
MokA: Multimodal Low-Rank Adaptation for MLLMs

Yake Wei^{1,2,3}, Yu Miao^{1,2,3}, Dongzhan Zhou⁴, Di Hu^{1,2,3*}

¹Gaoling School of Artificial Intelligence, Renmin University of China

²Beijing Key Laboratory of Research on Large Models and Intelligent Governance

³Engineering Research Center of Next-Generation Intelligent Search and Recommendation, MOE

⁴Shanghai Artificial Intelligence Laboratory

yakewei@ruc.edu.cn, ymiao@ruc.edu.cn

zhoudongzhan@pjlab.org.cn, dihu@ruc.edu.cn

Abstract

In this paper, we reveal that most current efficient multimodal fine-tuning methods are hindered by a key limitation: they are directly borrowed from LLMs, often neglecting the intrinsic differences of multimodal scenarios and even affecting the full utilization of all modalities. Inspired by our empirical observation, we argue that unimodal adaptation and cross-modal adaptation are two essential parts for the effective fine-tuning of MLLMs. From this perspective, we propose *Multimodal low-rank Adaptation (MokA)*, a multimodal-aware efficient fine-tuning strategy that takes multimodal characteristics into consideration. It compresses unimodal information by modality-specific parameters while explicitly enhancing cross-modal interaction, ensuring both unimodal and cross-modal adaptation. Extensive experiments cover three representative multimodal scenarios (audio-visual-text, visual-text, and speech-text), and multiple LLM backbones (LLaMA2/3, Qwen2, Qwen2.5-VL, etc). Consistent improvements indicate the efficacy and versatility of the proposed method. Ablation studies and efficiency evaluation are also conducted to fully assess our method. Overall, we think MokA provides a more targeted solution for efficient adaptation of MLLMs, paving the way for further exploration. The project page is at <https://gewu-lab.github.io/MokA>.

1 Introduction

Large language models (LLMs) have gained remarkable popularity due to their impressive ability to understand and generate content. To extend their capabilities to more general multimodal scenarios, recent advancements of Multimodal Large Language Models (MLLMs) [38, 40, 20] have focused on aligning other modalities, such as images, with text tokens, thereby equipping LLMs with the ability to interpret and process content of other modalities. However, due to the massive parameter scale of LLMs, fully fine-tuning such models on downstream tasks is computationally prohibitive and inefficient in most cases.

A promising direction has emerged in the field of LLM fine-tuning before, which involves selectively updating a subset of parameters rather than the full model. These Parameter-Efficient Fine-Tuning (PEFT) strategies have seen widespread adoption and have been successfully extended to the fine-tuning of MLLMs. In particular, LoRA [12] and its variants, which assume that over-parameterized models in fact reside on a low intrinsic dimension, have been broadly applied [6, 7, 39], demonstrating strong adaptability and efficiency. However, the development of efficient multimodal LLM fine-tuning is at present obscured by a “dark cloud”: *most current methods are directly borrowed from LLMs, often overlooking the fundamental differences of multimodal scenarios*. Indeed, prior studies

*Corresponding Author

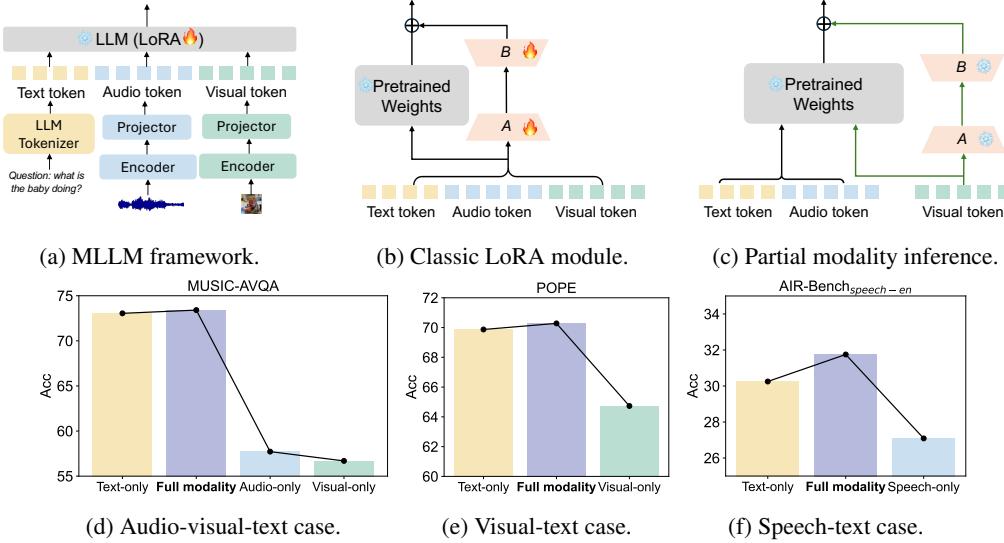


Figure 1: (a): Common MLLM fine-tuning framework. (b): Sketch of classic LoRA module for MLLM fine-tuning. (c): Partial modality inference setting. Take tokens of visual modality as an example. (d-f): Partial modality inference performance of LoRA. *Full modality*: Regular case where all multimodal tokens are processed by the LoRA module. *Text-/Audio-/Visual-/Speech-only*: Only text/audio/visual/speech tokens are passed through the LoRA pathway at the prefilling stage during inference. Results are based on LLaMA2.

in multimodal learning have demonstrated that the inherent heterogeneity of different modalities necessitates modality-specific utilization strategies, rather than a fully unified way [24, 34].

To this end, we are motivated to observe the fine-tuning efficacy of the widely used LoRA strategy. A common MLLM fine-tuning framework is shown in Figure 1a: encoded representation of non-text modality (e.g., audio or visual) is first aligned with the text embedding space via a projector (usually Q-former or MLP), after which the resulting multimodal tokens are integrated and processed jointly by the LLM. In the efficient fine-tuning case, the LLM backbone is frozen, and parameters of additional LoRA modules are optimized. Figure 1b provides the sketch of classic LoRA module. A and B matrices are shared across different modalities.

To further observe how well tokens of different modalities are utilized, we conduct *partial modality inference* experiments. The training stage retains the original setting, wherein all multimodal tokens are processed by the LoRA module. Specifically, we evaluate the model’s performance when only tokens from a selected modality are passed through the LoRA adaptation pathway at the prefilling stage during inference. As illustrated in Figure 1c, visual token inference is used as an example. And it should be noted that the pre-trained weights still receive full tokens of all modalities. Results shown in Figure 1d-1f demonstrate a surprising phenomenon across three representative multimodal scenarios, audio-visual-text case, visual-text case, and speech-text case. Text token inference can achieve quite comparable performance to the regular full modalities case. However, Non-text token inference (e.g., audio or visual) leads to a noticeable drop in performance.

The above results suggest that the optimization of all-modality-shared LoRA parameters is overly influenced by text tokens, resulting in non-text tokens being less effectively utilized during fine-tuning. Although these all-modality-shared parameters implicitly improve cross-modal interaction, this phenomenon reveals the need to consider individual modality during fine-tuning. *This fact inspires us that unimodal and cross-modal adaptation are equally critical in the fine-tuning of MLLMs*, which is mostly ignored as mentioned above.

