTL [24] and Aligned-MTL [25] combined with vanilla LoRA-16; 3) Efficient multi-task learners methods, including Dual-Prompt [49], Erudite [6], MLoRE [5] and MOELoRA [14]. As the performance of EMTAL depends on the inherent low-rank properties, we report the results of our method using distinct ranks including 1, 2 and 4, denoted by EMTAL-1, EMTAL-2 and EMTAL-4, respectively.

As summarized in Table 1, the proposed EMTAL method consistently improves the performance at different ranks, promoting the top-1 accuracy of the Separate full-finetuning baseline by an average of 3.47%, 3.86% and 4.24%, respectively. More importantly, EMTAL tunes only a negligible amount of parameters (*i.e.*, 1.20M) compared to the original model (*i.e.*, 343.92M), and incurs no extra inference

[2] ViT-B/16 supervised pre-trained on ImageNet-21K.

https://doi.org/10.52202/079017-2579

Table 2: Comparison results (%) with the state-of-the-art PEFT and MTL methods on Specialized VTAB-1k by using ViT-B/16 models supervised pre-trained on ImageNet-21K. 'FT' denotes 'Full Fine-tuning'. The best results are highlighted in **bold** and the second best is underlined.

Method	Reference	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	Mean	Tunable Params. (M)
MTL baselines							
Separate FT	Baseline	79.7	95.7	84.2	73.9	83.38	343.36
Union FT [6]	Baseline	84.3	93.9	83.0	75.2	84.08	85.99
Traditional PEFT							
Adapter [50]	ICML' 19	76.3	88.0	73.1	70.5	76.98	1.08
LoRA [34]	ICLR' 22	85.5	95.3	86.1	75.3	85.50	2.41
VPT-D [18]	ECCV' 22	81.8	96.1	83.4	68.4	82.42	2.40
SSF [19]	NeurIPS' 22	**87.4**	95.9	87.4	75.5	86.56	0.96
SPT-L [51]	ICCV' 23	85.7	**96.2**	85.9	75.9	85.92	2.16
ARC [20]	NeurIPS' 23	84.9	95.7	86.7	75.8	85.78	0.52
PEFT for MTL							
AMTL [25]	ICCV' 23	86.4	95.0	85.8	75.9	85.79	2.41
MOELoRA [14]	SIGIR' 24	86.9	96.1	88.4	76.7	87.01	2.41
EMTAL-1	**Ours**	85.5	**96.2**	88.0	77.5	86.78	**0.34**
EMTAL-2	**Ours**	87.3	95.7	88.1	78.7	87.43	0.49
EMTAL-4	**Ours**	**87.4**	96.1	**89.1**	**78.9**	**87.89**	0.78

Table 3: More evaluation results on NYUv2 with ViT-B/16. The best results are highlighted in **bold**.

Method	Semseg mIoU ↑	Depth RMSE ↓	Normal mErr ↓	Boundary odsF ↑	Mean Δ (%) ↑
TaskPrompter-Base [45]	50.40	0.5402	**18.91**	**77.60**	-
+ EMTAL (Ours)	**52.90**	**0.5284**	18.95	77.10	**1.57**

cost, implying that our framework is significantly more effective and efficient than the traditional separate training paradigm.

Moreover, EMTAL consists of a reparameterizable and efficient multi-task learner, a Quality Retaining MTO mechanism and a router fading strategy. Compared to the state-of-the-art methods, each component of our approach shows significant advantages. In terms of the design of MTL structures, MoEfied LoRA seamlessly integrates low-rank experts with LoRA, resulting in improved performance. Furthermore, the router fading strategy and the reparameterization effectively reduce inference time, significantly enhancing the overall efficiency. Meanwhile, by considering the intrinsic task-specific optimization pace, the QR mechanism clearly improves previous MTO strategies. Overall, it utilizes a sample-driven router and multiple experts to extract diverse feature representations while preserving high-quality knowledge during training, making it highly beneficial for multi-task learning. In this way, our method achieves the highest accuracy compared to the state-of-the-art approaches, surpassing the second best Aligned-MTL by 1.56% while tuning fewer parameters.

We further evaluate the performance on *Specialized VTAB-1k*, where MTL is highly beneficial against previous solutions. As displayed in Table 2, some existing works focus on applying parameter-efficient fine-tuning on each task individually to avoid over-fitting. However, we find that performance can be significantly enhanced by incorporating multi-task learning. Specifically, applying AMTL and MOELoRA on the vanilla LoRA yields improvements of 0.29% and 1.51%, respectively. Furthermore, when directly applying our EMTAL to these tasks, we observe consistent improvements across different ranks, achieving the highest accuracy compared to the state-of-the-art approaches. Notably, it enhances the overall performance by 1.43%, while utilizing fewer parameters. In addition, on the NYUv2 dataset, our method significantly enhances performance in semantic segmentation and depth estimation tasks, while achieving comparable results in surface normal estimation and object boundary detection tasks. Overall, this led to an average relative improvement of 1.57%, validating the effectiveness of our approach for pixel-level prediction tasks. We provide more results, including using self-supervised pre-trained model DINOv2-large and the few-shot learning in Appendix B and Appendix C, respectively.

