On Domain-Adaptive Post-Training for Multimodal Large Language Models

Daixuan Cheng[α] **Shaohan Huang**[β] **Ziyu Zhu**[γ] **Xintong Zhang**[ε] **Wayne Xin Zhao**[ϑ]
Zhongzhi Luan[β] **Bo Dai**[α†] **Zhenliang Zhang**[α†]
[α]State Key Laboratory of General Artificial Intelligence, BIGAI [β]Beihang University
[γ]Tsinghua University [ε]Beijing Institute of Technology [ϑ]Renmin University of China
https://huggingface.co/AdaptLLM

Abstract

Adapting general multimodal large language models (MLLMs) to specific domains, such as scientific and industrial fields, is highly significant in promoting their practical applications. This paper systematically investigates *domain adaptation of MLLMs via post-training*, focusing on data synthesis, training pipeline, and task evaluation. (1) **Data Synthesis**: Using only open-source models, we develop a generate-then-filter pipeline that curates diverse visual instruction tasks based on domain-specific image-caption pairs. The resulting data surpass the data synthesized by manual rules or strong closed-source models in enhancing domain-specific performance. (2) **Training Pipeline**: Unlike general MLLMs that typically adopt a two-stage training paradigm, we find that a single-stage approach is more effective for domain adaptation. (3) **Task Evaluation**: We conduct extensive experiments in high-impact domains such as biomedicine, food, and remote sensing, by post-training a variety of MLLMs and then evaluating MLLM performance on various domain-specific tasks. Finally, we fully open-source our models, code, and data to encourage future research in this area.

1 Introduction

General multimodal large language models (MLLMs; Alayrac et al., 2022; Liu et al., 2024c) have shown impressive capabilities in general scenarios. However, their expertise plummets in specialized domains due to insufficient domain-specific training (Cheng et al., 2024b; Li et al., 2025; Lee et al., 2025). For instance, scientific fields require learning from specialized images not commonly found in general scenarios (Luo et al., 2023a); and industrial applications face privacy constraints that limit data access for general training (Bhatia et al., 2024).

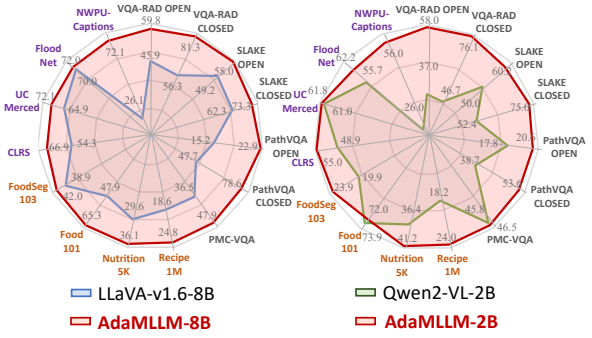


Figure 1: **Domain-Specific Task Performance of General MLLM and AdaMLLM.** For each domain, we conduct post-training to adapt the general MLLM and evaluate MLLM performance on various domain-specific tasks. Biomedicine, food and remote sensing tasks are colored gray, orange and purple, respectively.

Domain-specific training for MLLMs requires diverse visual instruction tasks infused with domain knowledge (Li et al., 2023a). For domain-specific instruction synthesis, while using closed- or open-source models to synthesize data is common in training general MLLMs, challenges arise when applying this approach to domain-specific MLLMs due to privacy concerns with closed-source models (OpenAI, 2023) and the lack of domain expertise in open-source models. For domain-specific training, the two-stage pipeline—first training on image-caption pairs, then on visual instruction tasks—is widely adopted for developing general MLLMs (Liu et al., 2024c). However, tasks in specialized domains are often limited, and splitting them into two stages can further reduce task diversity within each stage.

In this paper, we systematically investigate domain-specific instruction synthesis and training pipelines for MLLM post-training, referred to as *domain-adaptive training*. Specifically, we aim to (1) leverage only open-source models for data synthesis to avoid the privacy concerns of closed-source models and (2) maintain domain-specific task diversity throughout the post-training stage.

† Corresponding Author.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 274–296
November 4-9, 2025

For instruction synthesis with open-source models, we propose a generate-then-filter pipeline to address the lack of domain expertise. First, an open-source MLLM is fine-tuned to output instruction-response pairs based on input domain-specific image-caption pairs. To ensure diversity, we curate a seed data collection encompassing various domains and tasks based on existing datasets, without the need for additional expert annotation. The fine-tuned MLLM can effectively leverage domain knowledge in the image-caption source to generate diverse instruction-response pairs. Then, to ensure synthetic response accuracy, instead of directly verifying each response based on the instruction—which requires significant expertise—we propose selecting tasks with inherently consistent responses. This significantly improves accuracy while reducing the need for expert annotation. Although generated from open-source models, our synthetic tasks improve model performance more effectively than those generated by manual rules and strong closed-source models. **For training pipeline**, in the area of *domain-adaptive post-training* of MLLMs, two-stage training remains mainstream (Li et al., 2024a; Chen et al., 2024b; Mohbat and Zaki, 2024). However, we find that splitting the training data into two separate stages may hinder training task diversity and efficiency. Therefore, we apply a single-stage training pipeline that combines the synthetic task with its corresponding image-caption pair. This simple method enriches task diversity during training and leads to better performance.

In contrast to previous works (Moor et al., 2023; Zhang et al., 2023b; Ma et al., 2024) which focus on a single domain or a single series of MLLMs per work, we conduct experiments across a variety of high-impact domains, such as biomedicine, food, and remote sensing, on general MLLMs of different sources and scales, such as Qwen2-VL-2B (Wang et al., 2024), LLaVA-v1.6-8B (Liu et al., 2024b), and Llama-3.2-VL-11B (Dubey et al., 2024). As shown in Figure 1, our resulting model, AdaMLLM (short for **Ada**pted **M**ultimodal **L**arge **L**anguage **M**odel), consistently outperforms general MLLMs across various domain-specific tasks.

In summary, our contributions include:

- To the best of our knowledge, we present the first systematic investigation of *domain-adaptive post-training* of MLLMs through extensive experiments across diverse domains and MLLMs.
- We comprehensively analyze the data synthesis pipeline and training strategy, revealing that task diversity and domain knowledge are key to the success of our method.
- We fully open-source our models, code, and data to facilitate future research and easy adaptation to new MLLMs and domains.

2 Related Work

Our work is related to domain-specific MLLM (Zhan et al., 2025; Liu et al., 2025; Hu et al., 2025; Cui et al., 2025), instruction synthesis, and MLLM training strategy. This section focuses on domain-specific MLLM, and the other topics are discussed in Appendix A.

Domain-Specific Data Initially, Moor et al. (2023) utilized multimodal interleaved data. With the rise of visual instruction tuning, research has shifted to synthesizing visual instructions (Li et al., 2023a). The approaches fall into two categories: (1) transforming existing datasets into visual instruction formats (Mohbat and Zaki, 2024; Yin et al., 2023) (2) prompting closed-source models to generate tasks from images/annotations (Zhang et al., 2023b; Li et al., 2024a; Chen et al., 2024b). Our work aligns with the second in synthesizing domain-specific data based on image-caption pairs, but we utilize open-source models to avoid privacy concerns and lead to even better performance.

Domain-Specific Training One type of domain-specific training begins with an unaligned LLM and visual encoder (Zhang et al., 2023b). Another type is post-training which starts with a well-aligned general MLLM (Li et al., 2024a). Compared to the first type, post-training is more efficient in terms of data and computation, making it our preferred method. In domain-specific post-training, previous works (Li et al., 2024a; Mohbat and Zaki, 2024; Chen et al., 2024b) adopt the two-stage training pipeline originally proposed for general MLLMs. We simplify this into a single stage to enhance task diversity within the training phase.

3 Method

We adapt MLLMs to domains via post-training. Figure 2 provides the method overview: we begin by synthesizing domain-specific tasks using a unified visual instruction synthesizer, followed by a consistency-based data filter. The synthetic tasks are then combined with image-captioning tasks into a single stage for post-training.

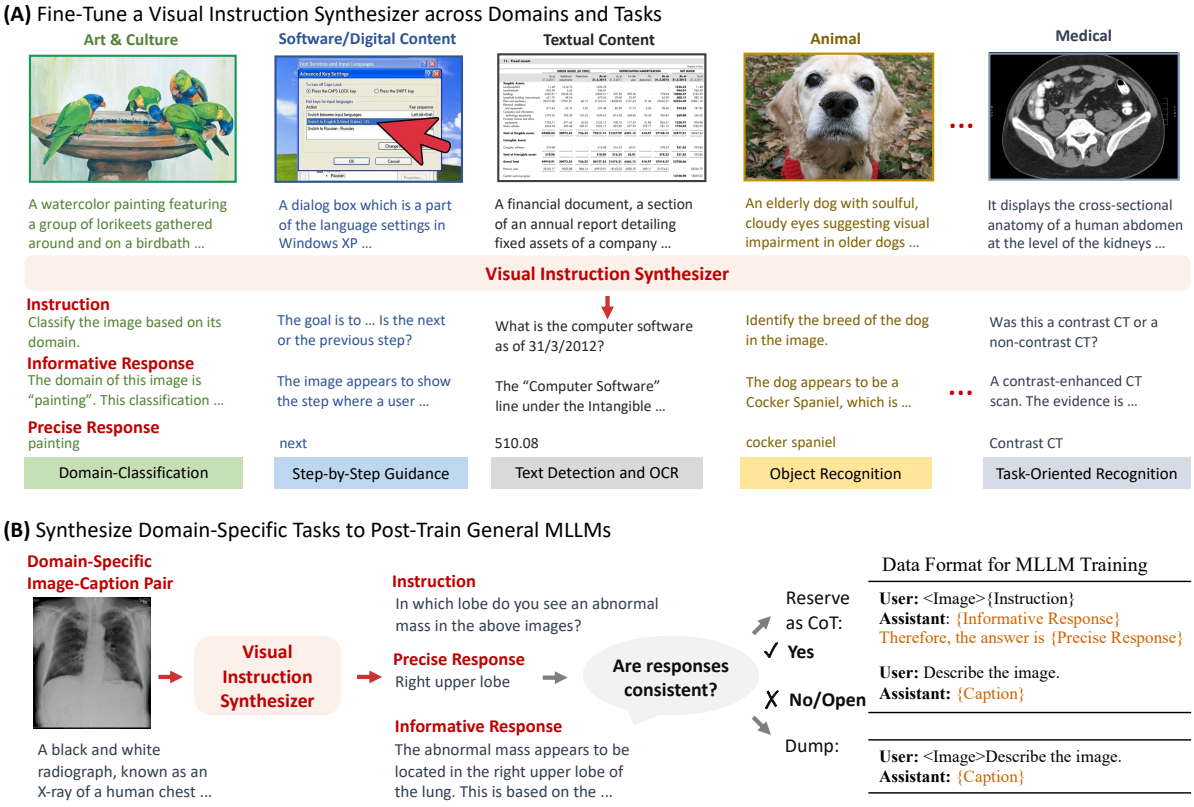


Figure 2: **Method Overview.** (A) We fine-tune a unified visual instruction synthesizer that generates diverse tasks based on image-caption pairs across various domains. (B) Using this synthesizer, we synthesize tasks based on domain-specific image-caption pairs and then apply a consistency-based data filter. The filtered synthetic tasks, combined with the original image captioning tasks, are employed to train general MLLMs through a single-stage post-training process, MLLM training loss is computed only on the part colored in orange.