To this end, we propose the **Multimodal low-rank Adaptation (MokA)**, a fine-tuning strategy designed to achieve unimodal adaptation while explicitly enhancing cross-modal interaction. While MokA retains the widely adopted low-rank decomposition matrices, it redefines the roles of matrices A and B to better accommodate multimodal characteristics. Specifically, matrix A is designed to be modality-specific, allowing each modality to compress information independently and thus avoid

interference from others. After that, a cross-attention mechanism is introduced to strengthen the interaction between text tokens and non-text tokens, emphasizing task-relevant features. Finally, a shared multimodal matrix B projects the unimodal low-rank representations into a unified space, facilitating effective alignment across modalities. These three parts jointly ensure both unimodal and cross-modal adaptation. In experiments, noticeable improvement in multiple multimodal scenarios demonstrates the effectiveness of our method. We think MokA represents a first-step attempt at multimodal-aware adaptation, and further possibilities exist under our basis that simultaneously accounts for both unimodal and cross-modal adaptation.

2 Method

2.1 Rethinking of low-rank adaptation in the multimodal scenario

LoRA [12] is based on the assumption that the weight updates during fine-tuning lie in a subspace of low “intrinsic rank.” Rather than updating the entire pre-trained weight matrix directly, LoRA introduces a low-rank decomposition approach, where the update $\Delta W \in \mathbb{R}^{d \times k}$ to a pre-trained matrix $W_0 \in \mathbb{R}^{d \times k}$ is parameterized as the product of two much smaller matrices: $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$, with $r \ll \min(d, k)$. The resulting fine-tuned weight matrix W' is given by $W_0 + \Delta W = W_0 + BA$. Therefore, for $h = W_0 \mathbf{x}$, the modified update forward pass yields:

$$h = W_0 \mathbf{x} + \Delta W \mathbf{x} = W_0 \mathbf{x} + BA \mathbf{x}. \quad (1)$$

Here, W_0 remains fixed during training, while only the matrices A and B are learned. To ensure stable training, A is initialized using a uniform Kaiming distribution [11], and B is initialized to zero, leading to an initial update $\Delta W = BA = 0$ at the beginning of fine-tuning. LoRA [12] and its variants have been extensively employed in the parameter-efficient fine-tuning of MLLMs [6, 7, 39]. These methods typically employ shared parameters to uniformly process tokens from all modalities, implicitly facilitating cross-modal interactions during adaptation. However, our empirical results reveal that such shared tuning leads to limited utilization of all modalities. This highlights the need to consider individual modality during fine-tuning.

To better support multimodal adaptation, we argue that both unimodal and cross-modal updates should be considered during fine-tuning. In other words, the model should be able to learn from each modality independently while also ensuring the cross-modal interaction. Therefore, the design of the update mechanism should ensure that both types of information are properly captured during the forward pass:

$$h = W_0 \mathbf{x} + \Delta W \mathbf{x} = W_0 \mathbf{x} + \Delta W[\mathbf{x}^{m_1}; \mathbf{x}^{m_2}; \dots; \mathbf{x}^{m_n}], \quad (2)$$

$$= W_0 \mathbf{x} + \underbrace{[\Delta W_1 \mathbf{x}^{m_1}; \Delta W_2 \mathbf{x}^{m_2}; \dots; \Delta W_n \mathbf{x}^{m_n}]}_{\text{unimodal adaptation}} + \underbrace{\Delta W_{\text{cross}} [\mathbf{x}^{m_1}; \mathbf{x}^{m_2}; \dots; \mathbf{x}^{m_n}]}_{\text{cross-modal adaptation}}, \quad (3)$$

where n is the number of modalities. \mathbf{x}^{m_i} is the token sequence of modality i . ΔW_i is the unimodal update parameters of modality i , and ΔW_{cross} is cross-modal update parameters.

2.2 Multimodal low-rank Adaptation (MokA)

Based on the above perspective, we propose **Multimodal low-rank Adaptation (MokA)** strategy, a parameter-efficient fine-tuning method tailored for the multimodal nature of MLLMs. Considering the efficiency advantage of LoRA, MokA retains the core idea of low-rank adaptation, but redefines the roles of the projection matrices A and B to better reflect the characteristics of multimodal scenarios. By unimodal compression and explicitly reinforcing cross-modal interaction, MokA enables both unimodal and cross-modal adaptation, leading to more effective fine-tuning of MLLMs. The overall structure of MokA is depicted in Figure 2.

Concretely, MokA has three core parts: unimodal matrix A , task-centric cross-attention, and shared multimodal matrix B . Here we take the audio-visual-text case as an example, and other cases can be well extended.

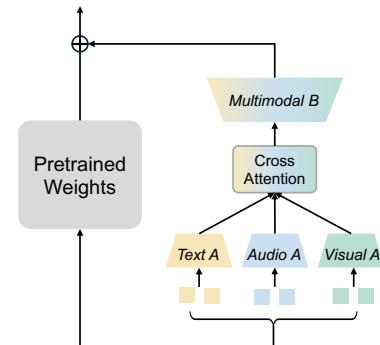


Figure 2: Illustration of MokA strategy.

2.2.1 Unimodal matrix A

For an arbitrary pretrained weight W_0 in the LLMs, we suppose its input sequence is $\mathbf{x} = [x_1^a; x_2^a; \dots; x_{N_a}^a; x_1^v; x_2^v; \dots; x_{N_v}^v; x_1^t; x_2^t; \dots; x_{N_t}^t]$. Here $\{N_i\}_{i \in \{a, v, t\}}$ is the token length of modality i . x_1^a is the first token of modality a , and so on. For simplicity, we use \mathbf{x}^i to denote the token sequence of modality i . Then the whole input sequence can be rewritten as $\mathbf{x} = [\mathbf{x}^a; \mathbf{x}^v; \mathbf{x}^t]$.

To ensure the well compression of unique unimodal information and avoid the interruption from others, matrix A is designed to be individual for each modality, allowing tokens from different modalities to be processed independently through their respective parameter. The compressed sequence after matrix A is:

$$A\mathbf{x} = [A^a\mathbf{x}^a; A^v\mathbf{x}^v; A^t\mathbf{x}^t], \quad (4)$$

where $\{A_i\}_{i \in \{a, v, t\}}$ is the parameter of modality i . After processing by unimodal matrix A , embeddings of each modality are individually mapped into a low-rank space, without the potential influence of other modalities.

2.2.2 Task-centric cross-attention

In the fine-tuning process of MLLMs, text and non-text tokens typically serve distinct roles. Specifically, under supervised instruction tuning, text tokens often function as task descriptions or prompts, whereas non-text tokens (e.g., audio or visual inputs) primarily convey contextual information upon which the task is based. The following example illustrates a typical instruction format:

<audio> <visual> Please answer the question: which clarinet makes the sound first?