3.4 Ablation Study

In this section, we evaluate the effectiveness of the proposed main components, *i.e.* MoEfied LoRA and Quality Retaining by extensive ablation studies.

Table 4: Ablation results (%) of the main components on the Multi-task FGVC by using ViT-B/16 backbone. The best results are highlighted in **bold**.

MoEfied LoRA	Quality Retaining	CUB-200 -2011	Stanford Cars	FGVC-Aircraft	Oxford Flowers	Mean	Tunable Params. (M)
✗	✗	83.1	90.7	78.3	97.5	87.39	86.26
✓	✗	88.5	91.3	81.5	**99.7**	90.27	**1.20**
✗	✓	88.5	91.6	84.4	99.6	91.04	86.26
✓	✓	**89.8**	**92.3**	**85.2**	**99.7**	**91.73**	**1.20**

Table 5: Ablation results (%) of the MoEfied LoRA and router fading strategy on Multi-task FGVC by using ViT-B/16 backbone. The best results are highlighted in **bold**.

Method	CUB-200 -2011	Stanford Cars	FGVC-Aircraft	Oxford Flowers	Mean	Tunable Params. (M)	Inference Time (ms)
Union FT	83.1	90.7	78.3	97.5	87.39	86.26	**7.15**
+MoEfied	85.2	91.3	80.8	98.7	89.20	86.40	13.87
+LoRA [34]	88.2	91.0	**81.7**	99.6	90.14	**1.20**	13.87
+Router fading	**88.5**	**91.3**	81.5	**99.7**	**90.27**	**1.20**	**7.15**

Table 6: Ablation results (%) of the proposed MoEfied LoRA with different numbers of clusters (*i.e.* k) and distinct ways to construct experts.

Hyper.	Clusters #					Method		
	1	4	16	64	192	Co-activation	Gradient-cluster	**Ours**
Params. (M)	1.05	1.09	1.20	1.64	2.82	1.20	1.20	1.20
Mean Acc	88.83	89.12	**90.27**	89.34	89.02	89.30	89.34	**90.27**

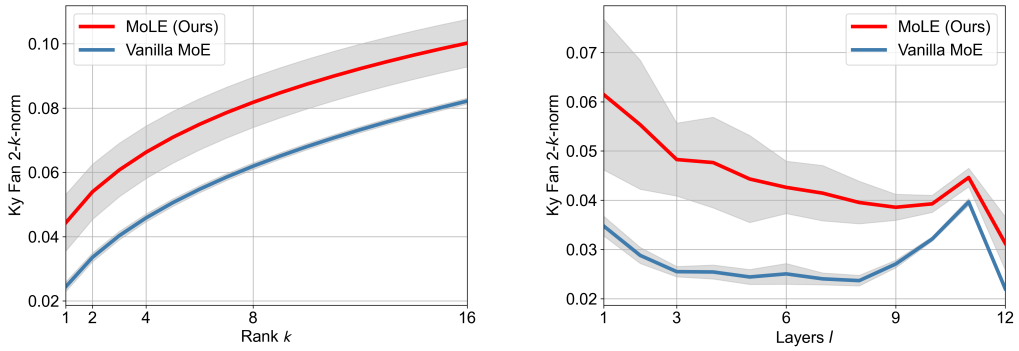
Effect of the Main Components. We evaluate the proposed components across the *Multi-task FGVC* Benchmark based on ViT-B/16. As Table 4 shows, MoEfied LoRA consistently boost the performance, achieving an average of 2.88% improvement with less tunable parameters. The results indicate that MoEfied LoRA is significantly beneficial to multi-task learning by providing more diverse features. In the mean time, the Quality Retaining MTO mechanism can further remarkably promote the accuracy, with a 3.65% improvement on average. A combination of these two components, *i.e.* MoEfied LoRA and QR, further boosts the overall performance across datasets, implying that MoEfied LoRA and Quality Retaining are complementary in multi-task learning.

On MoEfied LoRA and Router Fading. We further validate the effectiveness of MoEfied LoRA and the router fading strategy across the *Multi-task FGVC* benchmark based on ViT-B/16. Initially, we begin by decomposing the pre-trained model into a MOLE structure. A straightforward clustering and splitting of the FFN, combined with a sample-driven router, can achieve a 1.81% improvement by enhancing the the diversity of feature representations. Furthermore, applying LoRA to the low-rank experts yields a 0.94% gain in performance, as the small amount of tunable parameters reduces the risk of overfitting, and the the low-rank property of the experts aligns well with LoRA. Additionally, the proposed router fading strategy gradually diminishes the influence of the router over 50 epochs, effectively preserving the reparameterizable nature of LoRA and reducing the inference time.

