EvoMoE: Expert Evolution in Mixture of Experts for Multimodal Large Language Models

Linglin Jing[2], Yuting Gao[1], Zhigang Wang[2], Wang Lan[1], Yiwen Tang[2], Weiyun Wang[2], Wenhai Wang[2], Qingpei Guo[1]*

[1]Ant Group,
[2]Shanghai AI Lab
{jinglinglin}@pjlab.org.cn

Abstract

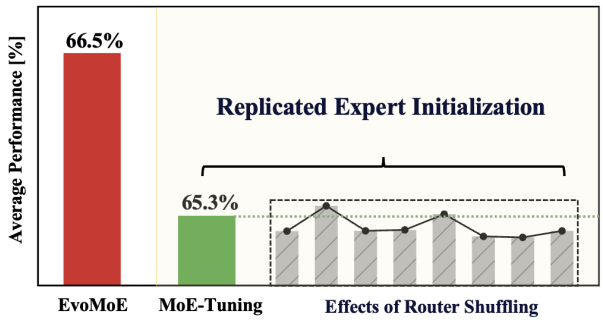
Recent advancements have shown that the Mixture of Experts (MoE) approach significantly enhances the capacity of large language models (LLMs) and improves performance on downstream tasks. Building on these promising results, multi-modal large language models (MLLMs) have increasingly adopted MoE techniques. However, existing multi-modal MoE tuning methods typically face two key challenges: ***expert uniformity*** and ***router rigidity***. Expert uniformity occurs because MoE experts are often initialized by simply replicating the FFN parameters from LLMs, leading to homogenized expert functions and weakening the intended diversification of the MoE architecture. Meanwhile, router rigidity stems from the prevalent use of static linear routers for expert selection, which fail to distinguish between visual and textual tokens, resulting in similar expert distributions for image and text. To address these limitations, we propose EvoMoE, an innovative MoE tuning framework. EvoMoE introduces a meticulously designed expert initialization strategy that progressively evolves multiple robust experts from a single trainable expert, a process termed expert evolution that specifically targets severe expert homogenization. Furthermore, we introduce the Dynamic Token-aware Router (DTR), a novel routing mechanism that allocates input tokens to appropriate experts based on their modality and intrinsic token values. This dynamic routing is facilitated by hypernetworks, which dynamically generate routing weights tailored for each individual token. Extensive experiments demonstrate that EvoMoE significantly outperforms other sparse MLLMs across a variety of multi-modal benchmarks, including MME, MMBench, TextVQA, and POPE. Our results highlight the effectiveness of EvoMoE in enhancing the performance of MLLMs by addressing the critical issues of expert uniformity and router rigidity.

Introduction

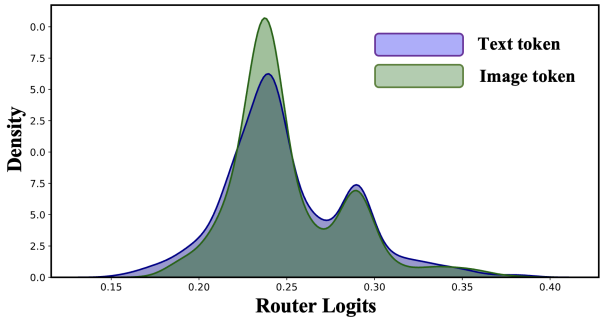
Multi-modal Large Language Models (MLLMs), such as GPT-4 (Achiam et al. 2023) and Llama 3 (Grattafiori et al. 2024), have achieved significant success in addressing open-world tasks, thanks to their scaled-up architectures. However, scaling up models often increases computational demands and is limited by device capacity. To address these challenges, sparsely activated mixture-of-expert (MoE) models have gained popularity in large language models (LLMs) (Zhai et al. 2023a; Zhong et al. 2024; Qu et al. 2025b, 2024, 2025a; Jing et al. 2022; Tang et al. 2024, 2025), reducing computational costs and enhancing efficiency. MoE models typically include multiple experts and a routing network that selects the optimal expert for each input token. This design minimizes interference among diverse input tokens, enabling each expert to specialize more effectively in specific tasks. For example, DeepSeek V3 (Liu et al. 2024a) employ MoE language models with 671B parameters, activating 37B parameters, and have achieved notable results.

The success of MoE in LLMs has spurred significant interest in its application to MLLMs (Liu et al. 2024a; Rang et al. 2025; Yang et al. 2024a; Zhang et al. 2024; Zong et al. 2024). A recent prominent approach is MoE-LLaVA (Lin et al. 2024), which introduces MoE-tuning, a multi-stage process that converts dense MLLM models into sparse MoE structures during instruction tuning, specifically tailored for multi-modal tasks. However, in MoE-tuning, experts are typically initialized by replication, leading to the first critical challenge: ***expert uniformity***. This results in expert homogenization during multi-stage tuning, thereby impeding specialization and undermining the efficacy of the MoE framework. Figure 1a illustrates an experiment on expert uniformity, in which we repeatedly shuffled the logits across router layers during evaluation and found no significant drop in average performance. This indicates that experts, which were initially replicated from a common source, tend to become homogeneous after training rather than developing specialized functions. This finding contradicts the fundamental principle of the Mixture-of-Experts (MoE) approach, which aims to enhance task-specific performance through the use of diverse experts. Additionally, MoE-tuning commonly employs a simple linear layer for expert assignment, mimicking the approach used in LLMs. This leads to the second challenge: ***router rigidity***. The shared linear router struggles to differentiate between visual and text tokens in MLLMs, resulting in uniform predictions. This limits the model's adaptability and effectiveness in multi-modal tasks. Figure 1b illustrates this rigidity using Kernel Density Estimation (KDE) plot, revealing significant overlap in the logit distributions of image and text tokens. This overlap indicates that the router becomes inflexible during training, producing uniform output distributions regardless of the input type, thus restricting the

*Corresponding author



(a) Expert Uniformity



(b) Router Rigidity

Figure 1: Two key challenges in MoE-tuning. (a) **Expert Uniformity**: Randomly shuffling the router during inference results in negligible performance degradation, suggesting uniformity among experts derived from replicated initialization. (b) **Router Rigidity**: Kernel density estimation (KDE) of the logits for image and text tokens reveals that the linear router generates input-insensitive selections, leading to static distributions with significant overlap in the density of logits for image and text token.

model's adaptability in multi-modal tasks.

