Task Navigator: Decomposing Complex Tasks for Multimodal Large Language Models

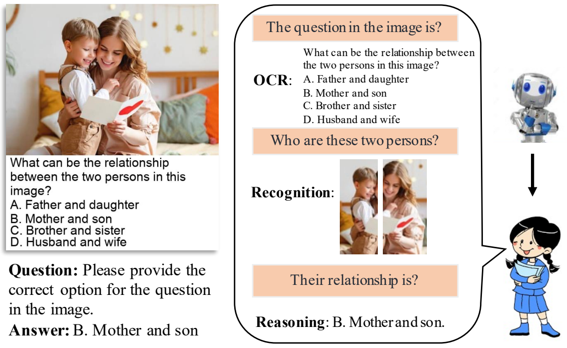
Feipeng Ma[1*], Yizhou Zhou[2], Yueyi Zhang[1†], Siying Wu[3]
Zheyu Zhang[1], Zilong He[1], Fengyun Rao[2], Xiaoyan Sun[1,3†]
[1]University of Science and Technology of China [2]WeChat, Tencent Inc.
[3]Institute of Artificial Intelligence, Hefei Comprehensive National Science Center
{mafp,zhangzy0,hezil}@mail.ustc.edu.cn {harryizzhou,fengyunrao}@tencent.com
wsy315@iai.ustc.edu.cn {zhyuey,sunxiaoyan}@ustc.edu.cn

Abstract

Inspired by the remarkable progress achieved by recent Large Language Models (LLMs), Multimodal Large Language Models (MLLMs) take LLMs as their brains, and have achieved surprising results in many downstream tasks by training on a large amount of task-specific data. However, when faced with complex tasks that require the collaboration of multiple capabilities, existing MLLMs recollect training data and retrain the model, ignoring the systematic utilization of LLMs and their possessed capabilities learned in downstream tasks. Inspired by the way humans tackle complex questions, in this paper, we propose a novel framework called Task Navigator. In our framework, LLMs act as navigators to chart a viable path for solving complex tasks and guide MLLMs through the process step by step. Specifically, LLMs iteratively break down sub-problems and refine them to be more reasonable and answerable, which are subsequently resolved by MLLMs to obtain relevant sub-answers, until the LLMs have collected enough information to answer the initial question. Task Navigator provides an effective way to extend MLLMs to tackle complex tasks, thus broadening MLLMs' applicability. To evaluate the performance of the proposed framework, we have curated a carefully designed benchmark called VersaChallenge. Experiments on VersaChallenge demonstrate the effectiveness of our proposed method.

1. Introduction

Recently, Large Language Models (LLMs) have attracted substantial public attention. By scaling up the training corpus and model size, LLMs [15, 24, 25, 38] have achieved



What can be the relationship between the two persons in this image?
A. Father and daughter
B. Mother and son
C. Brother and sister
D. Husband and wife

Question: Please provide the correct option for the question in the image.
Answer: B. Mother and son

The question in the image is?

OCR: What can be the relationship between the two persons in this image?
A. Father and daughter
B. Mother and son
C. Brother and sister
D. Husband and wife

Who are these two persons?

Recognition:

Their relationship is?

Reasoning: B. Mother and son.

Figure 1. An illustration of the procedure for decomposing a complex question into several corresponding sub-questions. MLLMs deduce the final answer by using their possessed capabilities to progressively solve each disassembled sub-questions.

exciting performance on most of Natural Language Processing (NLP) tasks. Using LLMs as brains, Multimodal Large Language Models (MLLMs) take one step towards developing Artificial General Intelligence (AGI) by allowing input of different modalities, which is more consistent with the way humans perceive the world.

Existing MLLMs [1, 3, 10, 39] perform pre-training or fine-tuning on task-related large-scale paired training data, which enables MLLMs to acquire task-specific capabilities, including image captioning, Visual Question Answering (VQA), Optical Character Recognition (OCR) [35], object detection [26], segmentation [26], and grounding [1, 3, 18]. For instance, fine-tuning MLLMs on COCO dataset [8] can enable the model to obtain the ability of image captioning, that is, to generate a sentence to describe the salient content of the input image. However, every time MLLMs encounter a complex task that requires multifarious capabilities to address, it is labor- and financial-intensive to solve it by collecting a large-scale corresponding dataset and retraining

[*] This work was performed while Feipeng Ma was an intern with WeChat, Tencent Inc.

