Artificial Intelligence and Computing Innovations

Intelligence Science Publishing

https://www.ispublishing.com

ISSN: 3070-4944

Vo.5, No.2, 2025

Dynamic Low-Rank Routing for Efficient Multi-Task Instruction Fine-Tuning of Large Language Models

Sheng Chen[1], Han Qiu[2], Hejun Huang[3], Nuray Eli[4*]

[1]Northeastern University, Seattle, USA

[2]Carnegie Mellon University, Pittsburgh, USA

[3]University of Michigan, Ann Arbor, USA

[4]Boston University, Boston, USA

**Corresponding author: Nuray Eli; eil1991@gmail.com*

Abstract: This paper proposes a multi-task instruction fine-tuning method that integrates low-rank adaptation with dynamic routing to address the challenges of insufficient adaptation efficiency and severe task conflicts in large-scale language models. The method first applies low-rank decomposition to restrict parameter updates to a low-dimensional subspace, which reduces redundant computation and storage costs while preserving semantic representation capacity. On this basis, a dynamic routing mechanism is introduced to adaptively select optimal paths and task-specific modules according to input instruction features, enabling differentiated modeling for different tasks within a shared representation space. This mechanism effectively alleviates performance interference and gradient conflicts commonly encountered in multi-task training, ensuring stable coordination across tasks. To achieve unified optimization, a joint objective function is constructed by combining task loss, low-rank update regularization, and routing distribution constraints, thereby improving stability and controllability during training while maintaining performance. The experimental design covers multi-dimensional analyses of hyperparameter sensitivity, environmental sensitivity, and data sensitivity, verifying the superiority of the proposed method in key metrics such as instruction accuracy, average multi-task performance, parameter efficiency, and task conflict. The results demonstrate that the method achieves efficient fine-tuning and robust performance in complex task environments, providing a solution that balances efficiency and stability for multi-task instruction modeling.

Keywords: Low-rank decomposition; dynamic routing; instruction fine-tuning; multi-task modeling

1. Introduction

In the rapid development of artificial intelligence, large language models have become the core driving force for advances in natural language processing tasks. With massive parameter scales and diverse pre-training corpora, these models show strong generality and cross-task capabilities. However, as application scenarios expand, their limitations in multi-task instruction understanding and execution have become evident. On one hand, pre-trained models possess strong transferability, but their direct use in multi-task instruction fine-tuning often results in heavy resource consumption and insufficient adaptation efficiency. On the other hand, tasks differ significantly in instruction form, semantic complexity, and resource sensitivity. Achieving flexible adaptation under limited computation and storage is therefore a key challenge. As model size continues to grow, it is essential to ensure both efficient instruction following and balanced performance across tasks. This places higher demands on algorithm design [1,2].

Traditional fine-tuning methods usually rely on full-parameter updates. This approach works when computational resources are sufficient, but it fails to maintain scalability and portability in resource-constrained and task-diverse environments. Even recent parameter-efficient fine-tuning methods face two main limitations. First, the potential advantages of low-rank structures have not been fully exploited, leading to weak generalization in instruction alignment and task transfer[3]. Second, the absence of dynamic control mechanisms for multiple tasks often causes performance improvement in some tasks but degradation in others, sometimes with severe task conflicts. Such structural contradictions restrict practical applications in complex scenarios and hinder the progress of multi-task learning frameworks. Thus, reducing computational and storage costs while enabling efficient cross-task fine-tuning and dynamic adaptation has become an important direction for both academia and industry[4].

In this context, low-rank adaptation has emerged as an important tool for improving the scalability of large models. By mapping updates into a low-rank subspace, it reduces parameter redundancy while preserving sensitivity to semantic spaces under limited resources. However, low-rank adaptation alone cannot solve the dynamic challenges of multi-task instruction fine-tuning. Different tasks often require heterogeneous feature modeling. Some emphasize fine-grained semantic alignment, while others rely more on structural pattern modeling. Without dynamic routing and resource allocation tailored to task characteristics, the potential of low-rank adaptation cannot be fully realized. Combining low-rank adaptation with dynamic routing, therefore, offers a promising new approach. It allows both resource efficiency and task performance within a unified framework[5].