To tackle the aforementioned challenges, we introduce **EvoMoE**, a sparse MoE framework tailored for MLLM. Our method builds on MoE-tuning framework, gradually transforming dense models into MoE structures. To address the challenge of ***expert uniformity*** and enhance the diversity of expert initialization, we introduce a novel approach called Expert Evolution, which generates diverse MoE experts by iteratively adapting expert parameters through a dynamic evolution value. This evolution value integrates prior expertise with gradient-based updates, enabling continuous refinement and evolution. Consequently, the technique evolves multiple robust MoE experts from a single trainable expert. To address ***router rigidity*** and enhance the connection between the router and input modalities, we propose the Dynamic Token-aware Router (DTR). This router dynamically allocates input tokens to specific experts. Specifically, we employ a hypernetwork to generate unique parameters for each router, tailored to the unique value of each token.

Our core contributions are summarized as follows:

- We introduce EvoMoE, an innovative MoE-tuning framework specifically designed for MLLMs, which effectively addresses two critical challenges: expert uniformity and router rigidity.
- To tackle expert uniformity, we propose a novel method termed expert evolution, which flexibly generates a diverse set of MoE experts. Furthermore, to mitigate router rigidity, we introduce DTR, which assigns input tokens to specific experts based on their modality and intrinsic value.
- Extensive experiments on language models of various sizes demonstrate that EvoMoE achieves better performance with fewer activated parameters.

Related Works

Multi-modal Large Language Model

LLMs (Naveed et al. 2023; Chang et al. 2024; Jing et al. 2024b,a; Wang et al. 2025) have demonstrated outstanding capabilities in reasoning and question answering. Building on this, recent studies have extended LLMs into the visual domain, leading to the creation of MLLMs. LLaVA 1.5 (Liu et al. 2023b) marks a notable advancement in MLLMs by integrating visual and textual modalities using a simple yet effective architecture, employing a pre-trained vision encoder and language model linked by a lightweight projection layer, achieving strong performance across multi-modal tasks. Recent MLLMs continue to push the boundaries of vision-language integration through novel architectures. For instance, InternVL (Chen et al. 2024) enhances fine-grained visual-semantic alignment by decomposing high-resolution images into regional patches with a dynamic multi-scale module and fusing features through pixel-shuffle-based method. Meanwhile, scaling efforts in systems like GPT-4o (Hurst et al. 2024) and Gemini (Team et al. 2024) highlights the importance of model and data scaling, with expanded architectures showing emergent capabilities in multi-modal reasoning, code synthesis, and long-context comprehension.

Mixture of Experts in Multi-modal Learning

Given the substantial computational overhead associated with training and deploying MLLMs, researchers are increasingly turning to the MoE architecture to enhance efficiency. This approach can be broadly categorized into two main strategies: one directly integrates multi-modal capabilities into an LLM with an MoE architecture, as seen in MLLMs like Qwen2.5-VL (Bai et al. 2025), Kimi-VL (Team et al. 2025) and MiniMax-VL (Li et al. 2025), leveraging its inherent efficiency and scalability. The other strategy extends a dense LLM to an MoE-based architecture, offering greater flexibility in adapting to diverse tasks and modalities while maintaining computational efficiency. For instance, HyperLLaVA (Zhang et al. 2024) integrates additional visual experts within the vision encoder and the LLM. These hypernet-based experts dynamically capture input characteristics, thereby offering improved feature extraction capabilities within these components. LLaVA-MoLE (Chen, Jie, and Ma