[†] Corresponding authors: {zhyuey, sunxiaoyan}@ustc.edu.cn

the model. In this paper, we simulate the way humans solve complex problems to tackle the challenges mentioned above. Consider an example shown in Figure 1, given the question "*Please provide the correct option for the question in the image*", humans first need to recognize the text written in the image, "*What can be the relationship between the two persons in this image*", by using their OCR capability, which further requires humans to leverage their Object Detection capability for locating the two persons mentioned in the question. Combining the results of OCR and object detection, humans deduce the correct answer to the initial question is: "*B. Mother and son*".

To enable MLLMs to decompose complex tasks like humans and flexibly utilize their visual capabilities to solve the sub-questions, we introduce a new framework called *Task Navigator*. This framework enables a collaborative approach between LLMs and MLLMs, focusing on complex tasks that involve images and questions. In this setup, LLMs act as navigators, responsible for charting a viable path for solving complex tasks. MLLMs follow the chart formulated by LLMs, utilizing their full capabilities to accomplish the complex task. Initially, LLMs analyze the original question to generate a relevant sub-question. Subsequently, MLLMs are required to provide an answer of the sub-question. To ensure the feasibility and answerability of the sub-question, we incorporate a refinement process. If a sub-question is deemed unsuitable, LLMs will formulate an alternative, considering previous sub-questions and answers. MLLMs are also responsible for answering this alternative sub-question. LLMs then review the accumulated history, including the original question, the sub-questions, and their answers, to craft the next sub-question. This iterative process continues until LLMs determine that they have collected sufficient information from MLLMs. Finally, LLMs summarize the sub-questions and corresponding answers to generate the desired response for the initial complex question.

To evaluate the performance of MLLMs on complex tasks, we introduce *VersaChallenge*, a carefully designed benchmark for MLLMs on complex tasks. We sample images from various sources, design different tasks that require multifarious abilities to solve, and employ manual annotation. The *VersaChallenge* exhibits three salient characteristics: Firstly, it demonstrates considerable task heterogeneity, encompassing eight distinct sub-tasks. Secondly, it demands collaborative capabilities, especially necessitating the integrated application of visual and cognitive skills. Lastly, there is a pronounced dependence on visual information, imperative for the interpretation and recognition of the presented image. The results on *VersaChallenge* showcase the effectiveness of our proposed method. Extensive experiments demonstrate that our method can easily generalize to various domains and provide reliable predictions without hallucination.

Our main contributions are three-fold:

- We propose a novel framework called *Task Navigator*, which allows MLLMs to tackle unprecedented complex tasks by leveraging the embedded LLMs and harnessing their acquired capabilities.
- We collect *VersaChallenge*, to evaluate the capabilities of MLLMs in addressing intricate tasks.
- Extensive experiments demonstrate our proposed method achieve the state-of-the-art performance on *VersaChallenge*.

2. Related Work

Multimodal Large Language Models. Multimodal Large Language Models are developed from LLMs, and usually consist of a vision encoder, alignment network, and LLMs. MiniGPT-4 [39] and LLaVA [10] make the first try to develop MLLMs based on LLMs. Liu et al. [10] connect the visual encoder of CLIP [20] with the language decoder LLaMA. Further research explores the MLLMs' capabilities on various tasks. Kosmos-2 [18] and Shikra [3] unleash the grounding ability of MLLMs by constructing grounding image-text pairs Zhang et al. [35] collect high-quality instruction-following data to improve the OCR ability of MLLMs. This line of work demonstrates that MLLMs, trained on specifically designed data, can learn various abilities to solve different tasks. Although MLLMs can acquire diverse abilities, they encounter challenges in effectively integrating these abilities they owned to address complex tasks that demand multifarious capabilities.

LLM-Aided Visual Reasoning. Recently, tool-augmented LLMs have been explored to combine with visual experts to solve visual reasoning tasks. MMReact [29] leverages LLMs' planning and reasoning abilities to allocate multiple vision experts, including image captioning, tagging, OCR, etc, to address problems in different scenarios. Visual ChatGPT [28] also incorporates different visual foundation models to extend ChatGPT to solve complex visual questions. These methods usually require multiple experts, leading to high deployment costs, the need for carefully designed prompts, and extra processing of input and output forms for different experts. Our method does not require vision experts, the MLLMs are the only visual interface for all visual information requirements.