Moreover, we conduct more ablation studies on the proposed MoEfied LoRA. The number of clusters k significantly influence the performance of MoEfied LoRA, considering that a large number of clusters intends to incur simple experts with few channels, and a small number of clusters results in high-rank experts, either of which degrades the effectiveness of MoEfied LoRA. We empirically study the effect of k by using 1, 4, 6, 64 and 192 clusters. As Table 6 displays, MoEfied LoRA reaches the highest accuracy when $k = 16$. Moreover, we compare different ways to construct experts, including the co-activation clustering [28] that groups weights based on activations for each channel and the gradient-cluster [47] that clusters weights according to cumulative gradients. As shown in Table 6, our method achieves the best performance, clearly demonstrating its effectiveness.

https://doi.org/10.52202/079017-2579

3.5 Visualization on the Low-rank Property of MoLE



(a) Low-rank properties of experts across different ranks in the 4-th transformer block.

(b) Low-rank properties of experts across different transformer block l with a fixed rank 1.

Figure 4: Comparison of the low-rank properties by using the vanilla MoE and the proposed MoLE, based on the Ky Fan 2-k norm [52]. A higher value signifies a stronger low-rank property.

As shown in Figure 4 (a), we measure the low-rank properties of experts by using the Ky Fan 2-k norm [52], and the results indicate that the experts generated by our method consistently exhibit statistically more significant low-rank properties across distinct ranks. Additionally, we analyze the low-rank properties of experts across different layers of ViT when the ranks is fixed as 1. As Figure 4 (b) demonstrates, the low-rank properties of experts are more significant in lower layers that those in higher layers.

We kindly refer to Appendix E for more detailed discussion about the broader impacts and limitations of our work .

4 Conclusion

In this paper, we focus on decomposing a pre-trained Vision Transformer model for multi-task learning, reparameterizing the learned multi-task knowledge back into the original model and establishing a unified model. We propose a novel efficient multi-task learning framework dubbed EMTAL, which mainly consists of the MoEfied LoRA module, the Quality Retaining (QR) mechanism and the router fading strategy. Concretely, MoEfied LoRA decomposes a pre-trained ViT into multi-task learners by clustering similar weight of FFN into experts and applies LoRA to tune the experts with low-rank properties. Subsequently, we leverage the inherent asynchronous convergence property of tasks and employ QR to preserve the optimized performance for converged tasks. Finally, the router fading strategy is introduced to eliminate extra inference cost. Extensive experiments on the public benchmarks demonstrate that our method substantially promotes performance by comparing with the state-of-the-art multi-task learning approaches, while also being effective in few-shot learning with limited data. Our work delivers a new perspective for efficient multi-task learning by decomposing a pre-trained model and reparameterizing back with a low rank updating.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China (Nos. 62202034, 62176012, 62022011), the Beijing Natural Science Foundation (No. 4242044), the Beijing Municipal Science and Technology Project (No. Z231100010323002), the Aeronautical Science Foundation of China (2023Z071051002), the Research Program of State Key Laboratory of Virtual Reality Technology and Systems, and the Fundamental Research Funds for the Central Universities.

References

[1] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, pages 525–536, 2018.

[2] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, pages 5824–5836, 2020.

[3] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems*, pages 18878–18890, 2021.

[4] Zhiwen Fan, Rishov Sarkar, Ziyu Jiang, Tianlong Chen, Kai Zou, Yu Cheng, Cong Hao, Zhangyang Wang, et al. M^3vit: Mixture-of-experts vision transformer for efficient multi-task learning with model-accelerator co-design. In *Advances in Neural Information Processing Systems*, volume 35, pages 28441–28457, 2022.

[5] Yuqi Yang, Peng-Tao Jiang, Qibin Hou, Hao Zhang, Jinwei Chen, and Bo Li. Multi-task dense prediction via mixture of low-rank experts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27927–27937, 2024.

[6] Dongliang Chang, Yujun Tong, Ruoyi Du, Timothy Hospedales, Yi-Zhe Song, and Zhanyu Ma. An erudite fine-grained visual classification model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7268–7277, 2023.

[7] Mickaël Binois, Victor Picheny, Patrick Taillandier, and Abderrahmane Habbal. The kalai-smorodinsky solution for many-objective bayesian optimization. *Journal of Machine Learning Research*, 21(150):1–42, 2020.

[8] Shikun Liu, Andrew Davison, and Edward Johns. Self-supervised generalisation with meta auxiliary learning. In *Advances in Neural Information Processing Systems*, 2019.

[9] Aviv Navon, Idan Achituve, Haggai Maron, Gal Chechik, and Ethan Fetaya. Auxiliary learning by implicit differentiation. In *International Conference on Learning Representations, ICLR 2021*, 2021.

[10] Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. In *International Conference on Learning Representations*.

[11] Liyang Liu, Yi Li, Zhanghui Kuang, Jing-Hao Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Towards impartial multi-task learning. In *International Conference on Learning Representations*, 2021.

[12] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1930–1939, 2018.