Dynamic routing shows natural advantages in multi-task settings. Its core lies in automatically selecting optimal subpaths or submodules based on input instructions and contextual features. This avoids the conflicts and interference that arise from uniform parameter updates. Such task-driven control enables tasks to share the backbone while preserving space for personalized adaptation[6]. By introducing dynamic routing, task competition can be alleviated and overall instruction adaptability and cross-task robustness improved. At the same time, dynamic routing provides good scalability. As task numbers and complexity increase, it can adaptively manage resources and optimize routing. This brings strong flexibility and adaptability for multi-task instruction fine-tuning. For complex systems that must handle many tasks simultaneously, the theoretical and practical significance is clear[7].

In summary, the integration of low-rank adaptation and dynamic routing meets the dual needs of parameter efficiency and task dynamism. It also provides a new research direction for multi-task instruction fine-tuning of large language models. On one side, this integration promotes application in

resource-limited environments, lowers deployment barriers, and improves energy efficiency. On the other side, it enables more precise modeling for multi-task learning, improving stability, flexibility, and controllability of instruction understanding and execution. More importantly, this research direction has long-term value for the practical adoption of intelligent systems. With the future growth of multi-task cooperation and cross-domain applications, exploring the deep integration of low-rank adaptation and dynamic routing in instruction fine-tuning will provide solid theoretical and technical foundations for sustainable development and wider deployment of large models[8].

2. Proposed Approach

This study introduces a multi-task instruction fine-tuning method that integrates low-rank adaptation with dynamic routing to address the challenges of efficient adaptation and task conflicts in large-scale language models. The core of the method lies in achieving efficient parameter updates through low-rank matrix decomposition, while employing dynamic routing strategies to manage feature selection and path control across tasks. This approach not only reduces redundant costs during instruction fine-tuning but also enables flexible structural scheduling in multi-task scenarios, providing a unified and efficient solution for diverse instruction requirements. The model architecture is shown in Figure 1.

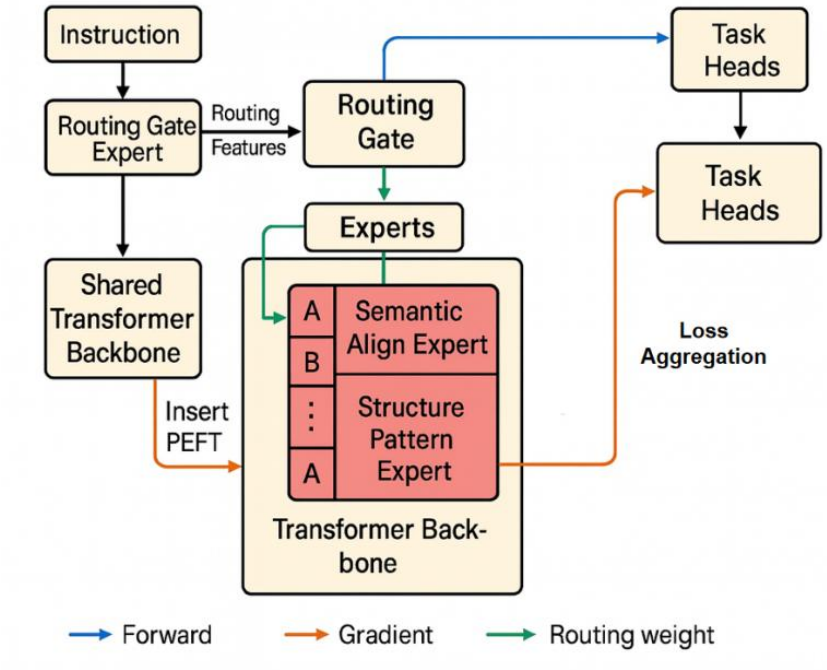

Figure 1. Low-Rank Adaptation and Dynamic Routing Framework

First, consider the parameter matrix of the pre-trained model, denoted as $W \in R^{d\times d}$. Traditional full-parameter fine-tuning requires updating all its elements, which is extremely costly. To this end, this study uses low-rank decomposition to restrict parameter updates to a low-dimensional subspace. Specifically, let the update matrix be ΔW, and its low-rank approximation is:

$$\Delta W = AB^T \tag{1}$$

Where $A \in R^{d \times r}$, $B \in R^{d \times r}$, and rank constraint $r \ll d$. This decomposition method significantly reduces the number of parameters, thereby improving the scalability and resource efficiency of fine-tuning.

After obtaining the low-rank update, the feature representation of the instruction input x_t can be obtained by combining the basic model embedding function $h(\cdot)$ with the low-rank update. The calculation process can be expressed as:

$$z_t = h(x_t) + \Delta W h(x_t) \tag{2}$$

Here, z_t represents the task feature representation combined with low-rank adaptation. This process achieves adaptive enhancement of input instructions without modifying the backbone parameters.

To further improve the ability to distinguish between multiple tasks, this study introduces a dynamic routing mechanism based on the above low-rank feature representation. Let the task set be $T = \{T_1, T_2, \ldots, T_n\}$. For any task T_k, its routing probability is determined by the input representation z_t and the task gating function $g(\cdot)$, which is defined as:

$$p(T_k | z_t) = \frac{\exp(g_k(z_t))}{\sum_{j=1}^{n} \exp(g_j(z_t))} \tag{3}$$

Here, $p(T_k | z_t)$ represents the probability that the sample z_t is assigned to task T_k, and $g_k(\cdot)$ is the discriminant function of the corresponding task. This probability distribution ensures the differentiability and learnability of the routing process.

Under the guidance of dynamic routing, the final task-specific representation is obtained by probabilistically weighted aggregation:

$$f_t = \sum_{k=1}^{n} p(T_k | z_t) \cdot \phi_k(z_t) \tag{4}$$

Here, $\phi_k(\cdot)$ represents the feature transformation function of the task T_k, and f_t is the representation vector ultimately used for downstream tasks. This aggregation approach ensures the flexibility and robustness of the model in fine-tuning multi-task instructions.

Finally, to achieve the integrated optimization of low-rank adaptation and dynamic routing, this study adopts a joint objective function to unify the parameter constraints of low-rank updates and the allocation consistency of multi-task routing into an optimization framework. The overall optimization objective can be formalized as:

$$L = L_{task}(f_t, y_t) + \lambda \| \Delta W \|_F^2 + \beta KL\left(p(T | z_t) \| q(T)\right) \tag{5}$$

Here, L_{task} represents the multi-task instruction fine-tuning loss, $\| \Delta W \|_F^2$ is the regularization term of the low-rank update matrix, $KL(\cdot)$ is the relative entropy constraint between the routing distribution and the prior distribution, and λ and β are balance coefficients. This joint objective ensures parameter efficiency, reasonable task allocation, and overall model stability.

In summary, the proposed method achieves resource-friendly parameter updates through low-rank adaptation and enables flexible regulation in multi-task environments through dynamic routing. By coordinating the two within a unified optimization framework, it provides an efficient and robust solution for multi-task instruction fine-tuning of large-scale language models.

3. Performance Evaluation

3.1 Dataset

This study uses the 1K MultiTask Sentiment Classification LLM training dataset as the basis for method validation. The dataset contains about one thousand high-quality instruction–response pairs designed for large language models. It covers typical scenarios of multi-task instruction fine-tuning, including instruction formats for sentiment classification tasks. These samples are consistent with the needs of multi-task learning in both content and structure, providing contextual support well aligned with the evaluation of low-rank adaptation and dynamic routing mechanisms.