Chain-of-Thought. Previous research [27] demonstrates that generating a series of intermediate reasoning steps, known as Chain-of-Thought (CoT), can improve the reasoning capabilities of LLMs. Zero-shot CoT [7] ignites CoT reasoning with an added prompt "Let's think step by step". Few-shot CoT [21, 27] leverages a handful of reasoning demonstrations to achieve superior results. Other methods consider diverse reasoning prompts. Khot et al. [6] propose decomposed prompting to solve complex tasks.

Yao et al. [30] develop Tree-of-Thought to enable LLMs to make deliberate decisions by considering various possible reasoning paths. Further study [36, 37] extends CoT reasoning into the multimodal domain. However, applying CoT in the multimodal domain poses a significant challenge, as it requires MLLMs to deeply understand both textual and visual inputs. Inspired by previous studies, we propose *Task Navigator* to disentangle reasoning and visual understanding for complex tasks.

Evaluation of Multimodal Large Language Models. Multimodal Large Language Models have shown great performance on various traditional vision and language tasks, but effectively evaluating them can be a challenging problem. Yin et al. [32] propose a benchmark on various 2D and 3D tasks, including 9 image tasks and 3 point cloud tasks. MME [4] and MMBench [11] both measure perception and cognition abilities. MME measures MLLMs on 14 sub-tasks, with the question type designed as true of false to avoid the impact of prompt engineering. MMBench contains approximately 3000 questions covering 20 abilities. Our *VersaChallenge* focuses on complex tasks that require combining various capabilities to solve.

3. Method and Benchmark

In this paper, we present a novel framework named *Task Navigator* to extend MLLMs to tackle complex tasks that require various capabilities. *Task Navigator* utilizes LLMs to decompose the complex questions into visual-related sub-questions and integrates a refining process to guarantee the feasibility and answerability of sub-questions. With the sub-answers obtained from MLLMs, LLMs can successfully provide the correct answer for complex tasks. The following is organized as follows. We first present the details of our *Task Navigator*. Then, we introduce *VersaChallenge* benchmark for evaluating the performance of MLLMs on unseen tasks.

3.1. Task Navigator

Although current MLLMs have demonstrated impressive performance in a range of tasks, such as image captioning, object detection and OCR, they still face challenges when it comes to complex tasks that require a combination of their visual and reasoning capabilities. This limitation in addressing problems within their expected scope significantly restricts their practical applications. A straightforward solution would be to enrich the training dataset with more examples covering all possible combinations of visual and reasoning skills. However, collecting such comprehensive data is not only impractical but also incurs significant training costs.

Our research takes note that current MLLMs developed by LLMs do not fully leverage the capabilities of LLMs. Current MLLMs only adopt LLMs as a language decoder and overlook the potential capability of LLMs. Compared with LLMs, which are trained on trillion tokens from various corpora [24, 25], MLLMs's training data is limited in scale and diversity. For example, Qwen-VL [1] are trained on about 1.5 billion image-text pairs [2, 16, 22, 23], and 95% of them are image captioning data. The large training corpus of LLMs endows them with strong generalization capabilities, enabling them to tackle diverse tasks in real-world scenarios. However, simply concatenating a powerful LLM with a visual encoder and fine-tuning it on limited text-image data can compromise the LLM's capabilities. Recent research [1, 9, 12, 34] notice this issue and incorporate language-only instruction data during training to balance MLLMs' language and multimodal abilities. However, this only mitigates the degradation of LLM's capabilities, rather than fully harnessing its powerful potential.