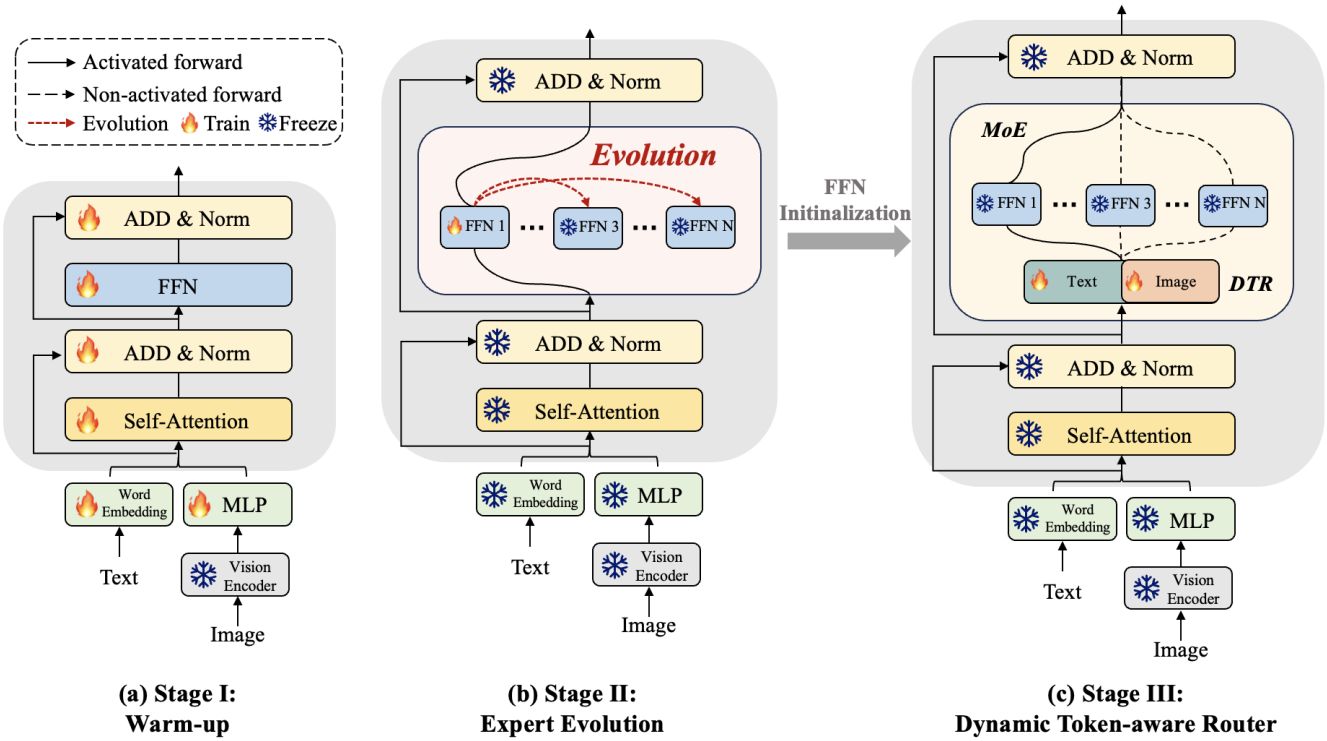


Figure 2: **The framework of EvoMoE.** EvoMoE comprises three stages of instruction-tuning: (a) Warm-up: Begin training with multi-modal instruction data to familiarize the model with understanding capabilities, utilizing parameters initialized during the MoE-LLaVA (Lin et al. 2024) pretraining stage. (b) Expert Evolution: Train only FFN1, while evolving other experts from FFN1, and (c) Dynamic Token-aware Router: Use FFNs evolved in Stage II for expert initialization and incorporate the DTR for MoE, with only the DTR trainable during this stage, while all other parameters remain frozen.

2024) creates a set of LoRA experts and a linear router for the FFN layer to mitigate data conflicts when combining multiple distinct instruction datasets. Recently, MoE-LLaVA (Lin et al. 2024) present MoE-tuning, a novel three-stage framework that progressively converts the dense FFN layers in LLMs into a MoE structure. This method incorporates a linear router, significantly lowering the required activation parameters while maintaining or even surpassing the performance of dense models. However, these MoE-tuning approaches encounter challenges concerning Expert Uniformity and Router Rigidity.

Methods

In this paper, we present EvoMoE, a novel MoE-tuning approach addressing expert uniformity and router rigidity. Section overviews the three-stage tuning framework of EvoMoE. In Section , we introduce the expert evolution strategy for generating diverse MoE experts during Stage II. Finally, Section details the Stage III routing mechanism, which dynamically assigns input tokens to suitable experts based on modality and inherent token values.

Framework Overview

In the MoE-tuning approach, the initial pre-training phase establishes cross-modal alignment by utilizing an MLP projector to map image tokens into the LLM's latent space, equipping the model with foundational visual-language understanding capabilities. Leveraging this groundwork, we introduce EvoMoE, building upon the pre-training phase of MoE-LLaVAA (Lin et al. 2024) as our initialization base. Our methodology further advances this foundation through a three-stage framework designed to systematically evolve the dense pre-trained backbone into a sparse MoE architecture. This framework incorporates two key components: (1) Expert Evolution: Gradually generate new experts using an evolution strategy. (2) Dynamic Token-aware Routing: Implement a dynamic routing mechanism that prioritizes input-relevant experts. Figure 2, coupled with the subsequent description, clarifies the details of the three-stage framework:

Stage I: Understanding Warm-up. To equip the model with basic instruction-following capabilities, we utilize a collection of instruction datasets to train all parameters of the dense LLM and the corresponding MLP layer.

Stage II: Expert Evolution. In this stage, we introduce a novel methodology for constructing Mixture-of-Experts (MoE) experts, where each expert is instantiated as a unique Feed-Forward Network (FFN) layer within the LLM. By employing expert evolution, a process that dynamically balances prior expertise with gradient-based updates, we progressively derive multiple robust FFN experts from a single trainable

Methods	LLM	Act.	Res.	Image Question Answering				Benchmark Toolkit			AVG
				VQA^{v2}	GQA	SQA	VQA^{t}	POPE	MME	MMB	
0-1B											
Sparse Model											
MoE-LLaVA*	Q′-0.5B	0.6B	336	72.0	56.1	58.0	39.6	84.4	1170.1	57.8	60.9
EvoMoE	Q′-0.5B	0.7B	336	**74.4**	**57.4**	**59.1**	**42.4**	**85.0**	**1188.6**	**58.2**	**62.3**
1-2B											
Sparse Model											
MoE-LLaVA (Lin et al. 2024)	S-1.6B	2.0B	336	76.7	60.3	62.6	50.1	85.7	1318.2	60.2	65.9
EvoMoE	S-1.6B	1.8B	336	**76.9**	61.2	**63.5**	**51.5**	86.4	**1359.7**	60.9	**67.0**
MoE-LLaVA (Lin et al. 2024)	Q-1.8B	2.2B	336	76.2	**61.5**	63.1	48.0	87.0	1281.6	59.7	65.7
MoE-LLaVA*	Q-1.8B	2.2B	336	76.2	61.0	62.6	48.0	86.5	1288.1	59.4	65.3
EvoMoE	Q-1.8B	2.0B	336	**76.9**	61.2	63.3	49.3	**87.1**	1315.6	**61.6**	66.5
2-3B											
Dense Model											
Mini-Gemini (Li et al. 2024)	G-2B	2.0B	336	-	-	-	56.2	-	1341.0	59.8	-
MobileVLM v2 (Chu et al. 2024)	ML-2.7B	2.7B	336	-	61.1	70.0	57.5	84.7	1440.5	63.2	-
LLaVA-Phi (Chu et al. 2024)	P-2.7B	2.7B	336	71.4	68.4	66.4	48.6	85.0	1335.1	59.8	66.6
Sparse Model											
Qwen-MoE* (Yang et al. 2024a)	P-2.7B	2.7B	336	77.5	61.1	67.7	52.6	85.9	1434.0	65.4	68.9
MoE-LLaVA (Lin et al. 2024)	P-2.7B	3.6B	336	77.6	61.4	68.5	51.4	86.3	1423.0	65.2	68.7
EvoMoE	P-2.7B	3.0B	336	77.8	61.6	69.5	52.0	**86.6**	**1450.5**	66.8	69.6
MoE-LLaVA (Lin et al. 2024)	P-2.7B	3.6B	384	79.9	62.6	70.3	57.0	85.7	1431.3	68.0	70.5
EvoMoE	P-2.7B	3.0B	384	**80.2**	62.8	**71.5**	**57.8**	86.5	1450.1	**69.6**	**71.6**
7B											
Sparse Model											
MoE-LLaVA*	O-7B	9.6B	336	78.1	61.5	62.8	52.7	86.8	1384.5	64.8	67.9
EvoMoE	O-7B	7.3B	336	**78.9**	**62.6**	**63.8**	**53.8**	**87.3**	**1391.5**	**65.8**	**68.8**