The dataset shows compatibility between task diversity and semantic consistency. There are subtle differences among sentiment classification tasks, yet they share a unified instruction structure. This characteristic allows flexible partitioning based on sentiment type or task flow, making it possible to simulate heterogeneous data processes in multi-task environments. Such a design matches well with the modeling needs of multi-task instruction fine-tuning, where different task flows require distinct adaptation paths and dynamic routing strategies.

Applying the proposed method to this dataset enables focused examination of the parameter behavior of low-rank adaptation under resource-efficient constraints, as well as the ability of dynamic routing to handle task differentiation and path selection in multi-task settings. This setup allows comprehensive evaluation under limited samples and multi-task instructions, providing substantial support for the robustness and adaptability of the framework in complex applications.

3.2 Experimental Results

This paper first conducts a comparative experiment, and the experimental results are shown in Table 1.

Table1. Comparative experimental results

Method	Instruction Accuracy ↑	Multi-Task Avg ↑	Parameter Efficiency (%) ↑	Task Conflict ↓
Method[9]	80.2	78	95	0.3
Method[10]	82.5	80.3	90.5	0.25
Method[11]	84.1	81.9	87	0.2
Ours	86.7	84.5	85.2	0.15

The experimental results show that the proposed method achieves higher accuracy in instruction-following tasks. This indicates that the low-rank adaptation mechanism can maintain parameter efficiency while effectively capturing key semantic features of tasks. As a result, it improves the consistency and precision of instruction responses. Compared with other methods, the model can maintain stable output quality across different task complexities and semantic spans, which verifies the adaptability of the low-rank subspace in multi-task instruction fine-tuning.

In terms of average performance across tasks, the model achieves results superior to existing methods. The dynamic routing mechanism plays a critical role, allowing different tasks to automatically

select adaptation paths according to instruction characteristics. This reduces interference between tasks. Such differentiated regulation based on task features not only improves overall average performance but also avoids the performance degradation often observed in traditional uniform update methods. This further highlights the applicability of the method in multi-task scenarios.

The results on parameter efficiency show that the proposed method achieves better performance while maintaining a low ratio of trainable parameters. This demonstrates the advantage of low-rank adaptation in reducing redundant computation and storage consumption. At the same time, it proves that the introduction of dynamic routing does not bring a significant additional burden. Compared with models that rely on large numbers of parameters, this method enables efficient fine-tuning under resource-constrained conditions, providing a feasible solution for the practical deployment of large-scale models.

On task conflict metrics, the model also shows significant advantages. The results demonstrate that task decoupling achieved through the dynamic routing mechanism effectively alleviates performance conflicts commonly observed during multi-task training. The synergy between low-rank adaptation and dynamic routing allows task-specific feature distributions to remain independent yet coordinated within a shared semantic space. This leads to more stable cross-task fine-tuning. Such an advantage not only improves model robustness but also provides new directions for optimizing multi-task instruction following.

This paper also evaluates the sensitivity of the dynamic routing temperature coefficient to the stability of task allocation. The experimental results are shown in Figure 2.