[13] Zitian Chen, Yikang Shen, Mingyu Ding, Zhenfang Chen, Hengshuang Zhao, Erik G Learned-Miller, and Chuang Gan. Mod-squad: Designing mixtures of experts as modular multi-task learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11828–11837, 2023.

[14] Qidong Liu, Xian Wu, Xiangyu Zhao, Yuanshao Zhu, Derong Xu, Feng Tian, and Yefeng Zheng. When moe meets llms: Parameter efficient fine-tuning for multi-task medical applications. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1104–1114, 2024.

[15] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

 https://doi.org/10.52202/079017-2579

[16] Robert A Jacobs and Michael I Jordan. Learning piecewise control strategies in a modular neural network architecture. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(2):337–345, 1993.

[17] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

[18] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge J. Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Proceedings of the European Conference on Computer Vision*, pages 709–727, 2022.

[19] Dongze Lian, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Scaling & shifting your features: A new baseline for efficient model tuning. In *Advances in Neural Information Processing Systems*, pages 109–123, 2022.

[20] Wei Dong, Dawei Yan, Zhijun Lin, and Peng Wang. Efficient adaptation of large vision transformer via adapter re-composing. In *Advances in Neural Information Processing Systems*, 2023.

[21] Bing Li, Jiaxin Chen, Xiuguo Bao, and Di Huang. Compressed video prompt tuning. *Advances in Neural Information Processing Systems*, 36:31895–31907, 2023.

[22] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning*, pages 794–803, 2018.

[23] Zirui Wang, Yulia Tsvetkov, Orhan Firat, and Yuan Cao. Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models. *arXiv preprint arXiv:2010.05874*, 2020.

[24] Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *International Conference on Machine Learning*, volume 162, pages 16428–16446, 2022.

[25] Dmitry Senushkin, Nikolay Patakin, Arseny Kuznetsov, and Anton Konushin. Independent component alignment for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20083–20093, 2023.

[26] Sumanth Chennupati, Ganesh Sistu, Senthil Yogamani, and Samir A Rawashdeh. Multinet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[27] Baijiong Lin, Feiyang Ye, Yu Zhang, and Ivor W. Tsang. Reasonable effectiveness of random weighting: A litmus test for multi-task learning. *Transactions on Machine Learning Research*, 2022, 2022.

[28] Shikun Liu, Edward Johns, and Andrew J Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1871–1880, 2019.

[29] Hayoung Yun and Hanjoo Cho. Achievement-based training progress balancing for multi-task learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16935–16944, 2023.

[30] Zhengyan Zhang, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Moefication: Transformer feed-forward layers are mixtures of experts. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 877–890, 2022.

[31] Mikołaj Piórczyński, Filip Szatkowski, Klaudia Bałazy, and Bartosz Wójcik. Exploiting transformer activation sparsity with dynamic inference. *arXiv preprint arXiv:2310.04361*, 2023.

[32] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[34] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[35] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, 2021.

[36] Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 8493–8502, 2022.

[37] Mikko I Malinen and Pasi Fränti. Balanced k-means for clustering. In *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshop*, pages 32–41, 2014.

[38] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[39] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[40] Timnit Gebru, Jonathan Krause, Yilun Wang, Duyun Chen, Jia Deng, and Li Fei-Fei. Fine-grained car detection for visual census estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4502–4508, 2017.

[41] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[42] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729, 2008.

[43] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

[44] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Proceedings of the European Conference on Computer Vision*, 2012.

[45] Hanrong Ye and Dan Xu. Taskprompter: Spatial-channel multi-task prompting for dense scene understanding. In *International Conference on Learning Representations*, 2023.

[46] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

[47] Yuanhan Zhang, Kaiyang Zhou, and Ziwei Liu. Neural prompt search. *arXiv preprint arXiv:2206.04673*, 2022.

[48] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

 https://doi.org/10.52202/079017-2579

[49] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *Proceedings of the European Conference on Computer Vision*, pages 631–648, 2022.

[50] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, pages 2790–2799, 2019.

[51] Haoyu He, Jianfei Cai, Jing Zhang, Dacheng Tao, and Bohan Zhuang. Sensitivity-aware visual parameter-efficient fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11825–11835, 2023.

[52] Xuan Vinh Doan and Stephen A. Vavasis. Low-rank matrix recovery with ky fan 2-k-norm. *J. Glob. Optim.*, 82(4):727–751, 2022.

[53] Rich Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

[54] Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020.

[55] Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[56] Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3614–3633, 2021.

[57] Yuan Gao, Jiayi Ma, Mingbo Zhao, Wei Liu, and Alan L Yuille. Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3205–3214, 2019.

[58] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3994–4003, 2016.

[59] Sebastian Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. Latent multi-task architecture learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4822–4829, 2019.

[60] David Bruggemann, Menelaos Kanakis, Stamatios Georgoulis, and Luc Van Gool. Automated search for resource-efficient branched multi-task networks. *arXiv preprint arXiv:2008.10292*, 2020.