In this case, <audio> and <visual> provide the event information. “Please answer the question: which clarinet makes the sound first?” describes the concrete task for LLMs. Successfully answering such questions relies on effectively capturing the semantic association between the task description conveyed by text tokens and the event cues provided by non-text tokens. Therefore, it becomes intuitive and necessary to explicitly emphasize the most relevant cross-modal information to support accurate reasoning. Since unimodal information has been extracted individually after the processing of unimodal matrices A , this stage is well-suited for introducing cross-modal interaction. Additionally, as the token embeddings are projected into a low-rank space, the computational burden of performing cross-modal interaction is significantly reduced. Hence, we place the cross-attention part after the low-rank compression to ensure both effectiveness and efficiency. The concrete attention mechanism is illustrated in Figure 3, and is conducted as follows:

$$\text{Att}(A^a\mathbf{x}^a, A^t\mathbf{x}^t, A^t\mathbf{x}^t) = \text{softmax}\left(\frac{(A^a\mathbf{x}^a)(A^t\mathbf{x}^t)^\top}{\sqrt{r}}\right) A^t\mathbf{x}^t, \quad (5)$$

$$\text{Att}(A^v\mathbf{x}^v, A^t\mathbf{x}^t, A^t\mathbf{x}^t) = \text{softmax}\left(\frac{(A^v\mathbf{x}^v)(A^t\mathbf{x}^t)^\top}{\sqrt{r}}\right) A^t\mathbf{x}^t, \quad (6)$$

where r is the rank. Then, the enhanced audio and visual tokens are:

$$A^a\mathbf{x}^a + \lambda_a \text{Att}(A^a\mathbf{x}^a, A^t\mathbf{x}^t, A^t\mathbf{x}^t), \quad (7)$$

$$A^v\mathbf{x}^v + \lambda_v \text{Att}(A^v\mathbf{x}^v, A^t\mathbf{x}^t, A^t\mathbf{x}^t), \quad (8)$$

where λ_a and λ_v are the hyperparameters that control the strength of explicit cross-modal interaction. Finally, the sequence after cross-attention is:

$$A\mathbf{x} = [A^a\mathbf{x}^a + \lambda_a \text{Att}(A^a\mathbf{x}^a, A^t\mathbf{x}^t, A^t\mathbf{x}^t); A^v\mathbf{x}^v + \lambda_v \text{Att}(A^v\mathbf{x}^v, A^t\mathbf{x}^t, A^t\mathbf{x}^t); A^t\mathbf{x}^t]. \quad (9)$$

Here we use $\text{Att}_{i,t,t}$ to simply denote the cross-attention between modality i and text. It should be noted that while we adopt a cross-attention module to explicitly enhance the interaction between text and non-text tokens, alternative designs that serve a similar purpose can also be considered. Further discussion is provided in Section 4.4. In addition, in MokA, linear projections (W_q , W_k , and W_v) are not included in the cross-attention module, since low-rank matrices A of each modality actually can be considered as the linear projection in attention in this case. We also provide more discussion and comparison in Appendix B.

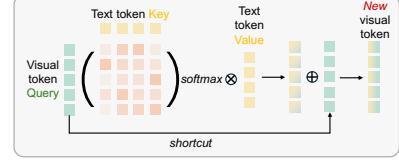


Figure 3: Cross-attention part of MokA. Take the visual token as an example.

2.2.3 Shared multimodal matrix B

After unimodal compression and explicit cross-modal interaction enhancement, it becomes crucial to project the resulting unimodal representations into a shared space to facilitate cross-modal alignment. To this end, a shared multimodal matrix B is employed to perform this projection. The final output of the MokA pathway is thus given by:

$$BA\mathbf{x} = [B(A^a\mathbf{x}^a + \lambda_a \text{Att}_{a,t,t}); B(A^v\mathbf{x}^v + \lambda_v \text{Att}_{v,t,t}); BA^t\mathbf{x}^t]. \quad (10)$$

2.2.4 Overview

In conclusion, in MokA, for a pretrained weight matrix $W_0 \in \mathbb{R}^{d \times k}$, its update $\Delta W \in \mathbb{R}^{d \times k}$ is parameterized as the product of much smaller matrices: $B \in \mathbb{R}^{d \times r}$ and $\{A^i \in \mathbb{R}^{r \times k}\}_{i \in \{a,v,t\}}$, with $r \ll \min(d, k)$. For input sequence \mathbf{x} , the forward pass yields:

$$h = W_0\mathbf{x} + \Delta W\mathbf{x} = W_0\mathbf{x} + \Delta W[\mathbf{x}^a; \mathbf{x}^v; \mathbf{x}^t], \quad (11)$$

$$= W_0\mathbf{x} + [B(A^a\mathbf{x}^a + \lambda_a \text{Att}_{a,t,t}); B(A^v\mathbf{x}^v + \lambda_v \text{Att}_{v,t,t}); BA^t\mathbf{x}^t], \quad (12)$$

$$= W_0\mathbf{x} + \underbrace{[BA^a\mathbf{x}^a; BA^v\mathbf{x}^v; BA^t\mathbf{x}^t]}_{\text{unimodal adaptation}} + \underbrace{[\lambda_a B\text{Att}_{a,t,t}; \lambda_v B\text{Att}_{v,t,t}; \mathbf{0}_{N_t}]}_{\text{cross-modal adaptation}}, \quad (13)$$

where $\mathbf{0}_{N_t}$ denotes the zero vector of dimension N_t , since text-token remains unchanged after cross-attention. During fine-tuning, W_0 remains unchanged, with A^i and B being subject to optimization. Also, A^i is initialized using the uniform Kaiming distribution [11], while B is initialized to zero. It leads to an initial update $\Delta W = 0$ at the beginning of fine-tuning, to provide a smooth starting point.

Based on Equation 13, MokA ensures both unimodal and cross-modal adaptation, offering a more tailored solution for fine-tuning MLLMs.

3 Training and evaluation details

3.1 Implement details

Our framework follows the common MLLM framework as illustrated in Figure 1a, but with MokA strategy. Text input is processed by the corresponding LLM tokenizer, and non-text input is first encoded by its encoder, and then aligned with the text embedding space via a projector. Here we use Q-former followed by a two-layer MLP as the projector. Finally, all tokens are fed into LLM.

For the visual branch of audio-visual-text and visual-text scenarios, we use CLIP-ViT/L-14 [25] as the visual encoder to extract the last layer patch level embedding of each frame or image. For the audio branch of the audio-visual-text scenario, we use the BEATs [5] encoder to extract features. For the speech branch of the speech-text scenario, OpenAI’s Whisper model [26] is used. The number of query tokens in Q-Former of all branches is 32.

3.2 Training procedure and benchmarks

Our experiment of MLLM follows the widely used two-stage training paradigm: pre-training stage that aims to cross-modal alignment and supervised instruction-tuning for downstream tasks.

Pre-training: LLM backbone is frozen. Projectors are trainable for cross-modal alignment. For the visual branch of audio-visual-text and visual-text scenarios, trainable modules are trained on video-LLaVA [19] dataset, including the video captioning and the image captioning tasks. For the audio branch of the audio-visual-text scenario, trainable modules are trained on AudioCaps [14] dataset on the audio captioning task. For the speech branch of the speech-text scenario, trainable modules are trained on GigaSpeech-M [4] dataset on the speech recognition task. During pre-training, using the AdamW optimizer with a cosine learning rate schedule. The initial learning rate is $1e-4$ with a warmup ratio of 0.03.