3.1 Domain Visual Instruction Synthesis

The effectiveness of visual instruction tasks depends on diversity and accuracy, with domain knowledge being essential for domain adaptation (Li et al., 2023a). To meet these requirements, we propose a data synthesis approach comprising two components: a synthesizer that generates diverse tasks infused with domain knowledge, and a consistency-based filter to enhance accuracy.

3.1.1 Visual Instruction Synthesizer

We fine-tune an open-source MLLM to generate diverse tasks based on image-caption pairs across various domains, developing a *visual instruction synthesizer*. Instead of generating domain-specific tasks from scratch, which requires significant expertise, our synthesizer extracts tasks from existing data (Cheng et al., 2024a), thus reducing the reliance on domain experts. Furthermore, we incorporate specific designs to balance the utilization of image and text modalities.

Seed Data across Domains and Tasks We convert existing datasets (Xu et al., 2024; Chen et al., 2024a) into the seed data for fine-tuning our synthesizer, without any additional annotation. As shown in part (A) of Figure 2, each seed data example includes an image-caption pair as input and a related task triplet—comprising an instruction, an informative response, and a precise response—as output. Compared to the precise response which is often a single phrase, the informative response contains more reasoning steps. The seed data cover a wide range of 20 image domains and 191 tasks. Details on data construction are in Appendix B.

Modality-Balanced Multitask Tuning We fine-tune an open-source MLLM on the seed data to generate task triplets from image-caption pairs.

User: <Image>Describe the image.
Assistant: {Caption}

User: Answer with a precise response. {Instruction}
Assistant: {Precise Response}

User: Answer with an informative response. {Instruction}
Assistant: {Informative Response}

Table 1: **Data Format for Synthesizer Tuning.** Tuning loss is computed only on the part colored in orange.

As shown in Table 1, each seed data example is converted into a multi-turn conversation to fit the MLLM's conversational format. During fine-tuning, we calculate the tuning loss only on the conversational turns related to the task triplet, ensuring the synthesizer focuses on them.

Furthermore, since the task instruction annotations in the seed data (Xu et al., 2024) rely solely on images—biasing the fine-tuned synthesizer toward over-reliance on visual inputs—we replace 10% of the fine-tuning images with blank ones. This modality-balancing strategy encourages the model to leverage textual captions when visual inputs are ambiguous or uninformative, while preserving the quality of synthetic tasks for both complete and text-corrupted inputs. A detailed analysis is provided in Appendix F.

Task Synthesis for Target Domain After tuning, we use the synthesizer to generate task triplets from image-pairs in the target domain. For each image-caption pair, we input it into the synthesizer using the conversational format in Table 1 and extract the task triplet from the output accordingly.

3.1.2 Consistency-Based Filter

Developed from an open-source model without sufficient domain expertise, our synthesizer inevitably produces some inaccurate responses, necessitating data filtering. We propose a filtering method based on inherent consistency, which improves data quality while reducing the need for expert validation.

As shown in part (B) in Figure 2, we prompt an open-source language model to classify each task triplet into one of three categories: consistent, inconsistent, or open. The consistent and inconsistent categories indicate whether the precise and informative responses align, while the open category indicates tasks that request open-ended responses (e.g., background information). The prompt template is in Figure 6 in Appendix. We discard triplets classified as inconsistent, and those classified as open due to their ambiguity. In contrast to ensemble methods (multi-model voting; Dietterich, 2000) and self-consistency (multi-output sampling; Wang et al., 2022), we select each task triplet based on a single output from a single synthesizer, leveraging internal consistency within that output.

For consistent triplets, we combine the informative and precise responses into a chain-of-thought (CoT; Wei et al., 2022) format. The informative response serves as the reasoning process, and the precise response serves as the final conclusion, ensuring both informativeness and accuracy.

3.2 Single-Stage Post-Training

Domain-specific post-training for MLLMs (Li et al., 2024a; Chen et al., 2024b; Mohbat and Zaki, 2024) typically follows the multi-stage paradigm used in general MLLM training: first on image-caption pairs, then on visual instruction tasks (Liu et al., 2024c). However, task diversity in domain-specific training is often more limited than in general training, and splitting the training into two stages may further reduce diversity within each stage, negatively impacting the task generalization of the trained models (Wei et al., 2021). To mitigate this, we propose combining the training data into a single stage. As shown in part (B) of Figure 2, each training example includes two tasks:

- *Image Captioning Task*: A question prompting the MLLM to describe the image is randomly chosen from a pool in (Chen et al., 2024a) as the task instruction, with the original caption as the ground-truth response.
- *Synthetic Visual Instruction Task*: For each image-caption pair with a synthetic task after filtering, we combine it with the image captioning task in a multi-turn format. If no synthetic task remains, only the captioning task is used.

Following (Liu et al., 2024c), we train on the data using the next-token prediction objective (Radford et al., 2018), computing loss only on the response part of each instruction-response pair.

4 Experiment Settings

We conduct experiments in high-impact domains including biomedicine, food, and remote sensing, because they are the only ones for which we can access public data without raising ethical concerns. For each domain, we perform post-training to adapt general MLLMs and evaluate model performance on various domain-specific tasks.

Note that our implementation is fully open-source, and the only required change for applying our method to a new domain is to replace the image-caption source with that of the new domain.

Image-Caption Data Source For biomedicine domain, we use two sources: PMC^{Raw} in Li et al. (2024a) and PMC^{Refined} in Chen et al. (2024b). For food domain, we collect data from Recipe1M (Salvador et al., 2017). For remote

Biomedicine	SLAKE		PathVQA		VQA-RAD		PMC-VQA
	OPEN	CLOSED	OPEN	CLOSED	OPEN	CLOSED	
LLaVA-v1.6-8B	49.2	62.3	15.2	47.7	45.9	56.3	36.5
LLaVA-Med-8B	43.4	50.2	10.1	59.2	35.0	62.5	37.1
PubMedVision-8B	50.0	68.3	17.0	67.5	43.3	67.3	40.4
AdaMLLM-8B from PMC^{Raw}	56.8	**76.4**	19.7	**79.3**	51.0	80.5	44.3
AdaMLLM-8B from $PMC^{Refined}$	**58.0**	73.3	**22.9**	78.6	**59.8**	**81.3**	**47.9**
Qwen2-VL-2B	50.0	52.4	17.8	38.7	37.0	46.7	45.8
LLaVA-Med-2B	43.4	55.5	11.8	60.1	37.1	58.8	41.2
PubMedVision-2B	45.2	63.2	18.2	**64.7**	41.3	67.3	43.2
AdaMLLM-2B from PMC^{Raw}	53.2	**75.2**	20.1	63.8	49.8	74.6	43.5
AdaMLLM-2B from $PMC^{Refined}$	**60.2**	75.0	**20.6**	53.6	**58.0**	**76.1**	**46.5**
Llama-3.2-11B	56.2	63.9	22.7	72.1	46.9	63.6	**51.9**
LLaVA-Med-11B	47.6	58.7	14.6	69.5	38.0	69.1	47.5
PubMedVision-11B	49.1	74.3	19.3	70.9	46.2	73.9	47.1
AdaMLLM-11B from PMC^{Raw}	56.7	**77.6**	22.2	**87.3**	55.0	76.1	49.9
AdaMLLM-11B from $PMC^{Refined}$	**59.5**	76.4	**24.3**	84.9	**57.4**	**79.8**	**51.9**

Table 2: **Biomedicine Task Performance** of general MLLMs and MLLMs after domain-adaptive post-training. The image-caption sources for AdaMLLM from PMC^{Raw} and AdaMLLM from $PMC^{Refined}$ are PMC^{Raw} and $PMC^{Refined}$, respectively.

sensing domain, we collect data from five image-captioning tasks. Data details are in Appendix C.

Visual Instruction Synthesis Our synthesizer is fine-tuned from LLaVA-v1.6-Llama3-8B. For the consistency-based filter, we prompt Llama-3-8B (Dubey et al., 2024) to evaluate the consistency of each synthesized task triplet. Detailed implementations and costs are in Appendix D.

Post-Training & Task Evaluation Using synthetic data from the LLaVA-v1.6-Llama3-8B-based synthesizer, we conduct post-training on LLaVA-v1.6-Llama3-8B itself. Besides, we use the same synthetic data to post-train Qwen2-VL-2B-Instruct and Llama-3.2-11B-Vision-Instruct to assess effectiveness across different models and scales. Training hyper-parameters and costs are in Appendix E.

After post-training, we evaluate MLLMs on domain-specific tasks without further fine-tuning. Evaluation details are in Appendix H.

Baseline For biomedicine domain, we compare with two baselines: (1) LLaVA-Med (Li et al., 2024a) which uses text-only GPT-4 to synthesize tasks from PMC^{Raw}, and (2) PubMedVision (Chen et al., 2024b) which uses GPT-4V to synthesize tasks from $PMC^{Refined}$. To ensure a fair comparison, we reproduce their methods using updated backbone models, and the original results of LLaVA-Med and PubMedVision are reported in Table 16 in the Appendix. For food domain, we compare with LLaVA-Chef (Mohbat and Zaki, 2024) which uses manual rules to transform image-recipe pairs from Recipe1M into multiple tasks. LLaVA-Med, PubMedVision and LLaVA-Chef all employ two-stage post-training. For remote sensing, we compare with a baseline that uses GPT-4o (Hurst et al., 2024) to synthesize instructions based on the same image-caption pairs as ours, and train MLLMs using the same single-stage post-training. The resulting model is referred to as RS-4o.

Food	Recipe	Nutrition	Food101	FoodSeg
LLaVA-v1.6-8B	18.6	29.6	47.9	38.9
LLaVA-Chef-8B	23.1	29.1	46.8	14.5
AdaMLLM-8B	**24.8**	**36.1**	**65.3**	**42.0**
Qwen2-VL-2B	18.2	36.4	**73.9**	19.9
LLaVA-Chef-2B	**24.1**	24.5	68.8	7.7
AdaMLLM-2B	24.0	**41.2**	72.0	**23.9**
Llama-3.2-11B	23.7	40.0	80.8	**47.6**
LLaVA-Chef-11B	25.7	26.2	82.1	16.7
AdaMLLM-11B	**26.1**	**41.0**	**82.2**	42.0

Remote Sensing	CLRS	UC Merced	FloodNet	NWPU
LLaVA-v1.6-8B	54.3	64.9	70.0	26.1
RS-4o-8B	50.3	64.5	58.1	**74.2**
AdaMLLM-8B	**66.9**	**72.1**	**72.0**	72.1
Qwen2-VL-2B	48.9	61.0	55.7	26.0
RS-4o-2B	51.2	**67.0**	53.7	**56.7**
AdaMLLM-2B	**55.0**	61.8	**62.2**	56.0
Llama-3.2-11B	55.7	74.2	60.8	20.8
RS-4o-11B	59.3	57.6	51.5	**71.9**
AdaMLLM-11B	**64.9**	**81.5**	**62.1**	67.8

Table 3: **Food and Remote Sensing Task Performance** of general MLLMs and MLLMs after domain-adaptive post-training.