Table 1: **Comparison of MLLMs on image understanding benchmarks.** 'LLM' is the language model component, 'Act.' is the number of activated parameters, and 'Res.' is the input image resolution. Models 'Q' ,'Q′', 'S', 'P', 'ML' , 'G', and 'O' refer to Qwen (Bai et al. 2023), Qwen2 (Yang et al. 2024b), StableLM (Bellagente et al. 2024), Phi-2 (Javaheripi et al. 2023), Mobile LLaMA (Kan et al. 2024), Gemini (Team et al. 2024) and OpenChat (Wang et al. 2023a), respectively. 'AVG' is the weighted mean across all benchmarks, with MME values divided by 20 for simplify calculation. * indicates results re-implemented using MoE-LLaVA (Lin et al. 2024). Rows are colored based on the same baseline settings as our method for easier comparison.

FFN during training. This iterative approach enables continuous refinement and adaptation, enhancing the diversity, specialization, and robustness of the experts. As a result, the model's adaptability and overall performance are significantly improved.

Stage III: Dynamic Token-aware Router. In this stage, we introduce the Dynamic Token-aware Router (DTR), which dynamically allocates input tokens to appropriate experts based on their modality. The DTR parameters are generated by a hypernetwork, creating routing decisions specifically tailored for each input token. The experts are initialized through the evolutionary process described in Stage II.

MoE Expert Evolution

The existing MoE-tuning approach typically replicates the FFN parameters to initialize multiple MoE experts, resulting in expert uniformity issues during training. To address this challenge, as illustrated in Figure 2 (b), we propose a novel initialization strategy, expert evolution, which gradually evolves new experts by dynamically balancing prior expertise with gradient-based updates:

$$\theta_n \leftarrow \beta \cdot \theta_n + (1-\beta) \cdot \theta_1, \tag{1}$$

where θ_1 represents the network parameters of the original trainable FFN, initialized using the output of stage I and designated as expert 1. θ_n denotes the newly generated FFN experts, where the index $n = [2, 3, \cdots, N]$ represents each of the N experts. β denotes the evolution value within the range [0,1] which controls the evolution rate. A larger β emphasizes historical data, producing a smoother average by diminishing the impact of recent changes. In our experiments, β is randomly assigned a value within a specified range at each training step for improved generalization.

We exclusively train θ_1 of expert 1, allowing it to evolve into different experts through varying evolution rates β. Importantly, these evolved experts and all other LLM and MLP parameters remain frozen during training.

Dynamic Token-aware Router (DTR)

In the MoE-tuning approach, a shared linear router selects the top K experts for both visual tokens $V = [v_1, v_2, \cdots, v_P] \in \mathbb{R}^{P \times C}$ and text tokens $T = [t_1, t_2, \cdots, t_M] \in \mathbb{R}^{M \times C}$, here, P is the sequence length of visual tokens, M is the sequence length of text tokens, and C denotes the hidden size of the LLM. This setup might not differentiate between visual and text tokens, limiting effectiveness in multi-modal tasks and resulting in uniform predictions, known as router rigidity. To address this challenge, we introduce the Dynamic Token-aware Router (DTR) as a key component of the EvoMoE.

The architecture of DTR, as depicted in Figure 3, innovatively manages input visual and text tokens through two hypernetworks, denoted as $\mathcal{H}^V$ and $\mathcal{H}^T$. Each hypernetwork consists of two MLP layers, allowing it to generate adaptive parameters customized to each token. For instance, when processing visual tokens V, the hypernetwork $\mathcal{H}^V$ predicts modality-specific parameters Θ^V, optimized for visual inputs, as detailed below:

$$\Theta^V = \left(w_1 \boldsymbol{z}'^{(V)} + b_1\right) w_2 + b_2, \tag{2}$$

where $w1$ and $w2$ denote the weights for two MLPs in H^V, while b_1 and b_2 represent the corresponding biases. Similarly, Θ^T denotes the dynamic weights for the text token input, which are generated in a manner analogous to that of the visual.

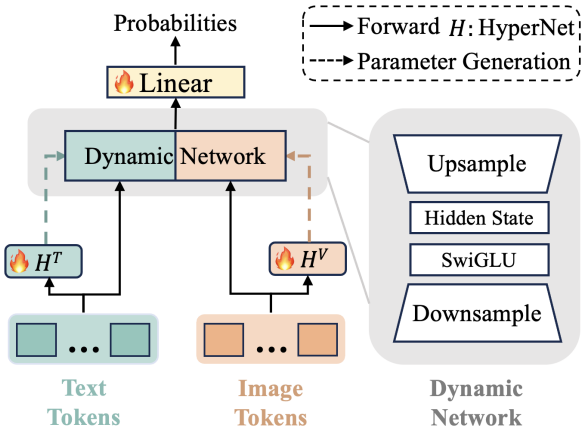


Figure 3: **Dynamic Token-aware Router (DTR).** Two input-guided hypernetworks dynamically generate network parameters for the up-sampling and down-sampling layers based on visual and text tokens. The final linear layer predicts probabilities and selects the top-k experts. In this module, only the hypernetworks and the linear layer are trainable.