Instruction-tuning: At this stage, we train the model on downstream tasks in different scenarios. Trainable parameters include all projectors and our MokA module. For the audio-visual-text case, the model is fine-tuned on the train set of MUSIC-AVQA [17], and AVE [30], respectively. For the visual-text case, the model is fine-tuned on the LLaVA-Instruct-150K [20] and a 12k subset of

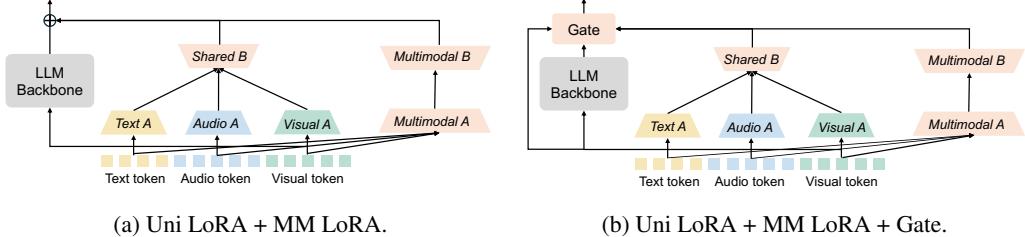


Figure 4: Framework of baselines that also follow our multimodal-aware basis, yet relying on more parameters and offering limited cross-modal interaction.

A-OKVQA. For the speech-text case, the model is fine-tuned on the LibriSpeech [23], using the annotations provided by [28]. Rank of low-rank matrices is 4. The remaining settings are the same as the first stage.

Inference: To well assess the effectiveness of our fine-tuning strategy, we evaluate our trained models on in-domain test sets or public benchmarks. Details are provided in the supplementary materials.

- **Audio-visual-text:** in-domain test set of MUSIC-AVQA and AVE dataset.
- **Visual-text:** public benchmarks: MME_{percep} [9], MMBench [22], POPE [18], SEED-Bench [16].
- **Speech-text:** public benchmarks: MMAU_{mini-speech} [27] as well as the foundation subset of AIR-Bench_{speech-en} [37].

Large Language Model. For all three cases, LLaMA-2-7b-Chat [31], LLaMA-3-8B-Instruct [10], and Qwen2-7B-Instruct [36] are used as the LLM base model, respectively. For the audio-visual-text case, Qwen2.5-VL-7B-Instruct [2] is also used as the LLM base model. Throughout the training process, weights of LLM are kept frozen. More experiments of Qwen3 are provided in Appendix A.

4 Experiments

4.1 Audio-visual-text scenario

To validate the effectiveness of our MokA fine-tuning strategy, we compare it with *LoRA* [12] and its variants, including *multiple LoRA*, *LoRAMoE* [8], *DoRA* [21], *HydraLoRA* [29], *Uni-modal LoRA* [1]. In addition, we also compare with two additional baselines, whose frameworks are provided in Figure 4. Concretely, the *Uni LoRA + MM LoRA* strategy employs unimodal low-rank matrices A to extract unimodal information independently, while incorporating an additional fully shared multimodal LoRA module to implicitly promote cross-modal interaction. The *Uni LoRA + MM LoRA + Gate* variant further introduces a gating mechanism to dynamically integrate the outputs of the Uni LoRA and MM LoRA branches for improved fusion. These two baselines incorporate our multimodal-aware basis that ensures both unimodal and cross-modal adaptation, but involve more parameters and offer limited cross-modal interaction. Based on the results in Table 1, we can have the following observations:

Our proposed MokA method achieves the superior overall performance across multiple audio-visual-text datasets, consistently outperforming other baselines and compared methods. While MokA introduces a slight increase in parameter scale compared to standard LoRA, this does not account for the observed performance improvements. Based on the Table 1, multiple LoRA, a baseline that uses 3 A matrices and B A matrices, underperforms both standard LoRA and MokA. Simply increasing the number of low-rank matrices does not necessarily lead to better fine-tuning performance. This suggests that MokA’s advantage stems not from parameter quantity, but from its insurance for both unimodal and multimodal adaptation.

The mentioned two baselines, *Uni LoRA + MM LoRA* and *Uni LoRA + MM LoRA + Gate*, achieve competitive results. These results further support the validity of our multimodal-aware basis that unimodal and cross-modal adaptation are both essential for the fine-tuning of MLLMs. However, despite their effectiveness, MokA achieves superior results with fewer parameters and further enhanced cross-modal interactions.

In addition, Qwen2.5VL with LoRA outperforms both LLaMA2 and Qwen2 under the LoRA fine-tuning setting. But when using MokA, the performance of Qwen2.5VL is slightly lower than that of LLaMA2 and Qwen2. A possible reason is that the official visual connector in Qwen2.5VL,

Table 1: Evaluation results of LoRA, its variants, and our MokA on audio-visual-text datasets, MUSIC-AVQA and AVE. #A and #B are the number of low-rank matrices. Here $N = 3$ refers to the number of modalities.

LLM	Method	MUSIC-AVQA	AVE	#A	#B
LLaMA2	LoRA [12]	73.41	69.84	1	1
	Multiple LoRA	72.66	71.77	N	N
	LoRAMoE [8]	73.57	72.81	N	N
	DoRA [21]	73.97	72.18	1	1
	HydraLoRA [29]	74.12	72.27	1	N
	Uni-modal LoRA [1]	74.37	71.44	N	N
	Uni LoRA + MM LoRA	74.43	72.36	$N + 1$	2
	Uni LoRA + MM LoRA + Gate	74.94	73.56	$N + 1$	2
	☕ MokA	75.71	74.68	N	1
Qwen2	LoRA	72.83	72.13	1	1
	Multiple LoRA	72.71	72.11	N	N
	LoRAMoE [8]	73.48	72.83	N	N
	DoRA [21]	73.29	72.91	1	1
	HydraLoRA [29]	73.14	72.59	1	N
	Uni-modal LoRA [1]	73.62	73.14	N	N
	Uni LoRA + MM LoRA	74.09	73.35	$N + 1$	2
	Uni LoRA + MM LoRA + Gate	74.71	73.96	$N + 1$	2
	☕ MokA	75.26	74.48	N	1
Qwen2.5-VL	LoRA	73.00	71.38	1	1
	Multiple LoRA	73.13	71.27	N	N
	LoRAMoE [8]	73.28	71.91	N	N
	DoRA [21]	73.37	71.06	1	1
	HydraLoRA [29]	73.04	71.26	1	N
	Uni-modal LoRA [1]	73.46	72.11	N	N
	Uni LoRA + MM LoRA	73.75	72.27	$N + 1$	2
	Uni LoRA + MM LoRA + Gate	73.81	72.68	$N + 1$	2
	☕ MokA	74.87	73.14	N	1
LLaMA3	LoRA	78.31	76.91	1	1
	Multiple LoRA	78.63	77.02	N	N
	☕ MokA	79.15	77.81	N	1

which serves a similar role to the projector used in our other LLM variants, remains frozen during fine-tuning. As a result, only the newly introduced audio branch is trainable, which may have limited the full potential of our method. But MokA still introduces noticeable improvement in this case, compared to other methods. In summary, our method achieves considerable improvement across various LLM backbones, demonstrating its broad versatility.