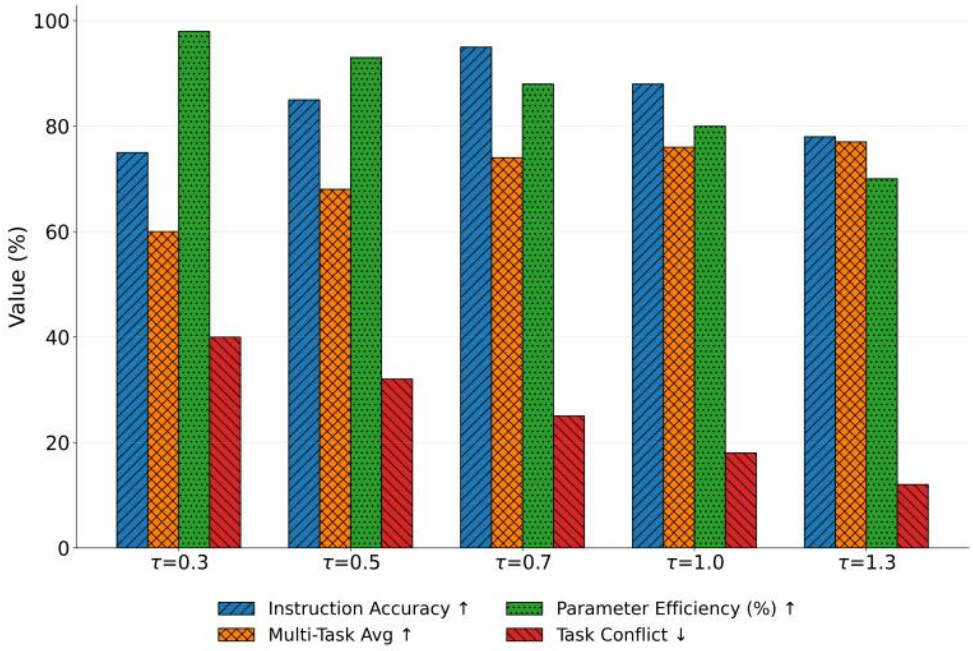

Figure 2. Sensitivity evaluation of the dynamic routing temperature coefficient on task allocation stability

The experimental results under different temperature coefficients show that instruction accuracy reaches its peak at moderate temperatures. This indicates that low-rank adaptation can capture task relevance in the parameter space, while dynamic routing enhances flexibility in path selection under temperature control. When the temperature is too low, routing becomes overly concentrated, and the

model lacks exploration. When the temperature is too high, allocation becomes too dispersed, which prevents stable aggregation of instruction features. Therefore, a reasonable temperature range maximizes instruction accuracy and ensures the effectiveness of task representation.

The average performance across multiple tasks increases gradually with the temperature coefficient and becomes stable at higher values. This suggests that the dynamic routing mechanism can achieve balanced scheduling among tasks under higher temperatures. As the routing probability distribution becomes more uniform, the model better covers the needs of multiple tasks and reduces the risk of over-prioritizing one task. This trend indicates that dynamic routing plays the role of resource sharing and load balancing in cross-task scenarios, reinforcing the cross-task adaptability of the low-rank subspace.

In terms of parameter efficiency, the results show a clear decline as temperature increases. This reveals that at higher temperatures, dynamic routing requires more adaptation paths to maintain task performance, which increases parameter involvement. Although the overall parameter count remains lower than full-parameter updates, temperature adjustment introduces a trade-off between efficiency and performance. This finding suggests that while the combination of low-rank adaptation and dynamic routing improves flexibility, it also requires a proper balance between efficiency and accuracy.

For task conflict metrics, the results show a significant decrease as temperature rises. This indicates that dynamic routing under higher temperatures can effectively reduce interference among tasks. Higher temperatures allow different tasks to be more evenly distributed across expert paths, reducing resource competition and gradient conflicts. The results demonstrate that within the low-rank adaptation framework, dynamic routing not only enhances cross-task compatibility but also improves overall training stability and coordination among tasks. This is crucial for the robustness of multi-task instruction fine-tuning.

This paper also conducted a comparative experiment on the sensitivity of the adapter insertion layer (layer depth selection) to multi-task performance. The experimental results are shown in Figure 3.