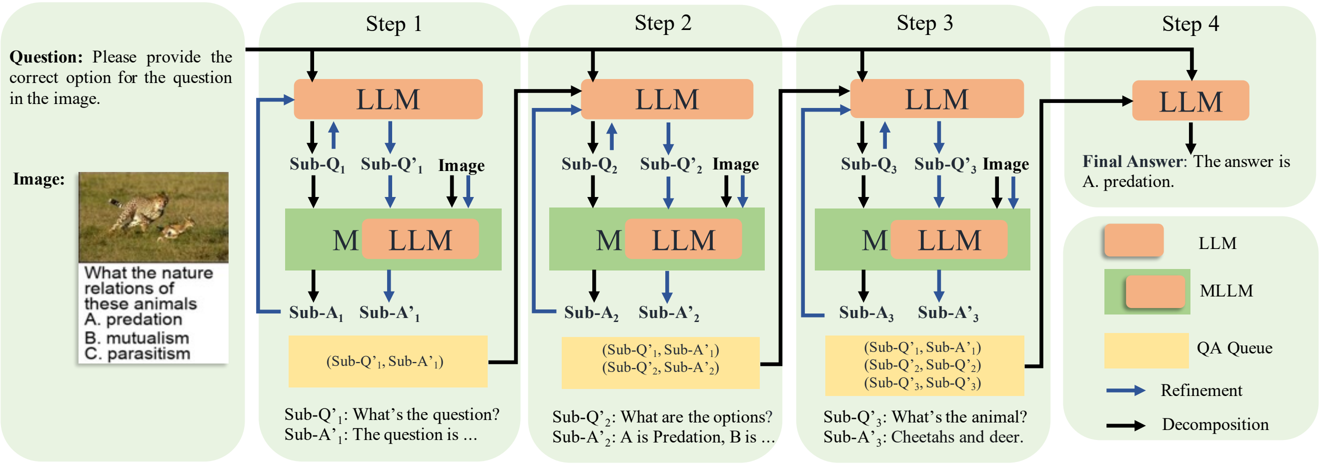
Inspired by the way humans tackle complex problems, the key of our approach is to rely on powerful and generalizable LLMs to analyze the problem and decompose the complex question into sub-questions. As shown in Figure 2, given an input image X_v and a complex question Q that require multifarious capabilities, we employ the MLLM $g_{\phi,\theta}(\cdot)$, which is built upon LLM $f_\phi(\cdot)$ parameterized by ϕ and the LLM $f_{\phi'}(\cdot)$ parameterized by ϕ', in which $\phi' = \phi$. At the first step, LLM receives Q and a designed prompt P to generate the first sub-question $Q_1 = f_{\phi'}(Q; P)$. The sub-question Q_1 will be answered by MLLMs to obtain the corresponding answer $A_1 = g_{\phi,\theta}(Q_1; X_v)$. To ensure the feasibility and answerability of sub-question Q_1, we incorporate a refinement process $Q'_1 = f_{\phi'}((Q_1, A_1), Q; R)$ with refinement prompt R. The details of prompt P and R are shown in the supplementary material. The refined sub-question will be answered by MLLM, $A'_1 = g_{\phi,\theta}(Q'_1; X_v)$. Then for step i, LLM will continue to pose a new sub-question based on the previous information, including previous sub-questions and the corresponding answers, and refine it, this process is formulated as:

$$Q_{i+1} = f_{\phi'}((Q'_i, A'_i), ..., (Q'_1, A'_1), Q; P), \tag{1}$$

$$A_{i+1} = g_{\phi,\theta}(Q_{i+1}; X_v), \tag{2}$$

$$Q'_{i+1} = f_{\phi'}((Q_{i+1}, A_{i+1}), (Q'_i, A'_i), ..., (Q'_1, A'_1), Q; R). \tag{3}$$

The refined sub-question Q'_{i+1} and image X_v are fed into MLLM to obtain the corresponding answer $A'_{i+1} = g_{\phi,\theta}(Q'_{i+1}; X_v)$. Our *Task Navigator* employs this dialogue-based approach to question decomposition, distinguishing itself from methods that directly decompose the question at once. This iterative dialogue process is crucial because the LLM does not have direct visual perception capabilities and may lack comprehensive prior knowledge. Therefore, relying solely on the initial question to generate sub-questions could lead to the omission of essential information. Through this iterative dialogue, the *Task Navigator* progressively refines its understanding, ensuring that all necessary information is captured to formulate the final answer.



Question: Please provide the correct option for the question in the image.