[61] Pengsheng Guo, Chen-Yu Lee, and Daniel Ulbricht. Learning to branch for multi-task learning. In *International Conference on Machine Learning*, pages 3854–3863, 2020.

[62] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018.

[63] David Brüggemann, Menelaos Kanakis, Anton Obukhov, Stamatios Georgoulis, and Luc Van Gool. Exploring relational context for multi-task dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15869–15878, 2021.

[64] Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Mti-net: Multi-scale task interaction networks for multi-task learning. In *Proceedings of the European Conference on Computer Vision*, pages 527–543, 2020.

[65] Ling Zhou, Zhen Cui, Chunyan Xu, Zhenyu Zhang, Chaoqun Wang, Tong Zhang, and Jian Yang. Pattern-structure diffusion for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4514–4523, 2020.

[66] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4106–4115, 2019.

[67] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Zequn Jie, Xiang Li, and Jian Yang. Joint task-recursive learning for semantic segmentation and depth estimation. In *Proceedings of the European Conference on Computer Vision*, pages 235–251, 2018.

[68] Amir R Zamir, Alexander Sax, Nikhil Cheerla, Rohan Suri, Zhangjie Cao, Jitendra Malik, and Leonidas J Guibas. Robust learning through cross-task consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11197–11206, 2020.

[69] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022.

[70] Zhao Chen, Jiquan Ngiam, Yanping Huang, Thang Luong, Henrik Kretzschmar, Yuning Chai, and Dragomir Anguelov. Just pick a sign: Optimizing deep multitask models with gradient sign dropout. In *Advances in Neural Information Processing Systems*, volume 33, pages 2039–2050, 2020.

[71] Michelle Guo, Albert Haque, De-An Huang, Serena Yeung, and Li Fei-Fei. Dynamic task prioritization for multitask learning. In *Proceedings of the European conference on computer vision*, pages 270–287, 2018.

[72] Xiangyun Zhao, Haoxiang Li, Xiaohui Shen, Xiaodan Liang, and Ying Wu. A modulation module for multi-task learning with applications in image retrieval. In *Proceedings of the European Conference on Computer Vision*, pages 401–416, 2018.

[73] Siwei Yang, Hanrong Ye, and Dan Xu. Contrastive multi-task dense prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 3190–3197, 2023.

[74] Wei-Hong Li, Xialei Liu, and Hakan Bilen. Learning multiple dense prediction tasks from partially annotated data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18879–18889, 2022.

[75] Weiyi Lu, Sunny Rajagopalan, Priyanka Nigam, Jaspreet Singh, Xiaodi Sun, Yi Xu, Belinda Zeng, and Trishul Chilimbi. Asynchronous convergence in multi-task learning via knowledge distillation from converged tasks. In *NAACL*, pages 149–159, 2022.

[76] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

[77] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*, 2020.

[78] Simiao Zuo, Xiaodong Liu, Jian Jiao, Young Jin Kim, Hany Hassan, Ruofei Zhang, Tuo Zhao, and Jianfeng Gao. Taming sparsely activated transformer with stochastic experts. *arXiv preprint arXiv:2110.04260*, 2021.

[79] Hanrong Ye and Dan Xu. Taskexpert: Dynamically assembling multi-task representations with memorial mixture-of-experts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21828–21837, 2023.

[80] Shibo Jie and Zhi-Hong Deng. Fact: Factor-tuning for lightweight adaptation on vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1060–1068, 2023.

[81] Yen-Cheng Liu, Chih-Yao Ma, Junjiao Tian, Zijian He, and Zsolt Kira. Polyhistor: Parameter-efficient multi-task adaptation for dense vision tasks. volume 35, pages 36889–36901, 2022.

 https://doi.org/10.52202/079017-2579

[82] Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5227–5237, 2022.

[83] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv preprint arXiv:2012.13255*, 2020.

[84] Yerlan Idelbayev and Miguel A Carreira-Perpinán. Low-rank compression of neural nets: Learning the rank of each layer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8049–8059, 2020.

[85] Chi Su, Fan Yang, Shiliang Zhang, Qi Tian, Larry S Davis, and Wen Gao. Multi-task learning with low rank attribute embedding for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3739–3747, 2015.

[86] Madeleine Udell, Corinne Horn, Reza Zadeh, Stephen Boyd, et al. Generalized low rank models. *Foundations and Trends® in Machine Learning*, 9(1):1–118, 2016.

[87] Yongxin Yang and Timothy M Hospedales. Trace norm regularised deep multi-task learning. *arXiv preprint arXiv:1606.04038*, 2016.

[88] Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. Lorahub: Efficient cross-task generalization via dynamic lora composition. *arXiv preprint arXiv:2307.13269*, 2023.

[89] Yunhao Gou, Zhili Liu, Kai Chen, Lanqing Hong, Hang Xu, Aoxue Li, Dit-Yan Yeung, James T Kwok, and Yu Zhang. Mixture of cluster-conditional lora experts for vision-language instruction tuning. *arXiv preprint arXiv:2312.12379*, 2023.