DTR consists of a pair of down-sampling and up-sampling layers designed to dynamically extract visual and textual information from various input tokens. Finally, the prediction of expert probabilities ρ is formulated as follows:

$$\Theta^\tau_{\text{up}}, \Theta^\tau_{\text{down}} = \mathcal{H}^\tau\left(z^{\tau\prime}\right), \text{ where } \tau \in V, T \tag{3}$$

$$\mathcal{E}^\tau = \Theta^\tau_{\text{up}}\left(\text{SwiGLU}\left(\Theta^\tau_{\text{down}}\left(z^{\tau\prime}\right)\right)\right), \tag{4}$$

$$\rho^\tau = \left(\phi\left(\mathcal{E}^\tau\right)\right), \tag{5}$$

where τ denotes the visual and text input tokens. z' is the output of a single MSA block. Θ_{up} and Θ_{down} correspond to the weights of the up-sampling and down-sampling layers, respectively, which are generated by the hypernetwork $\mathcal{H}$. In this context, the SwiGLU activation function is utilized. The symbol ϕ denotes an MLP layer that serves as the final router. During training, only $\mathcal{H}^V$, $\mathcal{H}^T$, and ϕ are fine-tuned. Each token is processed by the top K experts with the highest probability.

Training Objective

In alignment with (Lin et al. 2024), the overall loss function of EvoMoE consists of two components: the regression loss: $\mathcal{L}_{\text{regressive}}$ and the auxiliary loss $\mathcal{L}_{\text{aux}}$. Regression loss is designed to optimize model performance, while auxiliary loss aims to promote a balanced load distribution across the:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{regressive}} + \alpha \cdot \mathcal{L}_{\text{aux}}. \tag{6}$$

Here, α is a hyperparameter that controls the weight of the auxiliary loss and is set to 0.001 during the training process. Details are provided in the supplementary materials.

Experiments

Experiments Setup

Model Details. EvoMoE is built on the MoE-tuning and LLaVA 1.5 frameworks, centering on the Evolution Strategy and the Dynamic Token-aware Router (DTR). The training framework suits various sizes, with experiments on LLMs with 0.5B, 1.8B, 2.7B, and 7B parameters showing strong generalization. Importantly, EvoMoE achieves SOTA performance by activating only top-1 expert, which offers a significant advantage in terms of the number of activated parameters.

Training Datasets. In Stage I, following MoE-LLaVA (Lin et al. 2024), we use a diverse dataset collection, including MIMIC-IT (Li et al. 2023a), LRV (Liu et al. 2023a), SViT (Zhao et al. 2023), and LVIS (Wang et al. 2023b), to enhance the MLLM's general multi-modal comprehension skills. Stage II employs the LLaVA-mix-665k (Lai et al. 2024) dataset to develop evolution experts. In Stage III, the same LLaVA-mix-665k dataset is used to train the DTR.

Evaluation. We evaluate the effectiveness and robustness of EvoMoE across diverse scenarios through performance evaluations on an extensive range of multi-modal benchmarks, including VQA-v2 (Goyal et al. 2017), GQA (Hudson and Manning 2019), SQA (Lu et al. 2022), TextVQA (Singh et al. 2019), POPE (Li et al. 2023b), MME (Fu et al. 2023), and MMBench (Liu et al. 2024b).

Implementation Details. In our experiments, CLIP-L (Radford et al. 2021) and SigLIP-L (Zhai et al. 2023b) were utilized as the image encoders. Throughout all experiments, the batch size was consistently maintained at 4, with a gradient accumulation of 2. For all the three stages of instruction tuning, the initial learning rate was 2e-5, and we consistently select the top-1 expert across all experiments. In the evolution strategy, the evolution rate was randomly sampled from one of three ranges ([0.9–0.99], [0.8–0.89], [0.7–0.79]) per training iteration, each corresponding to one expert generated via evolution. Together with the original trainable expert, this formed a total of four MoE experts.

Comparison with State-of-the-Art

We evaluated our method against state-of-the-art approaches on four image question-answering benchmarks and three multi-modal understanding toolkits. As illustrated in Table **??**, the models were categorized according to the size of LLM into four groups: 0–1B, 1–2B, 2–3B, and 7B.

	M-T(Lin et al. 2024).	Evo.	DTR	Act.	Image Question Answering				Benchmark Toolkit			AVG
					VQA^{v2}	GQA	SQA	VQA^{T}	POPE	MME	MMB	
(a)				1.8B	76.3	61.0	62.1	48.2	86.4	1286.7	59.7	65.4
(b)	✓			2.2B	76.2	61.0	62.6	48.0	86.5	1288.1	59.4	65.5
(c)	✓		✓	2.4B	76.2	61.1	63.0	48.6	86.8	1310.4	61.4	65.9
(d)		✓		**1.8B**	**77.5**	**61.2**	62.9	48.8	86.8	1311.4	61.3	66.3
EvoMoE		✓	✓	2.0B	76.9	**61.2**	**63.3**	**49.3**	**87.1**	**1315.6**	**61.6**	**66.5**

Table 2: Ablation study on MLLM evaluation benchmarks.

	β	VQA^{v2}	GQA	SQA	VQA^{T}	POPE	MME	MMB
Expert 1	1.0	76.3	61.0	62.1	48.2	86.4	1286.7	59.7
Expert 2	0.9	76.4	60.8	62.7	48.6	**87.3**	**1305.7**	58.4
Expert 3	0.8	76.7	60.9	62.4	**49.0**	86.6	1297.3	**61.4**
Expert 4	0.7	**77.1**	**61.2**	**62.8**	48.7	86.4	1284.5	59.5

Table 3: Ablation study for evolution strategy.