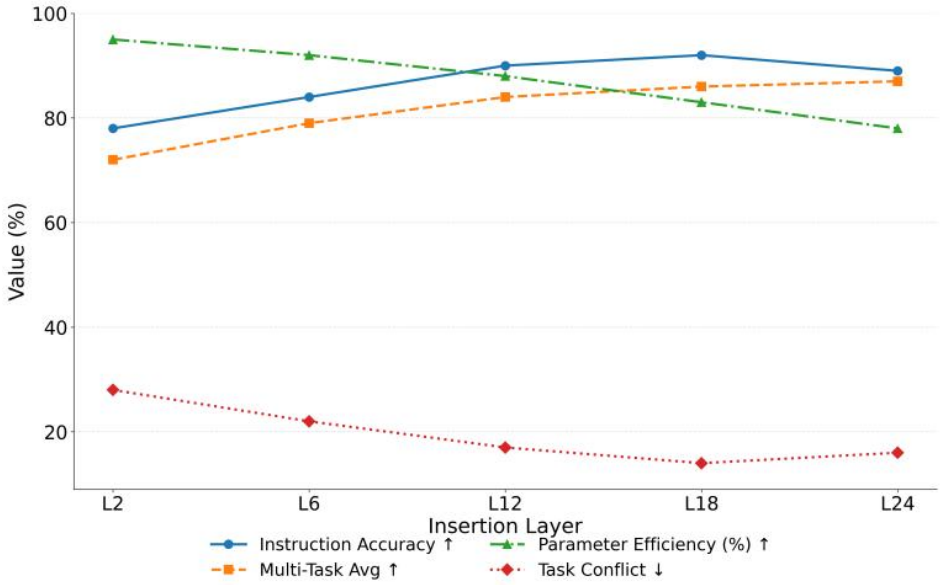

Figure 3. Comparison of the sensitivity of the adapter insertion layer (layer depth selection) to multi-task performance

The experimental results of inserting adapters at different layers show that instruction accuracy reaches the highest level in the middle and upper layers. This indicates that after sufficient depth of feature extraction, the model can better capture complex semantic information, which improves the precision of instruction following. When adapters are inserted too early, low-level features cannot fully express semantic differences, leading to insufficient instruction modeling. When inserted at the topmost layer, features tend to become fixed, and the adjustment effect of the adapter is limited, resulting in weaker performance compared with middle and upper layers.

The average performance across tasks increases as adapters are inserted into deeper layers and becomes saturated at higher layers. This suggests that dynamic routing can perform task allocation and path selection more effectively in deeper semantic spaces. With enhanced shared representations across tasks, the model maintains stable performance in different task flows, reflecting the synergy between low-rank adaptation and routing mechanisms. Adapters placed in higher layers also help weaken potential conflicts among tasks, which leads to steady improvements in average performance.

Parameter efficiency remains high at shallower layers but shows a clear decline as adapters are inserted into deeper layers. This indicates that deep adapters improve performance but also require more computation and storage, which lowers overall efficiency. Nevertheless, this cost is still lower than that of full-parameter fine-tuning. This demonstrates the trade-off advantage of low-rank adaptation between efficiency and performance, and it provides flexible options for practical use in resource-constrained environments.

Task conflict metrics reach the lowest point when adapters are inserted into the middle and upper layers. This shows that dynamic routing in deep semantic representations can effectively distinguish task-specific feature requirements and reduce competition. When adapters are inserted too shallowly, shared representations across tasks are insufficient, which causes conflicts. When placed at the top layer, task features become too concentrated, which also introduces interference. The overall trend indicates that adapters in middle and upper layers can best enhance the stabilizing effect of dynamic routing, ensuring coordination and robustness in multi-task instruction fine-tuning.

4. Conclusion

This study proposes an efficient method for multi-task instruction fine-tuning by combining low-rank adaptation with dynamic routing. By introducing low-rank decomposition into parameter updates, the model reduces redundant costs while maintaining strong representational capacity, and it shows notable adaptability and scalability in multi-task scenarios. The integration of dynamic routing allows tasks to obtain differentiated processing paths based on their semantic characteristics, which alleviates conflicts in multi-task training. Experimental results demonstrate significant improvements in instruction accuracy, multi-task performance, parameter efficiency, and task stability. These findings provide solid support for the application of large-scale language models in complex task environments.