Image-Caption	Recipe1M				PMC^{Raw}				PMC^{Refined}			
Train Pipeline	*Two-stage*		*Single-stage*		*Two-stage*		*Single-stage*		*Two-stage*		*Single-stage*	
Instruction	Rule	Ours	Rule	Ours	GPT-4	Ours	GPT-4	Ours	GPT-4V	Ours	GPT-4V	Ours
LLaVA-v1.6-8B	28.4	**29.0**	34.1	**42.0** ↑	42.5	**55.6**	46.1	**58.3** ↑	50.5	**58.6**	55.5	**60.3** ↑
Qwen2-VL-2B	31.3	**38.2**	31.9	**40.3** ↑	44.0	**55.5**	41.3	**54.3** ↓	49.0	**59.5**	51.6	**55.7** ↓
Llama-3.2-11B	37.7	**40.9**	36.6	**47.8** ↑	49.3	**59.2**	48.8	**60.7** ↑	54.4	**60.3**	53.7	**62.0** ↑

Table 4: **Domain-Specific Task Performance of MLLMs after Post-Training** with different synthetic data and training pipelines. We report the average performance in each domain, with detailed results in Table 17 in Appendix. When the image-caption source and training pipeline are fixed, synthetic data of better performance are marked in **bold**. When the image-caption source is fixed and our synthetic data are used, numbers marked with ↑ indicate that single-stage training outperforms two-stage training, while ↓ indicates the opposite.

	SLAKE		PathVQA		VQA-RAD		PMC-VQA	AVERAGE	IMPROV.
	OPEN	CLOSED	OPEN	CLOSED	OPEN	CLOSED			
Qwen2.5-VL-3B-Ins.	54.4	63.5	**19.0**	55.1	44.3	57.0	49.2	48.9	
AdaMLLM-3B	**62.2**	**79.8**	18.7	**83.6**	**54.4**	**82.4**	**51.4**	**61.8**	+12.9
Gemma-3-4B-it	44.5	75.2	**22.5**	62.4	36.0	61.4	30.9	47.6	
AdaMLLM-4B	**58.5**	**79.1**	20.7	**72.5**	**52.3**	**77.2**	**46.6**	**58.1**	+10.5
InternVL3-1B	57.4	63.7	18.7	65.1	40.6	64.7	35.3	49.4	
AdaMLLM-1B	**60.6**	**67.8**	**20.9**	**67.7**	**50.9**	**73.2**	**41.2**	**54.6**	+5.2

Table 5: **Biomedicine Task Performance** of general MLLMs and MLLMs after domain-adaptive post-training using our method. The image-caption source is PMC^{Refined}.

5 Main Results

Overall Performance As shown in Tables 2 and 3, ours consistently enhances MLLM performance, outperforming baselines across various domain-specific tasks. Although our synthesizer is based on LLaVA-v1.6-8B, we observe consistent improvements on Qwen2-VL-2B and Llama-3.2-11B, demonstrating its effectiveness across different models and scales. Among the evaluated tasks, VQA-RAD, Recipe1M and NWPU can be regarded as partially seen tasks, with VQA-RAD included in our seed data for fine-tuning the synthesizer, Recipe1M and NWPU included in the image-caption source[1]. Nevertheless, AdaMLLM shows consistent gains on other unseen tasks, demonstrating strong task generalization in the target domain.

Comparison of Synthetic Task and Training Pipeline In addition to the overall comparison, we assess the effectiveness of our synthetic tasks and single-stage training separately by varying one factor at a time. As shown in Table 4, we conduct both two-stage and single-stage post-training with synthetic tasks generated by different methods: manual rules in LLaVA-Chef, GPT-4 in LLaVA-Med, and GPT-4V in PubMedVision. Our synthetic tasks consistently outperform others across both training pipelines. Furthermore, with our synthetic tasks, single-stage training surpasses two-stage training in most of the experiments.

Extended Experiments on More MLLMs Despite the results on three MLLMs in Tables 2 and 3, we further validate our method on three more recent MLLMs: Qwen2.5-VL-3B-Instruct (Bai et al., 2025), Gemma-3-4B-it (Team et al., 2025), and InternVL3-1B (Zhu et al., 2025). As shown in Table 5, our models consistently outperform the latest MLLMs, achieving improvements of up to +12.9 points on average.

6 Ablations

To evaluate the effectiveness of each component, we conduct ablations to post-train LLaVA-v1.6-8B with different settings. We report the average task performance within each domain for the trained models in Table 6.

Domain Knowledge, Task (CoT) Format & Seed Data Our synthetic tasks are designed to integrate both (1) domain knowledge and (2) a visual instruction task format with Chain-of-Thought response. To assess the role of domain knowledge, we compare our method with general tasks that preserve the same CoT response format but exclude

[1]Test/validation sets of VQA-RAD, Recipe1M and NWPU are not included.

	Ours	General CoT Task	General CoT Task + Domain Caption	w/o Blank Image	w/o Consistency Filter		Image Caption Only	Synthetic Task Only	Two-Stage Reuse Caption
					Precise	Informative			
BioMed.	**58.3**	49.8	55.3	55.8	31.2	44.4	26.7	54.2	54.9
Food	**42.0**	36.0	38.6	35.9	37.9	37.6	25.6	36.8	36.6

Table 6: **Ablation Results.** "General CoT Task" trains on seed data processed into our task format,"General CoT Task + Domain Caption" mixes the processed seed data with domain-specific image-caption pairs. "w/o Blank Image" fine-tunes the synthesizer without replacing 10% of images with blank ones. "w/o Consistency Filter" removes the consistency-based filter and trains with either precise or informative responses. "Image Caption Only" removes synthetic task, and "Synthetic Task Only" removes image captioning task. "Two-Stage Reuse Caption" conducts two-stage training with the second stage reusing the caption data from the first stage.

domain-specific knowledge. As shown in Table 6, our method outperforms the "General CoT Task", underscoring the importance of domain knowledge.

Importantly, "General CoT Task" is constructed from our seed data—originally used to fine-tune the synthesizer—and filtered using the same consistency-based method. This suggests that the performance gain of our method is not simply due to knowledge distillation from the seed data.

Moreover, our method outperforms "General CoT Task + Domain Caption", indicating that naively combining domain-specific image-caption pairs with general tasks is insufficient. In contrast, our synthesis pipeline effectively transforms the domain knowledge embedded in image-caption pairs into a format that general MLLMs can learn from more effectively.

Visual Instruction Synthesis To balance modality-utilization, we replace some of the images with blank images during the fine-tuning of synthesizer. The effectiveness of this strategy on MLLM performance is demonstrated in "w/o Blank Image".

To improve response accuracy, we design a consistency-based filter. As shown in Table 6, removing this filter results in decreased model performance, regardless of whether the response contains only precise or informative content.

Single-Stage Post-Training Our motivation for combining data into a single stage is to enhance training task diversity. This efficacy is evident in the ablation results in Table 6, where removing either the synthetic task ("Image Caption Only") or the image captioning task ("Synthetic Task Only") degrades model performance, even when the caption data is reused in the second stage ("Two-Stage Reuse Caption").

Finetune Input	-	**Image**	**Caption**	**Image + Caption**	
Blank Image	-	-	-	✗	✓
Diversity	52.5	68.0	75.2	81.0	**85.5**
Knowledge	72.5	95.0	93.8	97.5	**98.1**
Complexity	43.8	77.9	75.3	80.0	**83.2**
Accuracy	63.8	60.0	65.6	66.3	**71.3**

Table 7: **Quality of Synthetic Tasks by Different Visual Instruction Synthesizers.** Column 1 presents results from the MLLM without fine-tuning (i.e., the base LLaVA in our experiment settings). Columns 2-5 show results after fine-tuning the MLLM using our seed data to synthesize tasks based on different inputs. Besides, Column 5 replaces 10% of the images with blank ones.

	w/o Filter			w/ Filter	
	Consist.	Precise Acc	Info. Acc	Consist.	Acc
BioMed.	30.3	64.3	61.0	92.2	75.1
Food	35.7	77.2	75.5	97.1	84.3

Table 8: **Quality of Responses with/without Using Consistency-Based Filter**, assessed in terms of consistency between precise and informative responses (Consist.), accuracy of precise responses (Precise Acc), accuracy of informative responses (Info. Acc), and accuracy of combined responses (Acc).

7 Analysis

We conduct a detailed analysis on the synthesis pipeline and the synthesized data

7.1 Domain Visual Instruction Synthesis

Visual Instruction Synthesizer We compare tasks generated by synthesizers with different designs using a validation set from our seed data. Specifically, we conduct human evaluation of data quality in the following aspects: task diversity, domain knowledge utilization, task complexity and response accuracy (detailed definition and scoring criteria are in Appendix G).

The results in Table 7 indicate that fine-tuning for task synthesis using either image (Zhao et al., 2023) or caption (Cheng et al., 2024a) inputs yields improvements in most aspects. Our design, which

Image-Caption	Recipe1M		PMC^{Raw}		PMC^{Refined}	
Instruction	Rule	Ours	GPT-4	Ours	GPT-4V	Ours
Diversity	23.5	**52.9**	47.1	**58.8**	64.7	**76.5**
Knowledge	20.9	**21.9**	44.9	**58.9**	**67.7**	63.2
Complexity	38.4	**69.9**	41.7	**83.2**	49.6	**80.5**
Accuracy	**98.7**	84.3	**84.4**	75.1	**87.5**	79.6

Table 9: **Quality of Synthetic Tasks** by our method, manual rules, GPT-4, and GPT-4V.

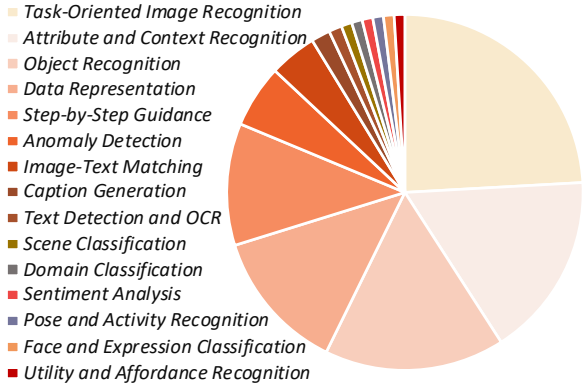


Figure 3: **Task Type Distribution** of all our synthetic tasks based on all image-caption sources.

employs both image and caption inputs, leads to even higher performance. Besides, replacing 10% of the images with blank ones achieves the highest quality (further analysis is in Appendix F).

Consistency-Based Filter Our consistency-based filter is designed to select tasks with inherent consistency, thereby increasing data accuracy. As shown in Table 8, using the filter significantly increases the consistency between precise and informative responses, making the combination of them in the CoT format reasonable. As a result, the filter successfully increase response accuracy.

7.2 Domain-Specific Synthetic Data

Quantitative Analysis Table 9 presents the data quality scores for synthetic tasks generated by different methods. Our tasks are diverse and complex, demonstrating a high utilization of domain knowledge. The distribution of task types for all our instruction-response pairs is displayed in Figure 3. This explains the effectiveness of our method in enhancing MLLM performance across domain-specific tasks. However, our method underperforms the baselines in terms of response accuracy, with manual rules achieving nearly 100% accuracy and GPT-4 and GPT-4V reaching around 85%. This may because of the increased complexity of our synthesized tasks, which make generating accurate responses more challenging. These results indicate the need for further improvements to enhance response accuracy, even with complex tasks.