Compared with the SOTA method MoE-LLaVA, which serves as a baseline for MoE-tuning, EvoMoE demonstrates strong multi-modal understanding capabilities across various LLM sizes and image resolutions. EvoMoE outperforms MoE-LLaVA in the LLMs Qwen2-0.5B, StableLM-1.6B, Qwen-1.8B, Phi-2.7B, and OpenChat-7B. It achieves average performance gains of 1.4% for the 0.5B model, 1.1% for the 1.6B model, 1.2% for the 1.8B model, 1.1% for the 2.7B model and 0.9% for the 7B model, all with fewer activated parameters (activating only top-1 expert). In particular, EvoMoE achieves remarkable improvements in the TextVQA, VQAv2, and GQA benchmarks. For instance, with the Qwen2-0.5B model, it surpasses the baseline by 2.8%, 2.4%, and 1.3%, respectively. In StableLM-1.6B, EvoMoE improves TextVQA performance by 1.4%. Additionally, it outperforms baselines in MMbench evaluations by 2.2%, 1.6%, and 1.0% with Qwen-1.8B, Phi-2.7B, and OpenChat-7B models, respectively. This is particularly noteworthy, given that the baseline approach relies on activating the top-2 experts, which results in a significantly higher number of activated parameters. Ultimately, under the same Phi-2.7 LLM, EvoMoE outperformed the baseline by both 1.6% on input image resolutions of both 336 and 384, demonstrating the flexibility of EvoMoE. Collectively, these results demonstrate that EvoMoE not only outperforms other sparse models but also achieves this with fewer activated parameters.

Comprehensive Analysis

In this section, we peform an ablation study to explore EvoMoE's core contributions using the Qwen-1.8B model. We conduct experiments on four image QA benchmarks and three multi-modal understanding benchmarks, using the same training data as (Lin et al. 2024) for fair comparison.

Design analysis of our framework. We conduct several ablation studies to assess the effectiveness of the proposed framework. As depicted in Table **??**, (a) represents a dense LLM without any MoE experts. (b) is a standard MoE model proposed by MoE-LLaVA (Lin et al. 2024) based on MoE-tuning, which includes four experts and a linear router. These

	Share	Image	Text	GQA	SQA	VQA^{t}	POPE
Linear							
(a)	✓			61.0	62.6	48.0	86.5
(b)		✓	✓	**61.2**	62.7	48.3	86.6
(c)	✓	✓	✓	61.1	62.2	48.2	86.4
HyperNet							
DTR		✓	✓	**61.2**	**63.3**	**49.2**	**87.1**
(d)	✓	✓	✓	60.9	62.7	48.4	86.7

Table 4: Ablation study for DTR.

results indicate that traditional MoE-tuning method does not provide a significant performance enhancement over dense LLMs on average accuracy. This is due to two challenges faced by MoE-tuning: expert uniformity and router rigidity. In (c), we replaced the linear router with our proposed DTR router based on the MoE-tuning approach. The results indicate that DTR addresses the issue of router rigidity, thereby enhancing performance. In (d), we tested our expert evolution strategy combined with a linear router. All newly created experts originate from the same dense LLM, which can be compared to (a). The results demonstrate that our expert evolution strategy significantly enhances performance while maintaining a model size comparable to dense LLMs, effectively addressing the issue of expert uniformity.

The Effectiveness of Evolution Strategy. Table **??** validates our expert evolution strategy in MLLMs. By removing the router and fixing β to a constant value, we systematically examined its impact on performance. Expert 1 is a FFN layer with $\beta = 1.0$, while Experts 2 to 4 are evolved from Expert 1 by progressively reducing β values. Notably, the evolved experts consistently outperform Expert 1 across the majority of benchmarks. This performance advantage persists even under large β decay, where the value is set to 0.9, equivalent to retaining only 10% of updates per step. These systematic improvements empirically validate our core hypothesis: experts generated through expert evolution exhibit significantly greater diversity compared to those generated through straightforward replication. Each evolved expert

Strategy	VQA^{v2}	GQA	SQA	VQA^{t}	POPE	MME	MMB
LLM MoE Insights							
w/o first layer	76.3	61.1	62.4	48.5	86.4	1285.6	60.6
Share Expert	76.2	61.0	62.6	48.8	86.6	1306.8	60.8
Trainable Parameters							
In Stage II	75.2	**61.2**	**63.8**	46.8	86.4	1263.8	**61.9**
In Stage III	**77.0**	60.9	62.3	48.4	86.5	1271.2	60.9
MoE Placement							
ALL Layers	74.4	61.0	62.5	47.6	86.3	1280.1	60.4
EvoMoE	76.9	**61.2**	63.3	**49.3**	**87.1**	**1315.6**	61.6

Table 5: Ablation study on MoE exploration.

	Method	GQA	SQA	VQA^{t}	POPE
(a)		61.0	62.6	48.0	86.5
Initialization					
(b)	Noise	60.8	63.1	47.2	86.1
(c)	V-Evo.	61.3	63.0	48.0	86.7
Training					
(d)	Dropout	60.6	62.3	47.4	86.2
(e)	Contrastive	**61.5**	62.6	47.5	86.3
(f)	Local Loss	60.9	62.2	48.2	85.9
Ours	**Evolution**	61.2	**63.3**	**49.3**	**87.1**

Table 6: Ablation study for expert diversity.

demonstrates specialized capabilities, excelling in different benchmarks, often outperforming the original expert and effectively addressing the challenge of expert uniformity. In the subsequent stage, we apply the proposed DTR to these evolved experts, enabling better utilization of their specialized capabilities and enhancing overall performance. In our experiments, to enhance generalization, we randomly sample the β value from a predefined range at each training step, rather than using a fixed value.

Design analysis of DTR. Table 4 summarizes the ablation studies. The modality-specific router in (b) outperforms the single router in (a), emphasizing the importance of modality distinction. The HyperNet adaptation improves attention to input token distribution, further improving performance. However, adding a weighted shared router in (c) and (d) results in a decline in overall performance. Ultimately, we adopt the structure of the DTR in Figure 3 in our framework, as it achieves the best performance. The visualization reveals that traditional MoE-tuning exhibits almost uniform distributions across different inputs, leading to router rigidity. In contrast, EvoMoE, using DTR, dynamically allocates tokens to suitable experts based on modality, allowing experts to learn specific patterns for efficient, input-guided processing.

Increasing Expert Diversity. To address homogenization from expert replication, we implemented strategies to improve expert diversity, classified into initialization and training phases. As shown in Table 6, (a) shows the MoE baseline. (b) adds noise during expert initialization, while (c) Vanilla-Evolution shifts expert evolution to Stage I and fine-tunes all experts in Stage II. For training: (d) uses random dropout; (e) incorporates NCE loss among experts; and (f) introduces local loss (Mustafa et al. 2022) to increase router entropy for better routing balance. These diversity strategies didn't significantly boost performance across all metrics, highlighting EvoMoE's superiority. For detailed comparison, see the Supplementary Material.