4.2 Visual-text and speech-text scenarios

To further validate our method across a broader range of multimodal scenarios, we conducted experiments beyond the challenging audio-visual-text case. Specifically, our method is further verified on two representative multimodal scenarios: visual-text and speech-text. For these tasks, we adopted three different LLM backbones, LLaMA2, LLaMA3, and Qwen2. The corresponding results are presented in Table 2 and Table 3. The experimental results demonstrate that our method achieves stable and consistent performance gains across multiple benchmark datasets, further confirming its effectiveness. This indicates the versatility of MokA in handling different multimodal combinations and LLM architectures.

Table 2: Evaluation results of LoRA, its variants, and our MokA on visual-text benchmarks. #A and #B are the number of low-rank matrices. Here $N = 2$ refers to the number of modalities.

LLM	Method	MME _{percep}	MMBench	POPE	SEED-Bench	#A	#B
LLaMA2	LoRA	908.52	50.64	70.28	39.71	1	1
	Multiple LoRA	882.87	49.83	68.20	38.44	N	N
	LoRAMoE [8]	938.52	51.98	71.15	39.13	N	N
	DoRA [21]	786.47	51.31	71.07	38.96	1	1
	HydraLoRA [29]	774.47	47.33	70.87	38.81	1	N
	Uni-modal LoRA [1]	992.31	51.98	72.24	39.27	N	N
	Uni LoRA + MM LoRA	972.87	50.96	73.39	39.74	$N + 1$	2
	Uni LoRA + MM LoRA + Gate	988.37	52.01	73.48	39.91	$N + 1$	2
Qwen2	LoRA	1062.34	57.89	81.17	55.25	1	1
	Multiple LoRA	1103.28	57.01	80.96	55.13	N	N
	LoRAMoE [8]	1157.39	57.29	81.29	56.39	N	N
	DoRA [21]	1024.42	56.19	80.75	55.03	1	1
	HydraLoRA [29]	1098.25	56.42	81.34	54.67	1	N
	Uni-modal LoRA [1]	1189.47	57.39	81.12	56.21	N	N
	Uni LoRA + MM LoRA	1191.81	57.17	81.46	56.84	$N + 1$	2
	Uni LoRA + MM LoRA + Gate	1201.49	57.91	81.72	57.18	$N + 1$	2
LLaMA3	LoRA	1030.64	68.45	77.47	56.34	1	1
	Multiple LoRA	1032.74	68.79	78.73	56.06	N	N
	MokA	1072.67	69.90	79.27	56.60	N	1

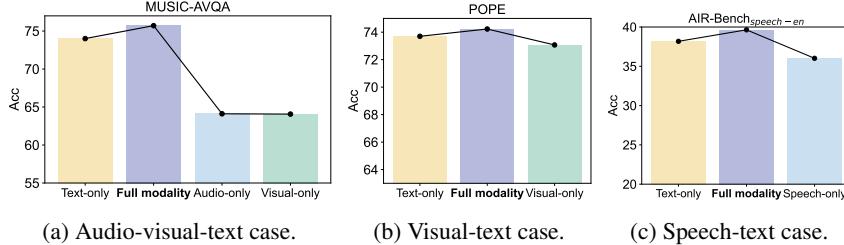


Figure 5: **Partial modality inference performance of MokA w/o cross-attention.** *Full modality*: Regular case where all multimodal tokens are processed by the MokA module w/o cross-attention. *Text-/Audio-/Visual-/Speech-only*: Only text/audio/visual/speech tokens are passed through the LoRA pathway at the first generation step during inference. Results are based on LLaMA2.

4.3 Partial modality inference of MokA

To further examine how effectively MokA leverages tokens from different modalities, we also conduct partial modality inference experiments where only tokens from a selected modality are passed through the LoRA adaptation pathway at the first generation during inference. It should be noted that the evaluated model is *MokA w/o cross-attention*, as cross-attention computation requires the presence of both text and non-text tokens. The results, presented in Figure 5, show that MokA w/o cross-attention significantly enhances the utilization of individual modalities compared to LoRA (as shown in Figure 1d-1f). These findings highlight that the multimodal-aware design of MokA facilitates more effective use of all available modalities.

4.4 Cross-modal interaction variants

In the original MokA framework, cross-attention is employed to explicitly strengthen the interaction between text and non-text tokens, thereby facilitating improved cross-modal adaptation. As previously discussed, alternative modules that similarly enhance this interaction can also be considered. In this

Table 3: Evaluation results of LoRA, its variants, and our MokA on speech-text benchmarks, MMAU_{mini-speech} and AIR-Bench_{speech-en}. #A and #B are the number of low-rank matrices. Here $N = 2$ refers to the number of modalities.

LLM	Method	MMAU	AIR-Bench	#A	#B
LLaMA2	LoRA	30.33	31.75	1	1
	Multiple LoRA	29.73	31.91	N	N
	LoRAMoE [8]	27.63	33.97	N	N
	DoRA [21]	26.73	34.36	1	1
	HydraLoRA [29]	29.13	31.66	1	N
	Uni-modal LoRA [1]	38.14	35.14	N	N
	Uni LoRA + MM LoRA	37.54	32.56	$N + 1$	2
	Uni LoRA + MM LoRA + Gate	32.13	33.04	$N + 1$	2
	_DIGEST MokA	38.44	39.64	N	1
Qwen2	LoRA	50.15	44.55	1	1
	Multiple LoRA	50.45	41.13	N	N
	LoRAMoE [8]	50.75	42.99	N	N
	DoRA [21]	52.55	44.11	1	1
	HydraLoRA [29]	53.45	43.94	1	N
	Uni-modal LoRA [1]	54.05	47.01	N	N
	Uni LoRA + MM LoRA	54.16	43.55	$N + 1$	2
	Uni LoRA + MM LoRA + Gate	54.35	46.15	$N + 1$	2
	_DIGEST MokA	55.26	49.17	N	1
LLaMA3	LoRA	46.25	43.04	1	1
	Multiple LoRA	44.74	43.87	N	N
	_DIGEST MokA	51.05	44.39	N	1

Table 4: Evaluation results of MokA and variants on audio-visual-text and visual-text cases. Results are based on LLaMA2.

Method	Music-AVQA	AVE	MME _{percep}	MMBench	POPE	SEED-Bench
LoRA	73.41	69.84	908.52	50.64	70.28	39.71
Multiple LoRA	72.66	71.77	882.87	49.83	68.20	38.44
Cross-attention*	74.94	72.59	955.18	51.25	72.94	39.91
Naive interaction	75.04	73.18	996.73	51.49	73.52	40.17
_DIGEST MokA	75.71	74.68	1025.86	52.74	74.23	40.45

section, we explore several variants of the cross-modal interaction module, as summarized in Table 4. The *cross-attention** variant also adopts a cross-attention mechanism; however, it uses text tokens as queries. Consequently, the updated text tokens integrate information from the relevant non-text tokens—reversing the direction of interaction compared to the original MokA. The *naive interaction* variant performs a simple, uniform mapping from text tokens to non-text tokens without employing any attention mechanism.

Experimental results show that all proposed variants outperform the LoRA baseline, demonstrating the general effectiveness of explicitly enhancing cross-modal interactions. However, the *cross-attention** variant performs slightly worse than the others. One possible explanation is that, unlike in other variants where text tokens remain unchanged, this variant alters text tokens by integrating non-text features. Although cross-modal interaction is enhanced, the modification of text representations may adversely affect language modeling capabilities. In addition, while *naive interaction* yields competitive results, MokA achieves further improvement through its dynamic attention mechanism. These findings suggest that the core idea of explicitly reinforcing cross-modal interactions is beneficial, and the effectiveness is not restricted to one specific module design.