Qualitative Analysis Figure 4 presents cases of synthetic tasks by different methods when given the same image-caption pair. In case (A), the rule-based task is a simple transformation of the recipe caption, ignoring the image information. In contrast, our task conducts a detailed analysis of the food's state in the image and accurately matches it with the cooking step in the caption, demonstrating a higher level of domain knowledge utilization. In case (B), both our task and the GPT-4 synthesized task focus on interpreting intent. While the GPT-4 task straightforwardly asks for the intent, our task increases task complexity by requiring inference from the context to make a "yes/no/not sure" choice. In case (C) with multiple sub-images, our task type is distinct in requiring the identification of the least similar image among the group, showcasing task diversity. More cases are provided in Figure 8 in Appendix.

8 Conclusion

This paper investigates adapting general MLLMs to specific domains via post-training. To synthesize domain-specific visual instruction tasks, we develop a unified visual instruction synthesizer that generates instruction-response pairs based on domain-specific image-caption data, and then apply a consistency-based filter to improve data accuracy. This enables us to effectively synthesize diverse tasks with high domain knowledge utilization. For the post-training pipeline, we propose combining the synthetic tasks with image-captioning tasks into a single training stage to enhance task diversity. In multiple high-impact domains, our resulting model, AdaMLLM, consistently outperforms general MLLMs across various domain-specific tasks.

Limitations

While synthetic data reduce the need of expert annotation, it is crucial to acknowledge the potential limitations. Our work, along with other works utilizing synthetic data (Liu et al., 2024d), is inevitably constrained by the possibility of introducing hallucinations. As shown in our analysis in Section 7.2, the accuracy of our synthetic tasks remains imperfect, underscoring the need for further improvements to enhance response reliability, even with highly complex tasks.

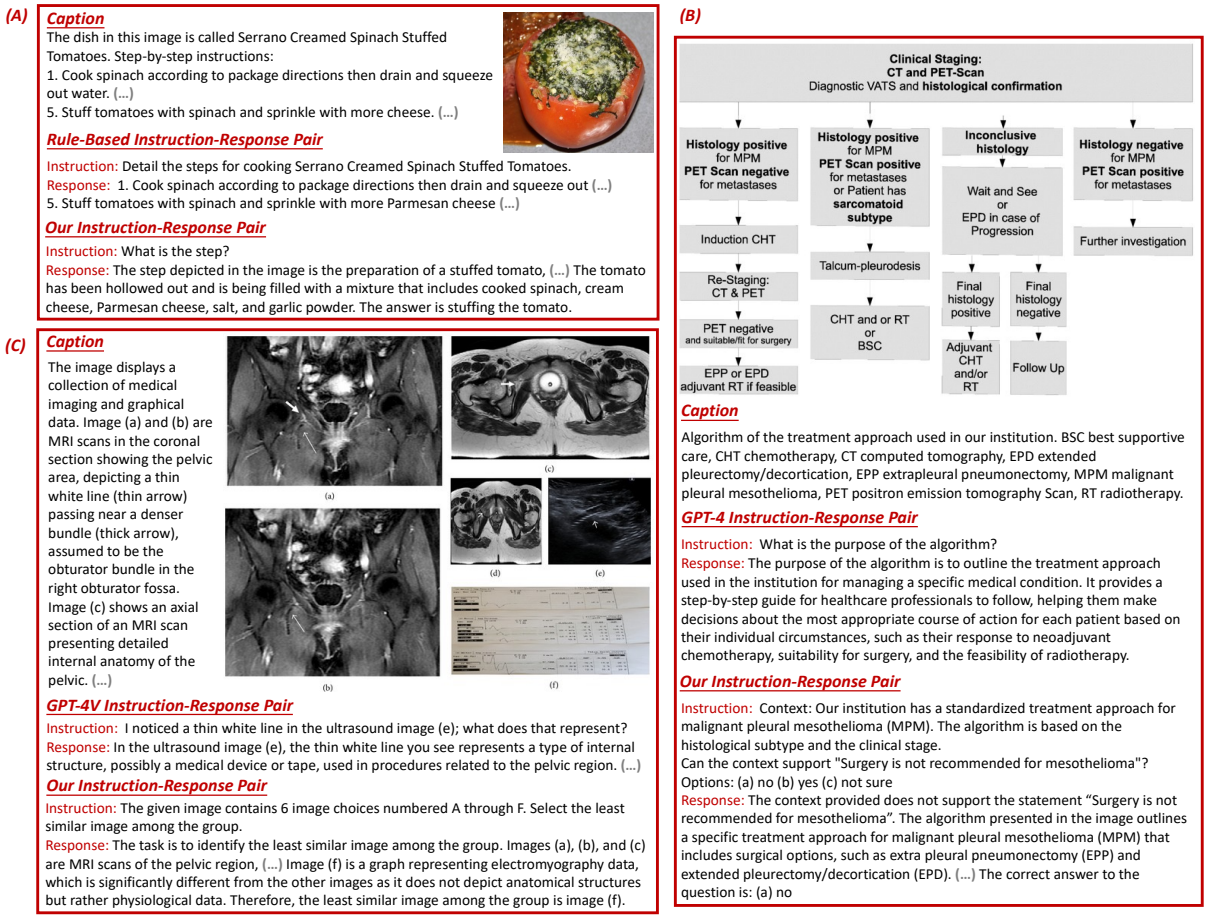


Figure 4: **Cases of Instruction-Response Pairs** synthesized by our method, manual rules, GPT-4, and GPT-4V.

Furthermore, future research may take into account the preferences of the specific domain. For instance, when dealing with animals, semantic-level information might be more important, while for medical images, local detail information should be given greater attention.

Ethics Statement

All the datasets and models used in this work are publicly available.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, and 1 others. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Gagan Bhatia, El Moatez Billah Nagoudi, Hasan Cavusoglu, and Muhammad Abdul-Mageed. 2024. Fintral: A family of gpt-4 level multimodal financial large language models. *arXiv preprint arXiv:2402.10986*.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, pages 446–461. Springer.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. 2024a. Allava: Harnessing gpt4v-synthesized data for a lite vision-language model. *arXiv preprint arXiv:2402.11684*.

Junying Chen, Ruyi Ouyang, Anningzhe Gao, Shunian Chen, Guiming Hardy Chen, Xidong Wang, Ruifei Zhang, Zhenyang Cai, Ke Ji, Guangjun Yu, and 1 others. 2024b. Huatuogpt-vision, towards injecting medical visual knowledge into multimodal llms at scale. *arXiv preprint arXiv:2406.19280*.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua

Lin. 2023. Sharegpt4v: Improving large multimodal models with better captions. *arXiv preprint arXiv:2311.12793*.

Lin Chen and Long Xing. 2024. Open-llava-next: An open-source implementation of llava-next series for facilitating the large multi-modal model community. https://github.com/xiaoachen98/Open-LLaVA-NeXT.

Daixuan Cheng, Yuxian Gu, Shaohan Huang, Junyu Bi, Minlie Huang, and Furu Wei. 2024a. Instruction pre-training: Language models are supervised multitask learners. *arXiv preprint arXiv:2406.14491*.

Daixuan Cheng, Shaohan Huang, and Furu Wei. 2024b. Adapting large language models via reading comprehension. In *The Twelfth International Conference on Learning Representations*.

Qimin Cheng, Haiyan Huang, Yuan Xu, Yuzhuo Zhou, Huanying Li, and Zhongyuan Wang. 2022. Nwpu-captions dataset and mlca-net for remote sensing image captioning. *IEEE Trans. Geosci. Remote. Sens.*, 60:1–19.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, and 1 others. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Haotian Cui, Alejandro Tejada-Lapuerta, Maria Brbić, Julio Saez-Rodriguez, Simona Cristea, Hani Goodarzi, Mohammad Lotfollahi, Fabian J Theis, and Bo Wang. 2025. Towards multimodal foundation models in molecular cell biology. *Nature*, 640(8059):623–633.

Thomas G Dietterich. 2000. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yuxian Gu, Pei Ke, Xiaoyan Zhu, and Minlie Huang. 2022. Learning instructions with unlabeled data for zero-shot cross-task generalization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1617–1634.

Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. 2020. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*.

Yuan Hu, Jianlong Yuan, Congcong Wen, Xiaonan Lu, Yu Liu, and Xiang Li. 2025. Rsgpt: A remote sensing vision language model and benchmark. *ISPRS Journal of Photogrammetry and Remote Sensing*, 224:272–286.

Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Barun Patra, and 1 others. 2023. Language is not all you need: Aligning perception with language models. *Advances in Neural Information Processing Systems*, 36:72096–72109.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Jason J Lau, Soumya Gayen, Asma Ben Abacha, and Dina Demner-Fushman. 2018. A dataset of clinically generated visual questions and answers about radiology images. *Scientific data*, 5(1):1–10.

Nicholas Lee, Thanakul Wattanawong, Sehoon Kim, Karttikeya Mangalam, Sheng Shen, Gopala Anumanchipali, Michael W Mahoney, Kurt Keutzer, and Amir Gholami. 2024. Llm2llm: Boosting llms with novel iterative data enhancement. *arXiv preprint arXiv:2403.15042*.

Seowoo Lee, Jiwon Youn, Hyungjin Kim, Mansu Kim, and Soon Ho Yoon. 2025. Cxr-llava: a multimodal large language model for interpreting chest x-ray images. *European Radiology*, pages 1–13.

Chen Li, Yixiao Ge, Dian Li, and Ying Shan. 2023a. Vision-language instruction tuning: A review and analysis. *arXiv preprint arXiv:2311.08172*.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2024a. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36.

Haifeng Li, Hao Jiang, Xin Gu, Jian Peng, Wenbo Li, Liang Hong, and Chao Tao. 2020. CLRS: continual learning benchmark for remote sensing image scene classification. *Sensors*, 20(4):1226.

Haoran Li, Qingxiu Dong, Zhengyang Tang, Chaojun Wang, Xingxing Zhang, Haoyang Huang, Shaohan Huang, Xiaolong Huang, Zeqiang Huang, Dongdong Zhang, and 1 others. 2024b. Synthetic data (almost) from scratch: Generalized instruction tuning for language models. *arXiv preprint arXiv:2402.13064*.

Junxian Li, Di Zhang, Xunzhi Wang, Zeying Hao, Jingdi Lei, Qian Tan, Cai Zhou, Wei Liu, Yaotian Yang, Xinrui Xiong, and 1 others. 2025. Chemvlm: Exploring the power of multimodal large language models in

chemistry area. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 415–423.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Omer Levy, Luke Zettlemoyer, Jason E Weston, and Mike Lewis. 2023b. Self-alignment with instruction backtranslation. In *The Twelfth International Conference on Learning Representations*.

Bo Liu, Li-Ming Zhan, Li Xu, Lin Ma, Yan Yang, and Xiao-Ming Wu. 2021. Slake: A semantically-labeled knowledge-enhanced dataset for medical visual question answering. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1650–1654. IEEE.