MoE Strategy Exploration. We further explored additional attempts concerning MoE in Table **??**. Incorporating insights from advanced LLMs like DeepSeek-V3 (Liu et al. 2024a), it was found that removing the initial MoE layer, emphasized in DeepSeek-V3, is ineffective in MLLMs. While shared experts are common used in LLM MoE implementations, they have not provided significant benefits in MLLMs. Additionally, we explored the introduction of additional trainable parameters at various stages: (1) In stage II, unfreezing all parameters (MSA&FFN) within the LLM led to optimal performance on several benchmarks, though improvements were not consistent across all benchmarks. (2) In stage III, training the entire set of experts alongside the DTR led to a significant performance drop. Lastly, our framework preserved performance using an alternating approach for MoE layers, whereas replacing all dense layers with MoE structures decreased performance.

Conclusion

In this paper, we introduce EvoMoE, a MoE framework designed for MLLMs. EvoMoE redefines MoE-tuning via expert evolution and the dynamic token-aware router (DTR), effectively addressing two critical challenges in existing MoE-tuning approaches: expert uniformity and router rigidity. The superior performance of EvoMoE, validated through extensive experiments, highlights its potential to unlock new possibilities for the application of MoE in MLLMs.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Bellagente, M.; Tow, J.; Mahan, D.; Phung, D.; Zhuravinskyi, M.; Adithyan, R.; Baicoianu, J.; Brooks, B.; Cooper, N.; Datta, A.; et al. 2024. Stable lm 2 1.6 b technical report. *arXiv preprint arXiv:2402.17834*.

Chang, Y.; Wang, X.; Wang, J.; Wu, Y.; Yang, L.; Zhu, K.; Chen, H.; Yi, X.; Wang, C.; Wang, Y.; et al. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3): 1–45.

Chen, S.; Jie, Z.; and Ma, L. 2024. Llava-mole: Sparse mixture of lora experts for mitigating data conflicts in instruction finetuning mllms. *arXiv preprint arXiv:2401.16160*.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 24185–24198.

Chu, X.; Qiao, L.; Zhang, X.; Xu, S.; Wei, F.; Yang, Y.; Sun, X.; Hu, Y.; Lin, X.; Zhang, B.; and Shen, C. 2024. MobileVLM V2: Faster and Stronger Baseline for Vision Language Model. arXiv:2402.03766.

Fu, C.; Chen, P.; Shen, Y.; Qin, Y.; Zhang, M.; Lin, X.; Qiu, Z.; Lin, W.; Yang, J.; Zheng, X.; Li, K.; Sun, X.; and Ji, R. 2023. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. *ArXiv*, abs/2306.13394.

Goyal, Y.; Khot, T.; Summers-Stay, D.; Batra, D.; and Parikh, D. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6904–6913.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Hudson, D. A.; and Manning, C. D. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6700–6709.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Javaheripi, M.; Bubeck, S.; Abdin, M.; Aneja, J.; Bubeck, S.; Mendes, C. C. T.; Chen, W.; Del Giorno, A.; Eldan, R.; Gopi, S.; et al. 2023. Phi-2: The surprising power of small language models. *Microsoft Research Blog*, 1(3): 3.

Jing, L.; Ding, Y.; Gao, Y.; Wang, Z.; Yan, X.; Wang, D.; Schaefer, G.; Fang, H.; Zhao, B.; and Li, X. 2024a. HPL-ESS: hybrid pseudo-labeling for unsupervised event-based semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 23128–23137.

Jing, L.; Wang, Y.; Chen, T.; Dora, S.; Ji, Z.; and Fang, H. 2022. Towards a more efficient few-shot learning-based human gesture recognition via dynamic vision sensors. In *BMVC*, 938.

Jing, L.; Xue, Y.; Yan, X.; Zheng, C.; Wang, D.; Zhang, R.; Wang, Z.; Fang, H.; Zhao, B.; and Li, Z. 2024b. X4d-sceneformer: Enhanced scene understanding on 4d point cloud videos through cross-modal knowledge transfer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2670–2678.

Kan, K. B.; Mun, H.; Cao, G.; and Lee, Y. 2024. Mobile-llama: Instruction fine-tuning open-source llm for network analysis in 5g networks. *IEEE Network*.

Lai, X.; Tian, Z.; Chen, Y.; Li, Y.; Yuan, Y.; Liu, S.; and Jia, J. 2024. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9579–9589.

Li, A.; Gong, B.; Yang, B.; Shan, B.; Liu, C.; Zhu, C.; Zhang, C.; Guo, C.; Chen, D.; Li, D.; et al. 2025. Minimax-01: Scaling foundation models with lightning attention. *arXiv preprint arXiv:2501.08313*.

Li, B.; Zhang, Y.; Chen, L.; Wang, J.; Pu, F.; Yang, J.; Li, C.; and Liu, Z. 2023a. Mimic-it: Multi-modal in-context instruction tuning. *arXiv preprint arXiv:2306.05425*.

Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W. X.; and Wen, J.-R. 2023b. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Li, Y.; Zhang, Y.; Wang, C.; Zhong, Z.; Chen, Y.; Chu, R.; Liu, S.; and Jia, J. 2024. Mini-Gemini: Mining the Potential of Multi-modality Vision Language Models. arXiv:2403.18814.

Lin, B.; Tang, Z.; Ye, Y.; Cui, J.; Zhu, B.; Jin, P.; Huang, J.; Zhang, J.; Pang, Y.; Ning, M.; et al. 2024. Moe-llava: Mixture of experts for large vision-language models. *arXiv preprint arXiv:2401.15947*.

Liu, A.; Feng, B.; Xue, B.; Wang, B.; Wu, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Liu, F.; Lin, K.; Li, L.; Wang, J.; Yacoob, Y.; and Wang, L. 2023a. Aligning large multi-modal model with robust instruction tuning. *CoRR*.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2023b. Improved Baselines with Visual Instruction Tuning.