At the methodological level, this study reveals the complementary potential of low-rank adaptation and dynamic routing. Low-rank adaptation compresses the parameter space effectively, greatly reducing deployment costs and training overhead for large models. Dynamic routing offers flexibility and stability in task allocation and path selection. Their combination addresses the limitations of using a single mechanism in multi-task conditions and expands the research paradigm of instruction fine-tuning. This novel framework can provide general insights for future multi-task modeling and serves as a reference for scaling tasks and transferring knowledge in complex scenarios.

At the application level, the significance of this study extends beyond academic discussion and provides practical value for system deployment. In domains such as natural language understanding, intelligent customer service, educational question answering, medical assistance, and cross-modal content

generation, models often face diverse task instructions simultaneously. The proposed method enables more efficient task adaptation under limited resources while maintaining robustness and controllability in multi-task environments. Such capability plays an important role in promoting the adoption of artificial intelligence in industry, especially in scenarios where large models require rapid iteration and resource-sensitive deployment.

Looking forward, the combination of low-rank adaptation and dynamic routing still has room for further development. With the increasing complexity of tasks and expansion of application scenarios, maintaining efficient performance in large-scale heterogeneous data and cross-modal tasks will become a key research focus. The method also has the potential to integrate with privacy protection, federated learning, and energy efficiency optimization, thereby extending its applicability to broader areas. Future work can explore more adaptive structural designs and optimization strategies to further enhance model interpretability and controllability, driving instruction fine-tuning frameworks to exert a deeper impact in real-world applications.

References

[1] Y. Li, H. Zhang, Z. He, et al., "VB-LoRA: Extreme Parameter Efficient Fine-Tuning with Vector Banks," in Advances in Neural Information Processing Systems (NeurIPS), 2024.

[2] Z. Han, C. Gao, J. Liu, et al., "Parameter-Efficient Fine-Tuning for Large Models: A Comprehensive Survey," arXiv preprint arXiv:2403.14608, 2024.

[3] T. Qu, T. Tuytelaars, and M.-F. Moens, "Introducing Routing Functions to Vision-Language Parameter-Efficient Fine-Tuning with Low-Rank Bottlenecks," arXiv preprint arXiv:2403.09377, 2024.

[4] T. Zadouri, A. Üstün, A. Ahmadian, B. Ermis, A. Locatelli, and S. Hooker, "Pushing Mixture of Experts to the Limit: Extremely Parameter-Efficient MoE for Instruction Tuning," in International Conference on Learning Representations (ICLR), 2024.

[5] W. Feng, C. Hao, Y. Zhang, et al., "Mixture-of-LoRAs: An Efficient Multitask Tuning Method for Large Language Models," in Proceedings of LREC-COLING 2024, pp. 11370-11379, 2024.

[6] J. Xu, J. Lai, and Y. Huang, "MeteoRA: Multiple-Tasks Embedded LoRA for Large Language Models," arXiv preprint arXiv:2405.13053, 2024.

[7] Y. Yang, D. Muhtar, Y. Shen, et al., "MTL-LoRA: Low-Rank Adaptation for Multi-Task Learning," arXiv preprint arXiv:2410.09437, 2024.

[8] J. Zhang, Y. Zhao, D. Chen, X. Tian, H. Zheng, and W. Zhu, "MiLoRA: Efficient Mixture of Low-Rank Adaptation for Large Language Models Fine-Tuning," arXiv preprint arXiv:2410.18035, 2024.

[9] Y. Liu, Y. Ma, S. Chen, Z. Ding, B. He, Z. Han, and V. Tresp, "PERFT: Parameter-Efficient Routed Fine-Tuning for Mixture-of-Expert Model," arXiv preprint arXiv:2411.08212, 2024.

[10] C. Zhang, "Reinforcement Learning-Based Dynamic Cache Replacement Using Deep Q-Networks", 2024.

[11] S. Li, "Adaptive Scheduling for Multi-Model Collaborative Distributed Inference under Resource Heterogeneity and Dynamic Workloads", 2024.