Dongyang Liu, Renrui Zhang, Longtian Qiu, Siyuan Huang, Weifeng Lin, Shitian Zhao, Shijie Geng, Ziyi Lin, Peng Jin, Kaipeng Zhang, Wenqi Shao, Chao Xu, Conghui He, Junjun He, Hao Shao, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. 2024a. SPHINX-X: scaling data and parameters for a family of multimodal large language models. In *ICML*. OpenReview.net.

Fenglin Liu, Zheng Li, Qingyu Yin, Jinfa Huang, Jiebo Luo, Anshul Thakur, Kim Branson, Patrick Schwab, Bing Yin, Xian Wu, and 1 others. 2025. A multimodal multidomain multilingual medical foundation model for zero shot clinical diagnosis. *npj Digital Medicine*, 8(1):86.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024b. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024c. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, and 1 others. 2024d. Best practices and lessons learned on synthetic data for language models. *arXiv preprint arXiv:2404.07503*.

Xiaoqiang Lu, Binqiang Wang, Xiangtao Zheng, and Xuelong Li. Exploring models and data for remote sensing image caption generation. *IEEE Transactions on Geoscience and Remote Sensing*, 56(4):2183–2195.

Yizhen Luo, Jiahuan Zhang, Siqi Fan, Kai Yang, Yushuai Wu, Mu Qiao, and Zaiqing Nie. 2023a. Biomedgpt: Open multimodal generative pre-trained transformer for biomedicine. *arXiv preprint arXiv:2308.09442*.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023b. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Yingzi Ma, Yulong Cao, Jiachen Sun, Marco Pavone, and Chaowei Xiao. 2024. Dolphins: Multimodal language model for driving. In *ECCV (45)*, volume 15103 of *Lecture Notes in Computer Science*, pages 403–420. Springer.

Fnu Mohbat and Mohammed J Zaki. 2024. Llava-chef: A multi-modal generative model for food recipes. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 1711–1721.

Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Yash Dalmia, Jure Leskovec, Cyril Zakka, Eduardo Pontes Reis, and Pranav Rajpurkar. 2023. Med-flamingo: a multimodal medical few-shot learner. In *Machine Learning for Health (ML4H)*, pages 353–367. PMLR.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*.

OpenAI. 2023. Gpt-4v(ision) system card.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Qixiang Ye, and Furu Wei. 2024. Grounding multimodal large language models to the world. In *The Twelfth International Conference on Learning Representations*.

Bo Qu, Xuelong Li, Dacheng Tao, and Xiaoqiang Lu. 2016. Deep semantic understanding of high resolution remote sensing image. In *CITS*, pages 1–5. IEEE.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. Improving language understanding by generative pre-training.

Maryam Rahnemoonfar, Tashnim Chowdhury, Argho Sarkar, Debvrat Varshney, Masoud Yari, and Robin Roberson Murphy. 2021. Floodnet: A high resolution aerial imagery dataset for post flood scene understanding. *IEEE Access*, 9:89644–89654.

Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, and Antonio Torralba. 2017. Learning cross-modal embeddings for cooking recipes and food images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3020–3028.

Chameleon Team. 2024. Chameleon: Mixed-modal early-fusion foundation models. *CoRR*, abs/2405.09818.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Quin Thames, Arjun Karpur, Wade Norris, Fangting Xia, Liviu Panait, Tobias Weyand, and Jack Sim. 2021. Nutrition5k: Towards automatic nutritional understanding of generic food. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8903–8911.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Cong Wei, Yujie Zhong, Haoxian Tan, Yingsen Zeng, Yong Liu, Zheng Zhao, and Yujiu Yang. 2024. Instructseg: Unifying instructed visual segmentation with multi-modal large language models. *CoRR*, abs/2412.14006.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Xiongwei Wu, Xin Fu, Ying Liu, Ee-Peng Lim, Steven CH Hoi, and Qianru Sun. 2021. A large-scale benchmark for food image segmentation. In *Proceedings of the 29th ACM international conference on multimedia*, pages 506–515.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2023. Wizardlm: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*.

Zhiyang Xu, Chao Feng, Rulin Shao, Trevor Ashby, Ying Shen, Di Jin, Yu Cheng, Qifan Wang, and Lifu Huang. 2024. Vision-flan: Scaling human-labeled tasks in visual instruction tuning. *arXiv preprint arXiv:2402.11690*.

Yi Yang and Shawn Newsam. 2010. Bag-of-visual-words and spatial extensions for land-use classification. In *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM GIS)*.

Yuehao Yin, Huiyan Qi, Bin Zhu, Jingjing Chen, Yu-Gang Jiang, and Chong-Wah Ngo. 2023. Foodlmm: A versatile food assistant using large multi-modal model. *arXiv preprint arXiv:2312.14991*.

Zhiqiang Yuan, Wenkai Zhang, Kun Fu, Xuan Li, Chubo Deng, Hongqi Wang, and Xian Sun. 2022. Exploring a fine-grained multiscale method for cross-modal remote sensing image retrieval. *IEEE Trans. Geosci. Remote. Sens.*, 60:1–19.

Xiang Yue, Tuney Zheng, Ge Zhang, and Wenhu Chen. 2024. Mammoth2: Scaling instructions from the web. *arXiv preprint arXiv:2405.03548*.

Yang Zhan, Zhitong Xiong, and Yuan Yuan. 2025. Skyeyegpt: Unifying remote sensing vision-language tasks via instruction tuning with large language model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 221:64–77.

Sheng Zhang, Yanbo Xu, Naoto Usuyama, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, Cliff Wong, and 1 others. 2023a. Large-scale domain-specific pretraining for biomedical vision-language processing. *arXiv preprint arXiv:2303.00915*, 2(3):6.

Wei Zhang, Miaoxin Cai, Tong Zhang, Yin Zhuang, and Xuerui Mao. 2024. Earthgpt: A universal multimodal large language model for multi-sensor image comprehension in remote sensing domain. *IEEE Transactions on Geoscience and Remote Sensing*.

Xiaoman Zhang, Chaoyi Wu, Ziheng Zhao, Weixiong Lin, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023b. Pmc-vqa: Visual instruction tuning for medical visual question answering. *arXiv preprint arXiv:2305.10415*.

Henry Hengyuan Zhao, Pan Zhou, and Mike Zheng Shou. 2023. Genixer: Empowering multimodal large language models as a powerful data generator. *arXiv preprint arXiv:2312.06731*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, and 1 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

A Extended Related Work

Text-only/Visual-language Instruction Synthesis For text-only instruction synthesis, the most notable distinction between our work and these previous works is our inclusion of images as an additional source of information. In addition to the input modality, our method also differs in several key aspects. Instead of distilling knowledge from strong models (Xu et al., 2023; Mukherjee et al., 2023; Li et al., 2024b), we focus on learning from image-caption sources. Additionally, we outperform rule-based methods (Cheng et al., 2024b; Gu et al., 2022) by increasing instruction diversity. Iterative techniques (Li et al., 2023b; Yue et al., 2024; Lee et al., 2024) could complement our method. Among all text-only instruction synthesis works, we draw the most inspiration from Instruct Pre-Training (Cheng et al., 2024a). However, we introduce a consistency filter that significantly improves accuracy and reduces the need for domain expert annotation, allowing us to surpass synthesizers based on closed-source models. Additionally, we incorporate several design choices to balance the generalization challenge between image and text inputs.

For visual-language instruction synthesis, using closed-source/open-source models to generate data is common in general MLLM training (Li et al., 2023a). Notable works include LLaVA (Liu et al., 2024c), which uses text-only GPT-4 to generate instructions based on captions as if the model could "see" the image, leading to exceptional general-task performance. Additionally, ALLaVA (Chen et al., 2024a) leverages GPT-4V to generate large-scale visual instructions from image-caption pairs while also augmenting answers from Vision-FLAN (Xu et al., 2024). However, for domain-specific MLLMs, particularly in private domains, closed-source models pose privacy concerns. Thus, we focus on open-source models for instruction synthesis. Even with this focus, our method still outperforms LLaVA-Med (which uses text-only GPT-4) and PubMedVision (which uses GPT-4V), as shown in Section 5, due to our superior utilization of domain knowledge, task diversity, and complexity (discussed in Section 7). The most relevant open-source model approach is Genxier (Zhao et al., 2023), which fine-tunes open-source MLLMs to synthesize instructions from images. However, whereas Genxier focuses on general training, our work is specifically dedicated to domain-specific adaptation. Moreover, we reproduce a Genxier-like baseline (Column 2, Table 7), where image-only utilization underperforms our method.

Multi/Single-Stage MLLM Training The training of general MLLMs typically starts with an unaligned LLM (Brown, 2020; Chowdhery et al., 2023; Touvron et al., 2023) and visual encoder (Radford et al., 2021), and often proceeds in two stages. One representative example is LLaVA (Liu et al., 2024c), which first trains on image-caption pairs, and then on visual instructions. Note that "native multimodal language models" also exist, such as Kosmos (Huang et al., 2023; Peng et al., 2024) and Chameleon (Team, 2024), where all modalities are trained end-to-end from scratch. In this paper, we mainly focus on the first type which is more commonly used, possibly due to its training efficiency which avoids the need for pre-training LLMs. In addition to multi-stage training, some works have explored the benefits of single-stage training, such as SPHINX-X (Liu et al., 2024a) and InstructSeg (Wei et al., 2024). However, the primary motivation behind SPHINX-X is to simplify the process by eliminating the intensive effort of assigning tunable parameters and dataset combinations to different stages. Furthermore, neither SPHINX-X nor InstructSeg provides the detailed comparison of the impacts on downstream tasks with different training strategies that we present. Additionally, two-stage training remains the mainstream approach for domain-specific training of MLLMs.

B Seed Data Construction and Distribution

We convert the combination of VisionFLAN (Xu et al., 2024) and ALLaVA (Chen et al., 2024a) into our required format. Each seed data example consists of an image-caption pair as the input and a related task triplet as the output, which includes an instruction, an informative response, and a precise response. VisionFLAN is a visual instruction task dataset containing 191 tasks, each with 1K examples. ALLaVA builds on VisionFLAN by generating a caption for each image and regenerat-

ing a response for each instruction. In our format, the image, instruction, and response from Vision-FLAN are used as the image, instruction, and precise response, respectively, while the caption and re-generated response from ALLaVA are used as the caption and informative response. Benefiting from the diversity of existing datasets, our seed data encompass a wide range of image domains and task types, as shown in Figure 5.