Liu, Y.; Duan, H.; Zhang, Y.; Li, B.; Zhang, S.; Zhao, W.; Yuan, Y.; Wang, J.; He, C.; Liu, Z.; et al. 2024b. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, 216–233. Springer.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35: 2507–2521.

Mustafa, B.; Riquelme, C.; Puigcerver, J.; Jenatton, R.; and Houlsby, N. 2022. Multimodal contrastive learning with limoe: the language-image mixture of experts. *Advances in Neural Information Processing Systems*, 35: 9564–9576.

Naveed, H.; Khan, A. U.; Qiu, S.; Saqib, M.; Anwar, S.; Usman, M.; Akhtar, N.; Barnes, N.; and Mian, A. 2023. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*.

Qu, D.; Song, H.; Chen, Q.; Chen, Z.; Gao, X.; Ye, X.; Lv, Q.; Shi, M.; Ren, G.; Ruan, C.; et al. 2025a. EO-1: Interleaved Vision-Text-Action Pretraining for General Robot Control. *arXiv preprint arXiv:2508.21112*.

Qu, D.; Song, H.; Chen, Q.; Yao, Y.; Ye, X.; Ding, Y.; Wang, Z.; Gu, J.; Zhao, B.; Wang, D.; et al. 2025b. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*.

Qu, D.; Yan, C.; Wang, D.; Yin, J.; Chen, Q.; Xu, D.; Zhang, Y.; Zhao, B.; and Li, X. 2024. Implicit event-rgbd neural slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 19584–19594.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PmLR.

Rang, M.; Bi, Z.; Liu, C.; Tang, Y.; Han, K.; and Wang, Y. 2025. Eve: Efficient Multimodal Vision Language Models with Elastic Visual Experts. *arXiv preprint arXiv:2501.04322*.

Singh, A.; Natarajan, V.; Shah, M.; Jiang, Y.; Chen, X.; Batra, D.; Parikh, D.; and Rohrbach, M. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8317–8326.

Tang, Y.; Guo, Z.; Wang, Z.; Zhang, R.; Chen, Q.; Liu, J.; Qu, D.; Wang, Z.; Wang, D.; Li, X.; et al. 2025. Exploring the potential of encoder-free architectures in 3d lmms. *arXiv preprint arXiv:2502.09620*.

Tang, Y.; Zhang, R.; Liu, J.; Guo, Z.; Zhao, B.; Wang, Z.; Gao, P.; Li, H.; Wang, D.; and Li, X. 2024. Any2point: Empowering any-modality large models for efficient 3d understanding. In *European Conference on Computer Vision*, 456–473. Springer.

Team, G.; Georgiev, P.; Lei, V. I.; Burnell, R.; Bai, L.; Gulati, A.; Tanzer, G.; Vincent, D.; Pan, Z.; Wang, S.; et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Team, K.; Du, A.; Yin, B.; Xing, B.; Qu, B.; Wang, B.; Chen, C.; Zhang, C.; Du, C.; Wei, C.; et al. 2025. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*.

Wang, G.; Cheng, S.; Zhan, X.; Li, X.; Song, S.; and Liu, Y. 2023a. Openchat: Advancing open-source language models with mixed-quality data. *arXiv preprint arXiv:2309.11235*.

Wang, J.; Meng, L.; Weng, Z.; He, B.; Wu, Z.; and Jiang, Y.-G. 2023b. To see is to believe: Prompting gpt-4v for better visual instruction tuning. *arXiv preprint arXiv:2311.07574*.

Wang, W.; Gao, Z.; Gu, L.; Pu, H.; Cui, L.; Wei, X.; Liu, Z.; Jing, L.; Ye, S.; Shao, J.; et al. 2025. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; Dong, G.; Wei, H.; Lin, H.; Tang, J.; Wang, J.; Yang, J.; Tu, J.; Zhang, J.; Ma, J.; Yang, J.; Xu, J.; Zhou, J.; Bai, J.; He, J.; Lin, J.; Dang, K.; Lu, K.; Chen, K.; Yang, K.; Li, M.; Xue, M.; Ni, N.; Zhang, P.; Wang, P.; Peng, R.; Men, R.; Gao, R.; Lin, R.; Wang, S.; Bai, S.; Tan, S.; Zhu, T.; Li, T.; Liu, T.; Ge, W.; Deng, X.; Zhou, X.; Ren, X.; Zhang, X.; Wei, X.; Ren, X.; Liu, X.; Fan, Y.; Yao, Y.; Zhang, Y.; Wan, Y.; Chu, Y.; Liu, Y.; Cui, Z.; Zhang, Z.; Guo, Z.; and Fan, Z. 2024a. Qwen2 Technical Report. arXiv:2407.10671.

Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; et al. 2024b. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Zhai, M.; He, J.; Ma, Z.; Zong, Z.; Zhang, R.; and Zhai, J. 2023a. {SmartMoE}: Efficiently training {Sparsely-Activated} models through combining offline and online parallelization. In *2023 USENIX Annual Technical Conference (USENIX ATC 23)*, 961–975.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023b. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, 11975–11986.

Zhang, W.; Lin, T.; Liu, J.; Shu, F.; Li, H.; Zhang, L.; Wanggui, H.; Zhou, H.; Lv, Z.; Jiang, H.; et al. 2024. Hyperllava: Dynamic visual and language expert tuning for multimodal large language models. *arXiv preprint arXiv:2403.13447*.

Zhao, B.; Wu, B.; He, M.; and Huang, T. 2023. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*.

Zhong, S.; Liang, L.; Wang, Y.; Wang, R.; Huang, R.; and Li, M. 2024. Adapmoe: Adaptive sensitivity-based expert gating and management for efficient moe inference. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, 1–9.

Zong, Z.; Ma, B.; Shen, D.; Song, G.; Shao, H.; Jiang, D.; Li, H.; and Liu, Y. 2024. Mova: Adapting mixture of vision experts to multimodal context. *arXiv preprint arXiv:2404.13046*.