[90] Shihan Dou, Enyu Zhou, Yan Liu, Songyang Gao, Jun Zhao, Wei Shen, Yuhao Zhou, Zhiheng Xi, Xiao Wang, Xiaoran Fan, et al. Loramoe: Revolutionizing mixture of experts for maintaining world knowledge in language model alignment. *arXiv preprint arXiv:2312.09979*, 2023.

[91] Zeren Chen, Ziqin Wang, Zhen Wang, Huayang Liu, Zhenfei Yin, Si Liu, Lu Sheng, Wanli Ouyang, and Jing Shao. Octavius: Mitigating task interference in mllms via lora-moe. In *International Conference on Learning Representations*, 2024.

[92] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024, 2024.

[93] Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 1–9, 2022.

[94] Arnav Chavan, Zhuang Liu, Deepak Gupta, Eric Xing, and Zhiqiang Shen. One-for-all: Generalized lora for parameter-efficient fine-tuning. *arXiv preprint arXiv:2306.07967*, 2023.

[95] Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. In *Advances in Neural Information Processing Systems*, pages 79320–79362, 2023.

[96] Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, et al. Parameter-efficient orthogonal finetuning via butterfly factorization. In *International Conference on Learning Representations*, 2024.

[97] Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2018.

[98] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019.

[99] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 2017.

[100] Kaggle and EyePacs. Kaggle diabetic retinopathy detection, July 2015.

[101] Jiaxi Wu, Jiaxin Chen, Mengzhe He, Yiru Wang, Bo Li, Bingqi Ma, Weihao Gan, Wei Wu, Yali Wang, and Di Huang. Target-relevant knowledge preservation for multi-source domain adaptive object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5301–5310, 2022.

[102] Wu Ke, Jiaxin Chen, and Miao Wang. Domain adaptive object detection for uav-based images by robust representation learning and multiple pseudo-label aggregation. In *Proceedings of the 1st International Workshop on Efficient Multimedia Computing under Limited*, pages 59–67, 2024.

[103] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664, 2021.

 https://doi.org/10.52202/079017-2579

A Related Work

A.1 Multi-Task Learning

Multi-Task Learning (MTL) [53] aims to improve the generalization performance of models across individual tasks by leveraging shared representations to exploit the commonalities and interdependencies between tasks. This approach also reduces the number of parameters and accelerates both training and inference [53, 54, 55]. Existing works are roughly divided into two categories: 1) the efficient multi-task learners and the multi-task optimization methods [56].

Efficient multi-task learners. These approaches concentrate on analyzing the impact of parameter sharing within models and can be broadly categorized into encoder-focused and decoder-focused approaches, depending on where information is exchanged or shared between tasks [56]. In encoder-focused models, task parameters are shared exclusively within the encoder to extract general features, by leveraging mechanisms such as feature fusion [57, 58, 59], attention [28, 6], and dynamic branching [60, 61], while the decoder consists of independent task-specific heads with no cross-task information sharing [22, 62, 1]. In decoder-focused models, parameters are shared across tasks within the decoder. The model initially makes separate predictions for each task and then refines these results by leveraging inter-task correlations through mechanisms such as multi-model distillation [63, 64, 65, 66], sequential task prediction [67], or cross-task consistency [68]. Some continual learning frameworks [69, 49] are also applicable to MTL, incorporating task-specific and task-shared parameters to achieve effective isolation and sharing between tasks.

Multi-Task Optimization Methods. The optimization-based methods [22, 70, 71, 62, 72, 25, 24] focus on balancing how tasks are learned, exploring effective solutions from the perspective of model optimization. These approaches enhance the optimization process of MTL through various design of multi-task losses [26, 68, 62, 73, 74, 29], which assigns appropriate loss weights to minimize conflicts among tasks. [75] proposes recording the optimal checkpoint for each task and learning from it by distilling the soft labels of individual samples. However, this approach may record a local optimal solution, leading to suboptimal results. Gradient manipulations [70, 22, 23, 2, 3] techniques address task interference by directly adjusting gradients, with recent methods emphasizing the formulation of a unified gradient vector subject to diverse constraints.

A.2 Mixture-of-Experts

Mixture-of-Experts (MoE) [16, 15] are originally designed to combine the decisions of a series of sub-models, *i.e.*, expert networks, on the same input, enhancing conditional computational capabilities and enabling the scaling of parameters in neural networks [76, 77, 17, 78]. Therefore, MoE trains multiple specialized expert networks, each of which maintains its own unique set of trainable parameters. This design allows the expert networks to develop distinct internal representations tailored to their respective input data. Additionally, MoE employs a router that dynamically weights the outputs of each expert network, enabling their contributions to be combined into the final output.