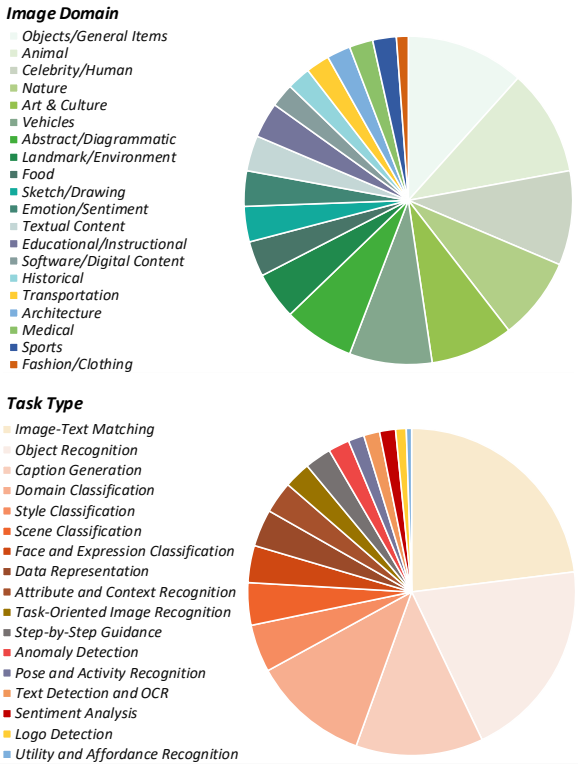


Figure 5: **Distribution of Image Domains and Task Types in Seed Data.**

C Image-Caption Data Source

For the biomedicine domain, we use two datasets from PubMed Central under the `MIT License`: (1) PMC^{Raw} (Zhang et al., 2023a), which comprises 470K publicly available images with human-annotated captions, and (2) PMC^{Refined}, which contains 510K image-caption pairs with captions refined by an MLLM. For the food domain, we collect 130K single-image examples from Recipe1M (Salvador et al., 2017), which is licensed under `CC-BY-4.0`. For the remote sensing domain, we collect 40K image-caption pairs from NWPU-Captions (Cheng et al., 2022), RSICD (Lu et al.), RSITMD (Yuan et al., 2022), Sydney-Captions (Qu et al., 2016), and UCM-Captions (Qu et al., 2016).

D Implementations and Costs of Visual Instruction Synthesis

Table 10 presents the hyper-parameters used for synthesizer tuning. We employ the vLLM inference framework (Kwon et al., 2023) to speed up task synthesis and consistency checks. On a single A100-80GB GPU, it takes approximately 10 hours to synthesize task triplets and an additional 2.5 hours to perform consistency-based filtering for every 100K image-caption pairs. On average, about 30% of the task triplets are reserved after filtering. Specifically, we collect 150K, 144K, and 32K, 15K instruction-response pairs for PMC^{Raw}, PMC^{Refined}, Recipe1M and remote sensing image-caption pairs, respectively.

Hyper-Parameter	Assignment
Base Model	LLaVA-v1.6-8B
Trainable	Full Model
Epoch	2
Batch Size	128
Max Seq Length	6144
$LR_{\text{projector \& LLM}}$	2e-5
$LR_{\text{visual encoder}}$	2e-6
LR Scheduler	Cosine
Weight Decay	0
Warm-Up Ratio	0.03
Computing Infrastructure	8 A100-80GB GPUs
Training Time	13 Hours

Table 10: **Hyper-Parameters for Synthesizer Tuning**

MLLM	LLaVA-v1.6	Qwen2-VL	Llama-3.2
Trainable	Full Model	Full Model	Full Model
Epoch	1	1	1
Batch Size	128	128	128
Max Seq Length	6144	6144	6144
$LR_{\text{projector \& LLM}}$	2e-5	1e-5	5e-6
$LR_{\text{visual encoder}}$	2e-6	1e-5	5e-6
LR Scheduler	Cosine	Cosine	Cosine
Weight Decay	0	0.1	0.1
Warm-Up Ratio	0.03	0.1	0.1

Table 11: **Hyper-Parameters for MLLM Single-Stage Post-Training.**

Image-Caption	PMC^{Raw}	PMC^{Refined}	Recipe1M	Remote.
LLaVA-v1.6-8B	21	23	6	2
Qwen2-VL-2B	3.5	4	1	0.5
Llama-3.2-11B	29	31	9	3

Table 12: **Training Time (Hours) for MLLM Single-Stage Post-Training** on 8 A100-80GB GPUs.

E MLLM Post-Training Settings and Costs

Tables 11 and 12 present the hyper-parameters and training time for the single-stage post-training of MLLMs. In the two-stage training experiments, we employ two common approaches for the first stage on image-caption pairs: (1) unfreezing only the vision-language projector (Liu et al., 2024c) for LLaVA-v1.6-8B, and (2) unfreezing the full model (Chen et al., 2023) for Qwen2-VL-2B and Llama-3.2-11B. During the second stage on visual instruction tasks, we unfreeze the full model across all setups. The hyperparameters for both stages on Qwen2-VL-2B and Llama-3.2-11B are the same as those listed in the table. For LLaVA-v1.6-8B, the first stage differs only in the trainable module, which is the vision-language projector, using a learning rate of 2e-3. The hyper-parameters for the second stage are the same as those listed in Table 11. We use the code implementations from (Chen and Xing, 2024) for experiments on LLaVA-v1.6-8B and from (Zheng et al., 2024) for experiments on Qwen2-VL-2B and Llama-3.2-11B.

F Further Analysis on Fine-tuning with Blank Images

In our initial synthesizer fine-tuning using 100% intact image-caption pairs, we observe two key limitations when testing on out-of-distribution images: (1) the synthesizer generates overly simple questions with low task diversity, and (2) it largely ignores caption information despite its rich knowledge content. Upon examining the seed data source—VisionFlan (Xu et al., 2024), we find that the task instructions are annotated based solely on images without using captions. This explains why our synthesizer learns to prioritize visual information while neglecting textual input.

To balance the utilization of both modalities, we introduce blank images by replacing 10% of training images. This modification forces the synthesizer to rely on caption information when visual information becomes unavailable. Empirical results (Table 6 in Section 6 and Table 7 in Section 7.1) demonstrate the effectiveness of this strategy. We further validate our design through robustness testing with corrupted inputs. Table 13 reveals that fine-tuning with blank images produces two key benefits: first, it yields significant quality improvements for corrupted images while maintaining comparable performance on intact inputs; second, it preserves task quality for text-corrupted cases. These results demonstrate that our approach successfully balances multimodal utilization while enhancing robustness to challenging visual inputs.

Fine-Tune \ Test	Intact	Corrupt Image	Corrupt Caption
w/o Blank	45.0	32.2	38.9
w/ Blank	44.7	36.1	38.9

Table 13: **Synthetic Task Quality Comparison** between synthesizers fine-tuned with/without blank images (10% replacement).“Intact” uses original image-caption pairs; “Corrupt Image” applies random noise or crops to images; “Corrupt Caption” randomly removes text segments from captions.

G Scoring Criteria for Data Quality

For each synthetic dataset, we sample 200 examples and use the following scoring criteria to evaluate data quality in each aspect. The final scores are rescaled to a 0-100 range for presentation uniformity.

Task Diversity For each instruction-response pair, the annotator selects the most appropriate category from the common vision instruction task types listed below. Once all data samples are annotated, we report the number of distinct task types normalized by the total number of common task types.

- *Domain Classification*: Classifying images into domains like race, animal categories, and environment types.
- *Object Recognition*: Recognizing detailed objects like animal species, car brands, and specific object types.
- *Pose and Activity Recognition*: Identifying specific human poses and activities.
- *Logo Detection*: Detecting and recognizing brand logos.
- *Face and Expression Classification*: Classifying facial attributes by age, gender, and detecting expressions.
- *Scene Classification*: Categorizing images into scene types like beaches, forests, and cities.
- *Sentiment Analysis*: Detecting sentiment in images.
- *Caption Generation*: Generating captions for images, including general and contextual descriptions.
- *Text Detection and OCR*: Recognizing text in images and structured text detection.

- *Image-Text Matching*: Assessing image-text similarity and coherence for multimodal content.
- *Anomaly Detection*: Identifying anomalies in settings like industrial and road scenes.
- *Style Classification*: Classifying images by artistic style and quality.
- *Attribute and Context Recognition*: Detecting image attributes and contexts, such as object presence and temporal classification.
- *Task-Oriented Image Recognition*: Recognizing objects in structured contexts, like weed species and quick-draw sketches.
- *Step-by-Step Guidance*: Recognizing steps in instructional content, like wikihow procedures.
- *Data Representation and Visualization*: Visual QA for charts and chart captioning.
- *Utility and Affordance Recognition*: Detecting object utility or affordance in images.
- *Visual Grounding*: Linking image parts to corresponding words or phrases.
- *Segmentation*: Dividing images into meaningful segments, identifying objects or regions.
- *Visual Storytelling*: Creating narratives based on a series of images.

Domain Knowledge Utilization For each instruction-response pair, the annotator evaluates the extent to which domain-specific knowledge from the image is utilized to complete the task. The scoring follows the criteria below, and we report the average score across all samples.

- 1: The task is totally irrelevant to the image.
- 2: The task is relevant, but the question is mundane and answerable without reviewing the image.
- 3: The task requires reviewing the image, but the question is vague, such as asking for a general caption.
- 4: The task is clear, but the question focuses on only one detail in the image.
- 5: The task is highly relevant to both the details and overall context of the image.

Task Complexity For each instruction-response pair, the annotator assesses task complexity, with higher scores for tasks requiring reasoning and instruction-following abilities, using the criteria below. We report the average score across all samples.

- 1: The task can be easily completed by mimicking part of the caption.
- 2: The task can be easily completed by reviewing the image, such as identifying an obvious object.
- 3: The task requires consideration of the details.
- 4: The task requires complex reasoning on details and overview.
- 5: The task requires complex reasoning and instruction-following abilities, such as returning the answer in a required format.

Response Accuracy For each instruction-response pair, the annotator assesses whether the response correctly addresses the task based on the context, using the following criteria. We report the average score across all samples.

- 1: The response is totally irrelevant to the task instruction.
- 2: The response attempts to address the instruction, but both the reasoning and conclusion are incorrect.
- 3: The reasoning is correct, but the conclusion is incorrect.
- 4: The conclusion is correct, but the reasoning is incorrect.
- 5: Both the reasoning and conclusion are correct.

H Task Evaluation Details

Tables 14 and 15 present the specifications and prompt templates for evaluated tasks in each domain. We conduct zero-shot prompting evaluations on these tasks.

For biomedicine, we follow the evaluation approach of (Li et al., 2024a) for SLAKE, PathVQA, and VQA-RAD, and the method of (Zhang et al., 2023b) for PMC-VQA.

- *SLAKE* (Liu et al., 2021) is a semantically-labeled, knowledge-enhanced medical VQA dataset with radiology images and diverse QA pairs annotated by physicians. The dataset includes semantic segmentation masks, object detection bounding boxes, and covers various body parts. "CLOSED" answers are yes/no type, while "OPEN" answers are one-word or short phrases. We use only the English subset.
- *PathVQA* (He et al., 2020) consists of pathology images with QA pairs covering aspects like location, shape, and color. Questions are categorized as "OPEN" (open-ended) or "CLOSED" (closed-ended).
- *VQA-RAD* (Lau et al., 2018) includes clinician-generated QA pairs and radiology images spanning the head, chest, and abdomen. Questions are categorized into 11 types, with answers as either "OPEN" (short text) or "CLOSED" (yes/no).

- *PMC-VQA* (Zhang et al., 2023b) is larger and more diverse MedVQA datasets, with questions ranging from identifying modalities and organs to complex questions requiring specialized knowledge. All questions are multiple-choice.

For food domain, the task descriptions are as follows:

- *Recipe1M* (Salvador et al., 2017) contains recipe information, including titles, ingredients, and cooking instructions. We evaluate models by taking an image and asking for the recipe name, ingredients, and steps.
- *Nutrition5K* (Thames et al., 2021) comprises real-world food dishes with RGB images and nutritional content annotations. We use the ingredient information to create an ingredient prediction task, where the model generates ingredients from an image.
- *Food101* (Bossard et al., 2014) features images across 101 food categories. We ask the model to classify each image into one of the 101 categories.
- *FoodSeg103* (Wu et al., 2021) includes 103 food categories with images and pixel-wise ingredient annotations. We ask the model to select one or multiple categories from a provided list.