4.5 Ablation study

To thoroughly validate the efficacy of our method, we conduct ablation studies across all three multimodal scenarios. Results are shown in Table 5. Based on the results, even without the cross-attention module, MokA w/o CA outperforms the LoRA baseline, demonstrating the effectiveness of enhancing unimodal adaptation. Furthermore, the introduction of the cross-attention module leads to additional performance improvements, indicating the benefit of explicitly enhancing cross-modal adaptation. These results indicate the necessity of each part in MokA.

4.6 Efficiency evaluation

To enable a more comprehensive comparison, we further evaluate the proposed MokA and LoRA baselines on the proportion of trainable parameters in the full model, and inference latency. As reported in Table 6, although MokA introduces additional parameters due to the inclusion of more low-rank matrices, the increase is quite modest compared to the full LLM. Also, despite MokA incurs a slight increase in inference latency compared to standard LoRA, it achieves a notable performance gain of 3.95% on the POPE benchmark. These results suggest that the additional computational cost introduced by MokA is acceptable, and the performance improvement is considerable. More detailed efficiency evaluations are provided in Appendix D.

5 Related works

MLLMs built upon powerful LLM backbones are increasingly demonstrating impressive capabilities across diverse downstream tasks [33, 13]. However, fine-tuning these models remains computationally expensive, prompting growing interest in parameter-efficient fine-tuning (PEFT) techniques that reduce memory and storage overhead during adaptation. Among them, LoRA has emerged as a widely adopted, and researchers have proposed several variants to further improve its efficiency and flexibility [8, 21, 29, 1]. For instance, LoRAMoE [8] introduces multiple LoRA heads combined via a gating mechanism, while DoRA [21] focuses solely on optimizing the gradient direction, enabling more efficient updates. Despite these advancements, most PEFT strategies for MLLMs are direct extensions of LLM techniques and fail to account for the inherent characteristics of multimodal learning. To address this gap, we propose MokA, a fine-tuning strategy specifically designed for MLLMs. It explicitly ensures both unimodal and cross-modal adaptation to better preserve unimodal representations and enhance cross-modal interaction, offering a targeted solution for efficient and effective multimodal adaptation.

6 Discussion

In this paper, we argue that both unimodal adaptation and cross-modal adaptation are essential parts for the effective fine-tuning of MLLMs, yet have largely been neglected before. To this end, we propose *Multimodal low-rank Adaptation (MokA)* for efficient multimodal fine-tuning. MokA redefines the roles of low-rank matrices A and B , ensuring unimodal information is preserved while enhancing cross-modal interaction by cross-attention. We think MokA is a preliminary step toward multimodal-aware adaptation, highlighting the potential for future extensions that jointly consider both unimodal and cross-modal adaptation.

Table 5: Ablation study of MokA. CA denotes Cross-Attention. Results are based on LLaMA2.

Method	MUSIC-AVQA	POPE	AIR-Bench
LoRA [12]	73.41	70.28	31.75
Multiple LoRA	72.66	68.20	31.97
MokA w/o CA	74.85	73.57	33.25
⌚ MokA	75.71	74.23	39.64

Table 6: Efficiency evaluation and performance comparison. Here, trainable parameters include low-rank matrices and all projectors. Results are based on LLaMA2.

Method	Trainable / Total Parameters	Avg. forward time per sample	POPE Acc
LoRA [12]	1.27%	1.000 ×	70.28
Multiple LoRA	1.43%	1.006 ×	68.20
⌚ MokA	1.33%	1.069 ×	74.23

Acknowledgement

The project was supported by the fund for building world-class universities (disciplines) of Renmin University of China, sponsored by CCF-Zhipu.AI Large Model Innovation Fund, and National Natural Science Foundation of China (NO.62106272). The project was also supported by the Shanghai Municipal Science and Technology Major Project, and Shanghai Artificial Intelligence Laboratory.

References

- [1] Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*, 2025.
- [2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.
- [3] Chi Chen, Yiyang Du, Zheng Fang, Ziyue Wang, Fuwen Luo, Peng Li, Ming Yan, Ji Zhang, Fei Huang, Maosong Sun, et al. Model composition for multimodal large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11246–11262, 2024.
- [4] Guoguo Chen, Shuzhou Chai, Guanbo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, et al. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. *arXiv preprint arXiv:2106.06909*, 2021.
- [5] Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, and Furu Wei. Beats: Audio pre-training with acoustic tokenizers. *arXiv preprint arXiv:2212.09058*, 2022.
- [6] Shaoxiang Chen, Zequn Jie, and Lin Ma. Llava-mole: Sparse mixture of lora experts for mitigating data conflicts in instruction finetuning mllms. *arXiv preprint arXiv:2401.16160*, 2024.
- [7] Zhehuai Chen, He Huang, Andrei Andrusenko, Oleksii Hrinchuk, Krishna C Puvvada, Jason Li, Subhankar Ghosh, Jagadeesh Balam, and Boris Ginsburg. Salm: Speech-augmented language model with in-context learning for speech recognition and translation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13521–13525. IEEE, 2024.
- [8] Shihan Dou, Enyu Zhou, Yan Liu, Songyang Gao, Jun Zhao, Wei Shen, Yuhao Zhou, Zhiheng Xi, Xiao Wang, Xiaoran Fan, et al. Loramoe: Alleviate world knowledge forgetting in large language models via moe-style plugin. *arXiv preprint arXiv:2312.09979*, 2023.
- [9] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. Mme: A comprehensive evaluation benchmark for multimodal large language models, 2024.
- [10] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.
- [12] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.
- [13] Yue Jiang, Yueming Lyu, Bo Peng, Wei Wang, and Jing Dong. Cmsl: Cross-modal style learning for few-shot image generation. *Machine Intelligence Research*, pages 1–17, 2025.
- [14] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 119–132, 2019.
- [15] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020.

[16] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension, 2023.

[17] Guangyao Li, Yake Wei, Yapeng Tian, Chenliang Xu, Ji-Rong Wen, and Di Hu. Learning to answer questions in dynamic audio-visual scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19108–19118, 2022.

[18] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

[19] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023.

[20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023.

[21] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*, 2024.

[22] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer, 2024.

[23] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[24] Xiaokang Peng, Yake Wei, Andong Deng, Dong Wang, and Di Hu. Balanced multimodal learning via on-the-fly gradient modulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8238–8247, 2022.

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[26] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR, 2023.

[27] S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. Mmau: A massive multi-task audio understanding and reasoning benchmark. *arXiv preprint arXiv:2410.19168*, 2024.

[28] Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. Salmonn: Towards generic hearing abilities for large language models. *arXiv preprint arXiv:2310.13289*, 2023.

[29] Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Cheng-Zhong Xu. Hydralora: An asymmetric lora architecture for efficient fine-tuning. *Advances in Neural Information Processing Systems*, 37:9565–9584, 2024.

[30] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *Proceedings of the European conference on computer vision (ECCV)*, pages 247–263, 2018.