MoE is also applied in Multi-Task Learning (MTL), effectively partitioning the parameter space and leveraging relevant model components for different tasks, making it a promising solution for MTL [15, 16, 17]. M^3ViT [4] customizes the MoE layer within a ViT backbone, and activates task-specific experts during training to mitigate gradient conflicts in MTL. Mod-Squad [13] introduces a modular multi-task leaner based on MoE, along with a novel loss function, to address the gradient conflicts among tasks. MMoE [12] designs a multi-gate MoE to ensemble expert networks for various census analysis tasks, each with a different router. TaskExpert [79] generates multi-task predictions for all tasks in a single forward pass simultaneously, leading to significantly higher multi-task training efficiency. MLoRE [5] explicitly builds global relationships among all tasks within the MoE structure and introduces low-rank experts, improving the efficiency of MoE compared to the vanilla approach.

A.3 Low-Rank Updating

Low-Rank Adaption (LoRA) [34] is a parameter-efficient fine-tuning method [34, 80, 81, 82], inspired by the observation from [83] that the difference in weights between the pre-trained model and the adapted model lies in a low intrinsic rank. With the success of low-rank structure [84, 85, 86, 87] as a parameter-efficient fine-tuning technique, numerous studies have demonstrated impressive results by combining LoRA and MoE for more efficient and effective model tuning. LoRAHub [88] first

trains multiple LoRA weight modules on upstream tasks. To adapt the model to a downstream task, it employs a gradient-free optimization method to determine the optimal coefficients for linearly combining the pre-trained LoRA modules. MOELoRA [14] utilizes a router network conditioned on a task identifier to dynamically combine the outputs of multiple LoRA experts. Similarly, MoCLE [89] designs a router network that is conditioned on the clustering information extracted from each input sample. LoRAMoE [90] splits the LoRA experts into two groups and explicitly learns distinct capabilities for each group. While these mixture-of-LoRA methods densely combine multiple LoRA experts, a sparse mixture of LoRA experts offers a more economical alternative, achieving comparable performance while maintaining roughly constant training and inference costs. The Octavius [91] method, for instance, selects the top-2 LoRA experts based on a router that conditions on the entire input instance, representing a more coarse-grained routing mechanism. Besides, earlier MTL methods use the low-rank structure to model task-generic features and generate task-specific features through linear combinations. VL-Adapter[82] exploits adapter-based methods to efficiently fine-tune generative models in a multi-task setting, while [87] introduces a framework for training multiple neural networks simultaneously, sharing all shareable layers and learning the sharing strategy in a data-driven manner.

B More Experimental Results on DINOv2-large

Table 7: Comparison results (%) with the state-of-the-art PEFT approaches on the Specialized VTAB-1k benchmark by using the self-supervised pre-training ViT-L/14 model, *i.e.*, DINOv2-large [92]. 'FT' denotes 'Full Fine-tuning'. The best results are highlighted in **bold** and the second best ones are underlined.

Method	Reference	Unified Model	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	Mean	Tunable Params. (M)
Separate FT	Baseline	✗	88.1	96.1	90.9	77.2	88.08	1217.6
BitFit [93]	ACL' 22	✗	85.2	96.1	90.7	75.7	86.92	1.08
FacTtt [80]	AAAI' 23	✗	87.1	94.3	88.7	74.0	86.02	**0.48**
FacTtk [80]	AAAI' 23	✗	86.1	94.6	89.5	74.2	86.10	**0.48**
LoRA [34]	ICLR' 22	✗	88.3	96.4	91.4	77.4	88.38	7.08
GLoRA [94]	arXiv' 23	✗	85.9	96.0	91.0	76.2	87.27	19.48
OFT [95]	NeurIPS' 23	✗	88.4	96.4	91.5	77.2	88.38	8.40
BOFT [96]	ICLR' 24	✗	88.9	**96.6**	91.6	77.3	88.60	7.96
EMTAL-4	**Ours**	✓	**89.4**	95.9	**91.7**	**80.1**	**89.27**	2.03

In addition to the ViT-B/16 pre-trained model as displayed in Table 2, we evaluate the performance of our method based on the self-supervised pre-training ViT-L/14 model, *i.e.*, DINOv2-large [3]. As summarized in Table 7, our method consistently outperforms the compared approaches. Notably, compared with the separate full-finetuning (Separate FT) baseline which incurs 1217.6M tunable parameters (4×304.4M) and the other alternative methods, EMTAL successfully constructs a unified multi-task framework on this benchmark. Our approach eliminates the need to develop multiple specialized models while delivering superior performance, thereby validating its generalizability when applied to larger self-supervised models.

C More Experimental Results on Few-shot learning

We also evaluate the performance of our method on the challenging few-shot learning task. As illustrated in Figure 5, the following observations can be made regarding the average performance : (i) EMTAL, LoRA, Adapter, and VPT exhibit similar performance with extreme limited amount of training data, such as with only 1 or 2 shots. (ii) As the amount of training data increases, EMTAL demonstrates a significant advantage. For instance, with 16 shots, EMTAL promotes the accuracy of the second best one by approximately 5%. This highlights that multi-task training has a particularly significant advantage over the separate training, especially in scenarios where reliable annotation information is limited.