For remote-sensing domain, we follow the evaluation approach of (Zhang et al., 2024).

- *CLRS* (Li et al., 2020) consists of 15,000 remote sensing images divided into 25 scene classes. We ask the model to classify each image into one of the 25 categories.
- *UC Merced* (Yang and Newsam, 2010) is a 21 class land use image dataset. The images were manually extracted from large images from the USGS National Map Urban Area Imagery collection for various urban areas around the country. We ask the model to classify each image into one of the 21 categories.
- *FloodNet* (Rahnemoonfar et al., 2021) is a UAV imagery dataset captured after Hurricane Harvey, with visual question answering that challenges models to detect flooded roads and buildings and distinguish between natural and floodwater.
- *NWPU-Captions* (Cheng et al., 2022) includes 31,500 images with annotated captions. The superiority of it lies in its wide coverage of complex scenes and the richness and variety of describing vocabularies. We evaluate models by taking an image and asking for the caption.

Task	Description	Metric	Test Num
Biomedicine			
SLAKE$_{open}$	VQA	Recall	645
SLAKE$_{clos.}$	Binary classification	Acc	416
PathVQA$_{open}$	VQA	Recall	3,357
PathVQA$_{clos.}$	Binary classification	Acc	3,362
VQA-RAD$_{open}$	VQA	Recall	179
VQA-RAD$_{clos.}$	Binary classification	Acc	272
PMC-VQA	Multi-chioice QA	Acc	2,000
Food			
Recipe1M	Recipe generation	Rouge-L	1,000
Nutrition5K	Ingredient prediction	Recall	507
Food101	Category classification	Acc	25,250
FoodSeg103	Multi-label classification	F1	2,135
Remote Sensing			
CLRS	Scene classification	Acc	15,000
UC Merced	Land-use classification	Acc	21,000
FloodNet	VQA	Acc	11,000
NWPU	Image captioning	Rouge-L	31,500

Table 14: **Specifications of the Evaluated Domain-Specific Task Datasets.**

Task	Instruction	Response
Biomedicine		
SLAKE	`{question}`	`{answer}`
PathVQA	`{question}`	`{answer}`
VQA-RAD	`{question}`	`{answer}`
PMC-VQA	`Question: {question}` `The choices are: {options}`	`{option}`
Food		
Recipe1M	`{question}`	`{recipe}`
Nutrition5K	`What ingredients are used to make the dish in the image?`	`{ingredients}`

Food101	`What type of food is shown in this image?`
	`Choose one type from the following options:`
	`{food type options}`
		`{food type}`

FoodSeg103	`Identify the food categories present in the image.`
	`The available categories are: {options}`
	`Please return a list of the selected food categories,`
	`formatted as a list of names like`
	`[candy, egg tart, french fries, chocolate].`
		`{categories}`

Remote Sensing		

CLRS	`What is the category of this remote sensing image?`
	`Answer the question using a single word or phrase.`
	`Reference categories include: {scene options}`
		`{scene category}`

UC Merced	`What is the category of this remote sensing image?`
	`Answer the question using a single word or phrase.`
	`Reference categories include: {land-use options}`
		`{land-use category}`

FloodNet	`{question}`	`{answer}`
NWPU-Captions	`Please provide an one-sentence caption for the` `provided remote sensing image in details.`	`{caption}`

Table 15: **Prompt Templates of the Evaluated Domain-Specific Task Datasets.**

	SFT	VQA-RAD		SLAKE		PathVQA		PMC-VQA
		OPEN	CLOSED	OPEN	CLOSED	OPEN	CLOSED	
LLaVA-Med (From LLaVA)	✗	28.2	61.4	39.2	52.2	12.3	54.1	-
	✓	61.5	84.2	83.1	85.3	38.0	91.2	-
LLaVA-Med (From Vicuna)	✓	64.4	82.0	84.7	83.2	38.9	91.7	-
LLaVA-Med (BioMed CLIP)	✗	37.8	60.7	39.7	54.3	11.7	49.1	-
	✓	64.8	83.1	87.1	86.8	39.6	91.1	-
PubMedVision (LLaVA-v1.5-8B)	✗	-	63.8	-	74.5	-	59.9	52.7
	✓	-	68.9	-	84.1	-	93.0	57.3
PubMedVision (Yi-1.5-34B)	✗	-	68.1	-	76.9	-	63.5	58.2
PubMedVision (LLaVA-v1.5-7B)	✗	-	57.5	-	57.6	-	57.6	46.3
PubMedVision (Qwen-VL-Chat)	✗	-	54.3	-	66.7	-	57.0	48.4

Table 16: **Original results of LLaVA-Med and PubMedVision** as reported in their papers. The column "SFT" indicates whether the domain-adapted MLLM is further fine-tuned on the training set of the target task.

```
I have an image-related question for which I have received two answers: one is informative, and the
other is precise. Please help me assess whether the precise answer is consistent with the informative
answer.

Below are some examples. For each example, write your judgement as follows:
- Yes: Consistent, meaning the precise answer can be inferred from the informative answer.
- No: Inconsistent, meaning the precise answer cannot be inferred from the informative answer.
- Open: Indicates that the question invites multiple interpretations or that answers can be expressed
through open-ended phrasing, acknowledging complexity, or encouraging personal perspectives. It is also
applicable to questions that require image description, captions, or background knowledge.

## Question: Given an image of something from the kitchen, identify what it could be. The image could be
of cooking tools or items that are used for eating. It could also be used for serving food or storing
it.\nOptions: (a) This image contains a cup (b) This image contains a bottle (c) This image contains a
knife (d) This image contains a wine glass (e) This image contains a spoon (f) This image contains a
bowl (g) This image contains a fork
## Informative Answer: The options provided relate to identifying kitchenware or items related to food
service. From the given list, the most clearly identifiable object in the image is a plastic bottle with
a blue label, which is visible in the background behind the dishes. No cups, knives, wine glasses,
spoons, bowls (in the sense of a single, empty bowl), or forks are distinctly visible. Therefore, the
correct identification from the provided options is:\n(b) This image contains a bottle.
## Precise Answer: (a) This image contains a cup
## Consistent: No

## Question: You are provided with an image which contains two pictures side by side. Your task is to
identify the differences between the two pictures. Separate the differences with a comma.
## Informative Answer: Upon close inspection, the differences between the two images are subtle but
distinct. The shadows cast by the trees and lamp posts differ in length and direction, suggesting a
slight variation in the time or position of the sun when the photos were taken. The arrangement and
presence of cars vary\u2014some are present in one image but not in the other. Architectural details
such as windows, roof lines, and the design on the building exteriors also show discrepancies. The color
of the roof on the small central structure is different in each picture, with one appearing darker than
the other. Lastly, the patterns on the sidewalks, especially the crosswalk lines, are not identical,
indicating a variation in either the perspective or actual changes in the environment.
## Precise Answer: the redish brown car moves slightly.
## Consistent: Open
```

Figure 6: **Prompt Template for Consistency-Based Filter (Part 1)**, continued in Part 2.

```
## Question: Given an image of an animal, identify the kind of animal in the image. The picture could
be of more popular animals that are visible around zoos or are sometimes domesticated at home. They
could also sometimes be found in the wild.\nOptions: (a) This image contains a zebra (b) This image
contains a cow (c) This image contains a horse (d) This image contains an elephant (e) This image
contains a giraffe (f) This image contains a sheep (g) This image contains a cat (h) This image
contains a bear (i) This image contains a bird (j) This image contains a dog
## Informative Answer: The animal in the image can be identified as a dog, specifically a golden
retriever. This determination is based on several visual cues such as the golden coat, the size and
shape of the ears, the facial structure, and the overall body size, which are characteristic of this
breed. Golden retrievers are a popular domesticated breed known for their friendly and intelligent
demeanor. They are commonly kept as pets and are not typically found in zoos or in the wild. The image
does not contain any of the other animals listed in the options (a) through (i).
## Precise Answer: (j) This image contains a dog
## Consistent: Yes

## Question: Question: Is the picture taken outdoors?\nPlease answer the question by analyzing the
scene in the provided image.
## Informative Answer: Analyzing the elements within the image, there are several indicators confirming
that the picture is taken outdoors. The presence of natural elements like grass, trees, and the open
sky, along with architectural features that are not enclosed, such as the adobe-style building with
exterior walls and windows, support this observation. The lighting and shadows cast by the trees are
consistent with natural sunlight, further corroborating the outdoor setting. Moreover, the materials
used in the path and bench are typical of outdoor furnishings, designed to withstand weather conditions.
All these aspects contribute to the conclusion that the photograph was taken in an exterior space,
likely a garden or park associated with the building in the background.
## Precise Answer: yes
## Consistent: Yes

## Question: This task tests your ability to count number of objects. Here is the question \"How many
chairs are in the photo?\"
## Informative Answer: Upon careful observation of the photo, each individual is seated in a distinct
chair. The positioning of the subjects and their engagement with the camera suggests they are occupying
all the chairs present at their table. There are no other chairs discernible in the immediate vicinity
of the table or in the background. Therefore, it can be concluded that there are exactly four chairs in
the photo, corresponding to the four people seen around the table.
## Precise Answer: two
## Consistent: No

## Question: Can you give me a caption and some background knowledge about this image?
## Informative Answer: A detailed caption for this image could read: \"A moment of reflection and
accomplishment, as a retired NBA All-Star sits with the Championship Trophy, surrounded by peers and
the spirit of the game.\" The image is probably taken at an NBA event dedicated to honoring the
achievements of past basketball legends. The trophy next to the man symbolizes the pinnacle of success
in the league, suggesting that he may be one of the accomplished players who has reached this level of
success. The attire of the individuals indicates a formal celebration of basketball history, and the
relaxed atmosphere hints at a session of storytelling or interviews about their experiences in the
sport.\"
## Precise Answer: Moses Malone was selected by the New Orleans Jazz with the first pick. On December 9,
1975, the NBA planned to host a supplementary draft to settle negotiating rights to five ABA players
who had never been eligible for the NBA draft because their college classes had not graduated and they
had not apply for hardship. The teams selected in reverse order of their winloss record in the previous
season. The team that made a selection must withdraw their equivalent selection in the 1976 Draft. The
teams were allowed to not exercise their rights on this hardship draft and thus retained their full
selection in the 1976 Draft. The draft itself attracted strong opposition from the ABA who accuse the
NBA trying to reduce confidence in the stability of their league. Despite the initial postponement of
the draft, the draft was finally held on December 30, 1975.
## Consistent: Open

## Question: {Instruction}
## Informative Answer: {Informative Response}
## Precise Answer: {Precise Response}
## Consistent:
```