[31] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[32] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[33] Xiao Wang, Guangyao Chen, Guangwu Qian, Pengcheng Gao, Xiao-Yong Wei, Yaowei Wang, Yonghong Tian, and Wen Gao. Large-scale multi-modal pre-trained models: A comprehensive survey. *Machine Intelligence Research*, 20(4):447–482, 2023.

[34] Yake Wei, Di Hu, Henghui Du, and Ji-Rong Wen. On-the-fly modulation for balanced multimodal learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[35] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025.

[36] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report, 2024.

[37] Qian Yang, Jin Xu, Wenrui Liu, Yunfei Chu, Ziyue Jiang, Xiaohuan Zhou, Yichong Leng, Yuanjun Lv, Zhou Zhao, Chang Zhou, et al. Air-bench: Benchmarking large audio-language models via generative comprehension. *arXiv preprint arXiv:2402.07729*, 2024.

[38] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023.

[39] Jeong Hun Yeo, Seunghee Han, Minsu Kim, and Yong Man Ro. Where visual speech meets language: Vsp-llm framework for efficient and context-aware visual speech processing. *arXiv preprint arXiv:2402.15151*, 2024.

[40] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

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: In this paper, we propose MokA, an efficient fine-tuning strategy for MLLMs. Experiments covering multiple multimodal scenarios verify the effectiveness of MokA.

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: In Section 4.6, we discuss the efficiency and efficacy of MokA and baselines. MokA is with a slight increase in parameters and inference latency, but also brings noticeable improvement.

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: The paper does not include theoretical results.

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 provide experimental information in Section 3.

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?

Answer: [\[Yes\]](#)

Justification: We provide source code in the supplementary material.

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, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: **[Yes]**

Justification: We provide the information of used dataset in Section 3.

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: **[No]**

Justification: The resource-intensive nature of LLM training prevents us from conducting multiple runs or including variance estimates.

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 provide this information in the supplementary material.

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: Yes, we have reviewed 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 provide this discussion in the supplementary material.

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.
- 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: **[No]**

Justification: Our model is used for multimodal understanding, which has low risk for misuse.

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: In Section 3, we have cited all used models and datasets.

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: **[Yes]**

Justification: We will release our source code.

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: This 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: [NA]

Justification: This 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.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: In this paper, we provide a new fine-tuning strategy for multimodal LLM. And in Section 3, we provide the training process.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Extension to Qwen3

To further validate our MokA method, we equip it with the latest Qwen3 [35] model. Qwen3-8B is used as the LLM base model. Throughout the training process, weights of LLM are kept frozen.

Results are shown in Table 7. We conduct experiments in the audio-visual-text case. Our method can still deliver improvements compared to the LoRA baseline, with the latest Qwen3 model. These findings provide additional evidence of the effectiveness and scalability of MokA.

Table 7: Evaluation results of LoRA, its variants, and our MokA on audio-visual-text datasets, MUSIC-AVQA and AVE. #A and #B are the number of low-rank matrices. Here $N = 3$ refers to the number of modalities.

Method	MUSIC-AVQA	AVE	#A	#B
LoRA	78.57	74.17	1	1
Multiple LoRA	78.24	74.01	N	N
⌚ MokA	79.54	75.38	N	1

B MokA with linear projection

In MokA, linear projections (W_q , W_k , and W_v) are not included in the cross-attention module:

$$\text{Att}(A^a \mathbf{x}^a, A^t \mathbf{x}^t, A^t \mathbf{x}^t) = \text{softmax} \left(\frac{(A^a \mathbf{x}^a)(A^t \mathbf{x}^t)^\top}{\sqrt{r}} \right) A^t \mathbf{x}^t, \quad (14)$$

$$\text{Att}(A^v \mathbf{x}^v, A^t \mathbf{x}^t, A^t \mathbf{x}^t) = \text{softmax} \left(\frac{(A^v \mathbf{x}^v)(A^t \mathbf{x}^t)^\top}{\sqrt{r}} \right) A^t \mathbf{x}^t. \quad (15)$$

The reason is that low-rank matrices A of each modality actually can be considered as the linear projection in attention in this case. Therefore, we do not introduce other linear projections in the cross-attention module. In addition, projections of key and value are shared in this case (A^t). In fact, this kind of projection sharing strategy has been widely used to increase the efficiency of attention [15, 32]. For example, in the Linformer [32], it also utilizes the sharing key-value projection strategy to reduce computation cost. What's more, here non-text tokens are used as queries to merge textual information into non-text modalities. In this way, cross-modal interaction is explicitly enhanced, while text tokens are kept unchanged to avoid potential disruption to the model's original strong text understanding capability.

To further validate the idea of cross-attention, we conduct experiments that include linear projections in cross-attention. The concrete attention mechanism with linear projection is conducted as follows, and the notation is consistent with the main manuscript:

$$\text{Att}(A^a \mathbf{x}^a, A^t \mathbf{x}^t, A^t \mathbf{x}^t) = \text{softmax} \left(\frac{(W_q^a A^a \mathbf{x}^a)(W_k^t A^t \mathbf{x}^t)^\top}{\sqrt{r}} \right) W_v^t A^t \mathbf{x}^t, \quad (16)$$

$$\text{Att}(A^v \mathbf{x}^v, A^t \mathbf{x}^t, A^t \mathbf{x}^t) = \text{softmax} \left(\frac{(W_q^v A^v \mathbf{x}^v)(W_k^t A^t \mathbf{x}^t)^\top}{\sqrt{r}} \right) W_v^t A^t \mathbf{x}^t. \quad (17)$$

Here, W_q^a is the linear projection of the audio query, and the others are similar.

In Table 11, we provide the results in the audio-visual-text and visual-text cases. Based on the results, MoKA with linear projection, yields improvements compared with LoRA baseline. However, it does not consistently outperform the original MoKA and introduces additional trainable parameters along with increased computational overhead.

C More than three modality case

In this section, we extend our experiments to scenarios with more than three modalities. Specifically, we consider a four-modality setting involving audio, visual, point cloud, and language data, and

evaluated it on the MCUB-3 benchmark [3]. In this 3+ modality case, LoRA has an accuracy of 37.41%. Our MokA has an accuracy of 45.58%, indicating its scalability and effectiveness under 3+ modality cases.

D Efficiency evaluation

Compared to LoRA, MokA introduces additional A matrices and a cross-attention module. However, it should be noted that the additional computational cost of MokA only comes from the cross-attention module, which is flexible and can be replaced by a more efficient strategy if needed. In this section, we provide a more detailed efficiency analysis comparing FLOPs, GPU memory usage, and average forward time per sample (proportional to training time). For clarity, we report these metrics for the two-modality (VL), three-modality (AVL), and four-modality (AVPL) settings. As shown in the tables, MokA’s extra computational and memory cost remains limited and acceptable for typical multimodal scenarios.

Table 8: Efficiency evaluation between LoRA and MokA on various datasets. Results are based on LLaMA2.