[3] DINOv2-large.

https://doi.org/10.52202/079017-2579

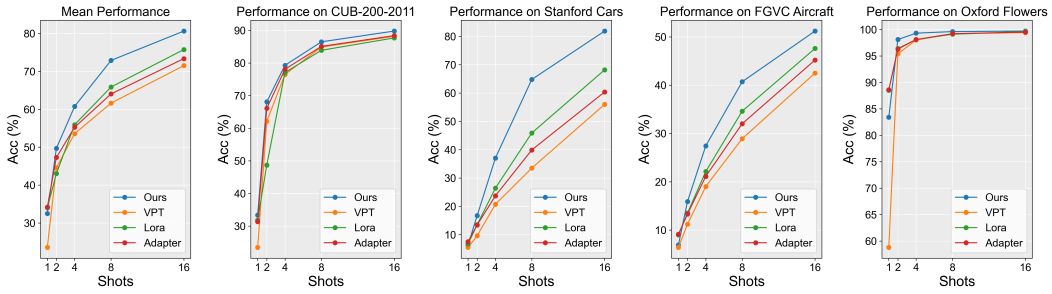


Figure 5: Comparison results using various separate training approaches in the context of few-shot learning on the multi-task FGVC datasets.

D Detailed Descriptions for the Evaluation Datasets

We provide detailed descriptions about the datasets used for evaluation. The train/val/test splits and the number of classes are summarized in Table 8.

Table 8: The statistics of the datasets used for evaluation. The train/val/test splits are the same as depicted in [6]. '-' indicates that the corresponding split is not available.

Dataset	Description	#Classes	Training set	Val. set	Test set
Multi-task FGVC					
CUB-200-2011 [39]	Bird species recognition	200	5,994	-	5,794
Stanford Cars [40]	Car classification	196	8,144	-	8,041
FGVC-Aircraft [41]	Aircraft classification	100	6,667	-	3,333
Oxford Flowers [42]	Flower species recognition	102	2,040	-	6,149
Specialized VTAB-1k [43]					
Patch Camelyon [97]	Specialized	2	800/1,000	200	32,768
EuroSAT [98]		10			5,400
Resisc45 [99]		45			6,300
Retinopathy [100]		5			42,670

By following Erudite [6], we employ the Multi-task Fine-Grained Visual Classification datasets to evaluate the performance of our proposed EMTAL, which consists of CUB-200-2011 [39], Stanford Cars [40], FGVC-Aircraft [41] and Oxford Flowers [42]. We also employ *Specialized VTAB-1k* [43], consisting of specialist images from specialized equipment, where multi-task learning is applied to fully leverage these annotated data.

E Limitations and Broader Impacts

Our work may have the following potential impacts. Firstly, compared to traditional multi-task learning approaches, EMTAL maximizes the use of limited data for downstream multi-task learning. This capability facilitates the rapid transfer of large models pre-trained on vast datasets to downstream tasks, significantly conserving computational resources. Secondly, our method is designed based on reparameterization, allowing the model to be transferred to downstream tasks without altering the deployed backbone architecture. This approach, which involves simply replacing a set of weights, is more convenient than many multi-task learning methods [4, 13, 5].

Regarding limitations, our model structure currently employs a unified rank across all experts. Despite that this design benefits parallel computing during training, our experiments reveal that different tasks have varying difficulties, necessitating different optimal ranks. For example, the optimal rank for the CUB-200-2011 dataset is one, whereas for the Stanford Cars dataset, it is four. Thus, dynamically and adaptively selecting the appropriate ranks for different tasks to improve multi-task learning is a promising research direction. Additionally, we observed that the low-rank property of shallow experts is more pronounced than that of deep experts in the network, suggesting that dynamically adjusting the rank of experts across different layers could lead to improved performance. Consequently,

adaptively assigning the appropriate rank to these experts is a potential key focus of future research. Furthermore, while our method seeks to establish a unified multi-task model for practical applications, it is essential to consider potential out-of-distribution (OOD) issues that may arise during deployment, as they could impact the performance. To address this challenge and enhance the applicability of our work in real-world scenarios, future research could explore the integration of domain adaptation techniques [101, 102]. Investigating datasets such as WILDS [103], which are tailored for OOD challenges, could provide valuable insights and further improve performance across diverse contexts.

https://doi.org/10.52202/079017-2579

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer:[Yes]

   Justification: We describe it in Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We describe it in Section E.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: No theoretical result.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We describe it in Section 3.2 and will release code upon acceptance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

https://doi.org/10.52202/079017-2579

Answer: [No]

Justification: We describe details of public datasets in Section D, provide details of our method, and will release code upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We describe it in Section D.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We describe it in Section 3.5.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We describe it in Section 3.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss it in Section E.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

https://doi.org/10.52202/079017-2579

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The creators or original owners of assets (e.g., code, data, models), used in the paper, are properly credited and the license and terms of use explicitly are mentioned and properly respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-2579