Figure 7: **Prompt Template for Consistency-Based Filter (Part 2).**

Recipe1M	*Train Pipeline*	Instruction	**Recipe**	**Nutrition**	**Food101**	**FoodSeg**	**AVERAGE**
LLaVA-v1.6-8B	*Two-Stage*	Rule	23.1	29.1	46.8	14.5	28.4
		Ours	16.2	28.3	43.5	28.0	29.0
	Single-Stage	Rule	21.8	36.7	63.9	13.9	34.1
		Ours	24.8	36.1	65.3	42.0	42.0
Qwen2-VL-2B	*Two-Stage*	Rule	24.1	24.5	68.8	7.7	31.3
		Ours	16.5	43.0	69.5	23.9	38.2
	Single-Stage	Rule	19.3	37.1	64.7	6.6	31.9
		Ours	24.0	41.2	72.0	23.9	40.3
Llama-3.2-11B	*Two-Stage*	Rule	25.7	26.2	82.1	16.7	37.7
		Ours	17.8	38.0	74.6	33.2	40.9
	Single-Stage	Rule	21.4	32.2	75.8	16.9	36.6
		Ours	26.1	41.0	82.2	42.0	47.8

PMC^{Raw}	*Train Pipeline*	Instruction	**SLAKE**		**PathVQA**		**VQA-RAD**		**PMC-VQA**	**AVERAGE**
			OPEN	CLOSED	OPEN	CLOSED	OPEN	CLOSED		
LLaVA-v1.6-8B	*Two-Stage*	GPT-4	43.4	50.2	10.1	59.2	35.0	62.5	37.1	42.5
		Ours	56.2	71.4	17.2	74.5	50.6	79.0	40.4	55.6
	Single-Stage	GPT-4	44.2	59.1	11.6	62.2	38.5	67.3	39.9	46.1
		Ours	56.8	76.4	19.7	79.3	51.0	80.5	44.3	58.3
Qwen2-VL-2B	*Two-Stage*	GPT-4	43.4	55.5	11.8	60.1	37.1	58.8	41.2	44.0
		Ours	55.2	74.5	18.4	68.4	48.8	79.8	43.8	55.5
	Single-Stage	GPT-4	43.6	59.6	13.2	47.4	37.3	57.0	31.2	41.3
		Ours	53.2	75.2	20.1	63.8	49.8	74.6	43.5	54.3
Llama-3.2-11B	*Two-Stage*	GPT-4	47.6	58.7	14.6	69.5	38.0	69.1	47.5	49.3
		Ours	60.0	75.7	22.1	76.8	51.4	80.5	47.9	59.2
	Single-Stage	GPT-4	46.8	56.5	16.0	69.9	41.9	65.4	45.3	48.8
		Ours	56.7	77.6	22.2	87.3	55.0	76.1	49.9	60.7

PMC^{Refined}	*Train Pipeline*	Instruction	**SLAKE**		**PathVQA**		**VQA-RAD**		**PMC-VQA**	**AVERAGE**
			OPEN	CLOSED	OPEN	CLOSED	OPEN	CLOSED		
LLaVA-v1.6-8B	*Two-Stage*	GPT-4V	50.0	68.3	17.0	67.5	43.3	67.3	40.4	50.5
		Ours	54.8	73.1	19.3	79.7	55.6	82.7	45.1	58.6
	Single-Stage	GPT-4V	52.3	76.2	20.1	73.3	47.0	76.5	43.1	55.5
		Ours	58.0	73.3	22.9	78.6	59.8	81.3	47.9	60.3
Qwen2-VL-2B	*Two-Stage*	GPT-4V	45.2	63.2	18.2	64.7	41.3	67.3	43.2	49.0
		Ours	60.8	76.9	21.4	75.0	55.0	82.7	44.7	59.5
	Single-Stage	GPT-4V	51.4	66.1	18.9	61.4	45.1	73.2	45.1	51.6
		Ours	60.2	75.0	20.6	53.6	58.0	76.1	46.5	55.7
Llama-3.2-11B	*Two-Stage*	GPT-4V	49.1	74.3	19.3	70.9	46.2	73.9	47.1	54.4
		Ours	58.5	76.4	27.0	73.2	58.3	77.6	51.3	60.3
	Single-Stage	GPT-4V	47.1	72.6	19.5	70.7	45.9	73.9	46.5	53.7
		Ours	59.5	76.4	24.3	84.9	57.4	79.8	51.9	62.0

Table 17: **Domain-Specific Task Performance of MLLMs after Post-Training** with different synthetic data and training pipelines. The image-caption sources are Recipe1M, PMC^{Raw} and PMC^{Refined}, respectively. In most cases using our synthetic data, we find that single-stage training outperforms two-stage training on domain-specific tasks, particularly evident in the **Recipe** generation results for the food domain. Recall that in the two-stage training approach for the food domain, the model first trains on recipe captions and then on our synthetic tasks. We examine the task performance of LLaVA-v1.6-8B on **Recipe** generation and observe that the model achieves a score of 25.3 after the first stage on recipe captions. However, this score drastically decreases to 16.2 after the second stage. From this, we infer that the two-stage approach causes the model to catastrophically forget the task/knowledge learned in the first stage when transitioning to the second stage (Luo et al., 2023b), leading to poorer performance after completing the second-stage training.

Caption

To make Evie'S Ranch Nuggets, you'll need these ingredients: (1). 2 pounds skinless, boneless chicken breasts, cut into 2 inch pieces (2). 1/2 cup sour cream (3). 1/2 cup all-purpose flour (4). 4 ounces Ranch-flavored tortilla chips (5). vegetable cooking spray
Follow these steps to cook it:
[+] Preheat the oven to 400 degrees F (200 degrees C). Line a baking sheet with aluminum foil.
[+] Place tortilla chips and flour into a food processor or blender, and process into crumbs. Pour into a bowl. Place chicken pieces into a sealable plastic bag with sour cream. Massage the bag to make sure chicken pieces are coated with sour cream. Let sit for 5 minutes.
[+] Dip coated chicken pieces into the chip crumbs, and place on the prepared baking sheet. Spray lightly with vegetable cooking spray.
[+] Bake for 30 minutes in the preheated oven, until chicken is cooked through and juices run clear.

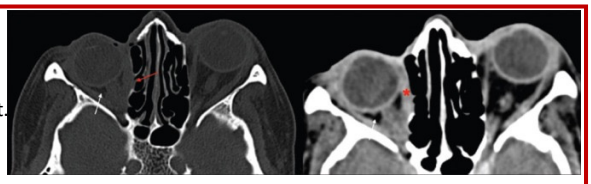
Rule-Based Instruction-Response Pair

Instruction: I'm curious! What's the name, ingredients, and cooking method for this mouth-watering creation?
Response: To make Evie'S Ranch Nuggets, you'll need these ingredients: (1). 2 pounds skinless, boneless chicken breasts, cut into 2 inch pieces (2). 1/2 cup sour cream (3). 1/2 cup all-purpose flour (4). 4 ounces Ranch-flavored tortilla chips (5). vegetable cooking spray
Follow these steps to cook it:
[+] Preheat the oven to 400 degrees F (200 degrees C). Line a baking sheet with aluminum foil. (...)

Our Instruction-Response Pair

Instruction: Texture is defined as the feel, appearance or consistency of a surface or substance from a human's perspective. Detect the primary texture represented in the image.
Response: The primary texture represented in the image is that of a crispy, crunchy exterior typical of fried foods. The uneven, rough surface of the chicken nuggets suggests a firm texture that would be audible upon bite. The golden-brown color indicates a Maillard reaction, which is a chemical reaction between amino acids and reducing sugars that gives browned food its distinctive flavor. This reaction also contributes to the crunchy texture, as it creates a rigid outer layer. The visible crumbs and the rough, uneven surface further emphasize the crunchy texture that one would expect from such a food item.

Caption



CT scan of the orbits revealed right eye enophthalmos, focal osseous defect of lamina papyracea (red arrow), deficient orbital fat (white arrow), fibrosis of soft tissue (red star) along medial wall of right orbit.

GPT-4 Instruction-Response Pair

Instruction: What is the red arrow pointing to?
Response: The red arrow in the CT scan is pointing to a focal osseous defect of the lamina papyracea.

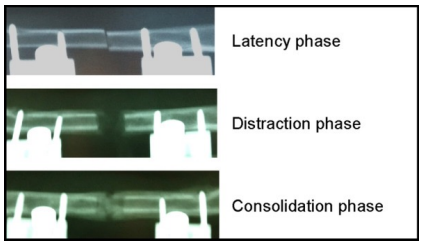
Our Instruction-Response Pair

Instruction: What is the location of the lesion?
Response: The lesion is located in the right orbit. On the CT scan, it appears as an area of increased density (hyperdense) relative to the surrounding brain tissue. The lesion is situated medial to the eyeball, which is the most prominent structure within the orbit. It is important to note that the interpretation of radiological images should be performed by a qualified radiologist, as it requires a detailed understanding of human anatomy and the ability to correlate findings with clinical symptoms for an accurate diagnosis. The answer is right orbit.

Figure 8: **Cases of Instruction-Response Pairs (Part 1)** synthesized by our method, manual rules, GPT-4, and GPT-4V, the image-caption sources for the cases are Recipe1M, PMC^{Raw} and PMC^{Refined}, respectively. Continued in Part 2. In the first case, the rule-based task simply transforms the recipe caption, ignoring the image content. In contrast, our task involves analyzing the food's state in the image and applying food-related knowledge to infer its texture, demonstrating a higher level of domain knowledge utilization. In the second case, the GPT-4 generated task straightforwardly asks about the pointing of the red arrow, while ours requires a detailed analysis and inference, showing greater task complexity.

Caption

The image displays three distinct panels, each labeled with a different phase of bone development under distraction osteogenesis. The top panel, labeled 'Latency phase', shows an X-ray image of a bone with a surgical device composed of two parallel metal plates attached by screws on either side of a bone segment. The middle panel, labeled 'Distraction phase', depicts the same bone but with a noticeable increase in the gap between the bone ends, suggesting bone lengthening. The bottom panel, labeled 'Consolidation phase', illustrates further progress of bone growth within the gap, indicating new bone formation and consolidation. Each panel uses the same angle and lighting conditions to emphasize the changes in the bone structure across different phases.



GPT-4V Instruction-Response Pair

Instruction: Can you identify the type of surgical hardware used in these phases based on the image, and explain its role in the manipulation of bone segments during distraction osteogenesis?

Response: The surgical hardware visualized in the image appears to be a type of external fixator, consisting of two parallel metal plates connected by screws that penetrate through the bone segments. This hardware plays a pivotal role in distraction osteogenesis by stabilizing the bone while also allowing for gradual bone lengthening. The screws can be adjusted to manipulate the distance between the divided bone segments, facilitating new bone growth in the process required by the distraction phase.

Our Instruction-Response Pair

Instruction: What is the gap between the bone ends called?
The choices are: (A) Consolidation; (B) Distraction; (C) Latency; (D) None of the above.

Response: The gap between the bone ends, as depicted in the middle panel of the image, is referred to as the 'Distraction phase'. This phase is characterized by the lengthening of the bone due to the application of a distraction device, which separates the bone ends and allows for new bone growth to fill the gap. Therefore, the correct answer to the question is:
(B) Distraction.

Figure 9: **Cases of Instruction-Response Pairs (Part 2)** synthesized by our method, manual rules, GPT-4, and GPT-4V, the image-caption sources for the cases are Recipe1M, PMC^{Raw} and PMC^{Refined}, respectively. In this case, our task stands out as a multiple-choice question, showcasing task diversity.