VL (MME_percep)	FLOPs	Memory Usage	Avg. forward time/sample
LoRA	1.000x	1.000x	1.000x
⌚ MokA	1.009x	1.001x	1.069x
AVL (MUSIC-AVQA)	FLOPs	Memory Usage	Avg. forward time/sample
LoRA	1.000x	1.000x	1.000x
⌚ MokA	1.021x	1.001x	1.134x
AVPL (MCUB-3)	FLOPs	Memory Usage	Avg. forward time/sample
LoRA	1.000x	1.000x	1.000x
⌚ MokA	1.013x	1.002x	1.213x

E Audio-visual interaction

In the original MokA, only the attention between text and non-text tokens is considered. It is motivated by the fact that text modality typically conveys the question or task description, while the audio and visual modalities provide environmental information, in the instruction. Therefore, cross-attention is applied to explicitly enhance the interaction between the task (text token) and environment (non-text token). It is also worth exploring whether further interactions between the scene modalities themselves—i.e., audio and visual—can be beneficial. To this end, we conduct additional ablation studies. Experiments are conducted based on LLaMA2. The table reports the results for audio-visual attention with both audio as query and video as query. The results show that introducing additional audio-visual attention can bring gains, but it is not very noticeable. The potential benefit from further enhancing explicit audio-visual interactions is relatively limited.

Table 9: Evaluation results of LoRA and MokA variants on audio-visual-text datasets. Results are based on LLaMA2.

Method	MUSIC-AVQA	AVE
LoRA	73.41	69.84
⌚ MokA	75.71	74.68
MokA w/ audio query att.	75.78	74.53
MokA w/video query att.	75.76	74.81

F Ablation study of rank

In this section, we conduct experiments of MokA with different ranks. It consistently outperforms the LoRA baseline across different rank settings.

Table 10: Evaluation results of LoRA and MokA on MUSIC-AVQA and AVE with different ranks. Results are based on LLaMA2.

Method	MUSIC-AVQA	AVE	Rank
LoRA	73.41	69.84	r=4
LoRA	73.56	70.01	r=8
LoRA	73.73	70.07	r=12
⌚ MokA	75.71	74.68	r=4
⌚ MokA	74.68	74.71	r=8
⌚ MokA	74.89	74.36	r=12

G Case study of cross-attention in MokA

In this section, we conduct a case study on the cross-attention module in MokA. This module explicitly integrates task description information from text tokens with non-text tokens, thereby facilitating cross-modal interaction. Here we provide two samples from an audio-visual-text scenario, as shown in Figure 6. The visualization shows the cross-attention weights of the q_{proj} at the 10-th layer. The LLM model is LLaMA2. The results indicate that during the process of explicit cross-modal integration (e.g., cross-attention), text tokens that are more related to a given modality can receive higher attention weights. For example, the token “sound” receives greater attention in relation to the audio modality. This cross-modal integration can better facilitate the alignment and interaction between text tokens and non-text tokens.

Table 11: Experiments of MokA with linear projection under the audio-visual-text and visual-text cases. The LLM backbone is LLaMA2.

Method	Music-AVQA	MME _{percep}	MMBench	POPE	SEED-Bench
LoRA	73.41	908.52	50.64	70.28	39.71
Multiple LoRA	72.66	882.87	49.83	68.20	38.44
MokA w/ linear projection	73.83	926.77	53.97	72.43	41.01
⌚ MokA	75.71	1025.86	52.74	74.23	40.45

H Broader impacts

In this paper, we aim to contribute to the efficient fine-tuning of MLLM, particularly how they well process and integrate information from different modalities. Improvements in this area may support downstream applications in fields like autonomous driving and education. At the same time, this line of research carries certain risks. For example, there is a possibility that MLLM could reflect or amplify biases present in the training data, or be misused in sensitive contexts. We do not directly address these issues in this work, but acknowledge them as important areas for future research. All datasets used are publicly available, and we follow standard filtering procedures to reduce exposure to harmful content.

I Datasets

Information about the datasets used in our experiments is provided in this section.

Video-LLaVA [19] used a mixed dataset of images and videos for video captioning and image captioning tasks. The dataset includes a 665k image-text instruction and a 100k video-text instruction. This dataset is used for pre-training the visual branch.

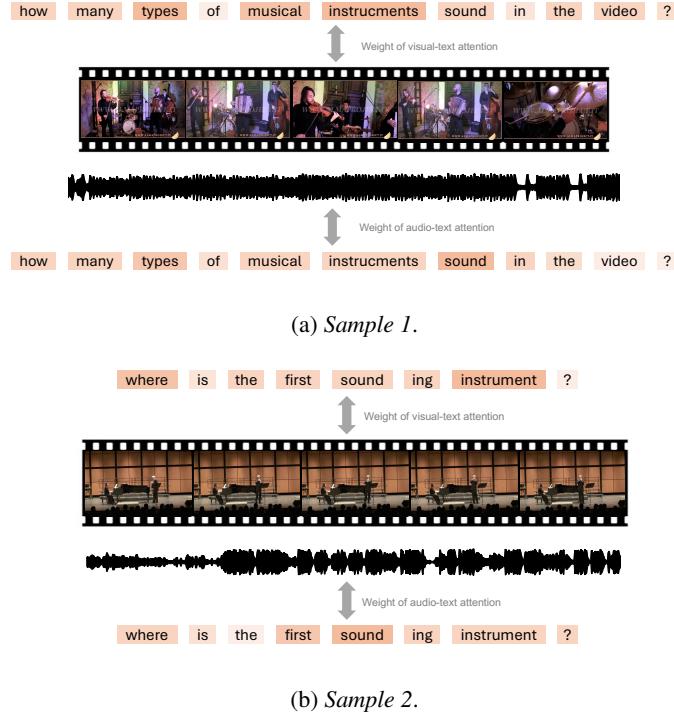


Figure 6: Cross-attention weight visualization, where deeper colors indicate higher attention weights.

AudioCaps [14] dataset is used for the audio captioning task. It consists of 46K pairs of audio clips and text descriptions. This dataset is used for pre-training the audio branch.

GigaSpeech-M [4] is a 1000h dataset for speech recognition task. This dataset is used for pre-training the speech branch.

MUSIC-AVQA [17] is an audio-visual-text dataset, which is introduced to support spatio-temporal understanding of musical content. It offers 45K QA pairs across 33 question templates that span multiple modalities and question types.

AVE [30] is an audio-visual-text dataset. It focuses on the audio-visual event localization task. This dataset covers 28 event classes and consists of 4,143 samples.

LLaVA-Instruct-150K [20] is a set of GPT-generated multimodal instruction-following data. It is used for instruction fine-tuning for the visual-text case.

LibriSpeech [23] is a 960-hour dataset. We use the instruction from [28] for instruction fine-tuning of the speech-text case.

MME_{percep} [9] is the perception subset of the MME benchmark, covering a total of 10 subtasks for the evaluation of the visual-text perception ability.

MMBench [22] is a collection of benchmarks to evaluate the visual-text understanding capability. It has 3,000 multiple-choice questions covering object detection, text recognition, action recognition, image captioning, relation reasoning, and so on.

POPE [18] is a benchmark that is used for evaluating the visual-text understanding ability of MLLM. The used image is the test set of MSCOCO dataset.

SEED-Bench [16] consists of 19K multiple-choice questions with accurate human annotations for evaluating the visual-text understanding ability of MLLM.

MMAU_{mini-speech} [27] is the speech subset of MMAU-mini benchmark. This benchmark is used for evaluating the speech-text understanding ability of MLLM.

AIR-Bench_{speech-en} [37] is the English speech subset of the foundation part of AIR-Bench. This benchmark is used for evaluating the speech-text understanding ability of MLLM.