(Sub-Q'$_1$, Sub-A'$_1$)

Sub-Q'$_1$: What's the question?
Sub-A'$_1$: The question is …

(Sub-Q'$_1$, Sub-A'$_1$)
(Sub-Q'$_2$, Sub-A'$_2$)

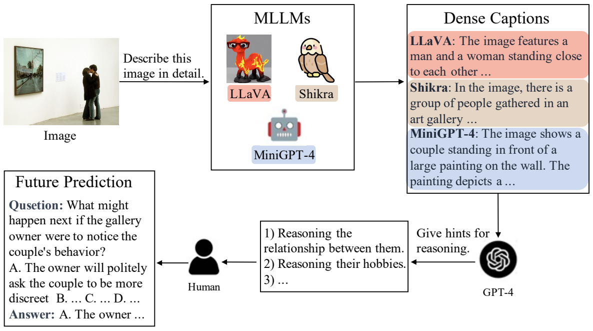
Sub-Q'$_2$: What are the options?
Sub-A'$_2$: A is Predation, B is …

(Sub-Q'$_1$, Sub-A'$_1$)
(Sub-Q'$_2$, Sub-Q'$_2$)
(Sub-Q'$_3$, Sub-Q'$_3$)

Sub-Q'$_3$: What's the animal?
Sub-A'$_3$: Cheetahs and deer.

Final Answer: The answer is A. predation.

Figure 2. **The pipeline of Task Navigator.** This figure provides a detailed explanation of the process of Task Navigator. We utilize the inner LLM from the MLLM. LLM acts as a navigator to sequentially break down complex tasks into visual-related sub-questions and refine them, which are then addressed by the MLLM. Specifically, a) in steps 1 and 2, the MLLM leverages its OCR ability to identify the question and options presented in the image. b) In step 3, MLLM employs its recognition ability to identify the animals in the image. c) In step 4, the LLM has gathered enough information and summarizes the sub-questions and the corresponding answers to generate the final answer.



LLaVA: The image features a man and a woman standing close to each other …
Shikra: In the image, there is a group of people gathered in an art gallery …
MiniGPT-4: The image shows a couple standing in front of a large painting on the wall. The painting depicts a …

1) Reasoning the relationship between them.
2) Reasoning their hobbies.
3) …

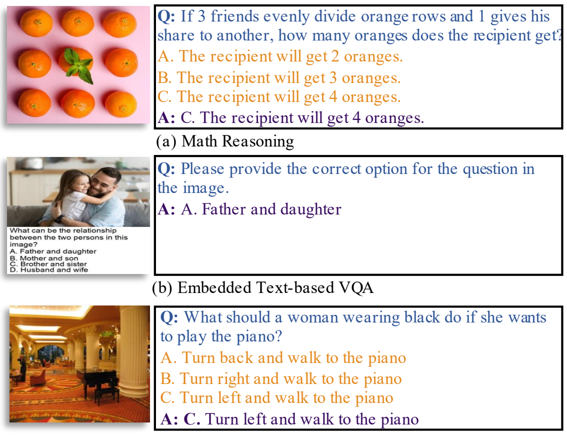
Qusetion: What might happen next if the gallery owner were to notice the couple's behavior?
A. The owner will politely ask the couple to be more discreet B. … C. … D. …
Answer: A. The owner …

Figure 3. **The pipeline of labeling benchmark.** In this figure, we illustrate our annotation process. We collect images from existing public datasets and prompt current MLLMs to generate dense captions for each image. Subsequently, we utilize GPT-4 to provide hints for human annotation, drawing on the dense captions and in-context examples designed for each task. Finally, human annotators compose questions that require reasoning and visual abilities, guided by the valuable hints provided by GPT-4.

Additionally, in our framework, the LLM can come from different sources:

$$\phi' = \begin{cases} \phi, & \text{Inner LLM} \\ \hat{\phi}, & \text{Fine-tuned Inner LLM} \\ \Omega, & \text{External LLM} \end{cases} \tag{4}$$

where $\phi' = \phi$ represents that we utilize the inner LLM of the MLLM, $\phi' = \hat{\phi}$ indicates that the fine-tuned inner LLM is employed, and $\phi' = \Omega$ means that we adopt an external LLM. The diverse sources from which the employed LLMs are drawn greatly enhance the flexibility of our framework, enabling it to be applied across a variety of scenarios.



Q: If 3 friends evenly divide orange rows and 1 gives his share to another, how many oranges does the recipient get?
A. The recipient will get 2 oranges.
B. The recipient will get 3 oranges.
C. The recipient will get 4 oranges.
A: C. The recipient will get 4 oranges.

(a) Math Reasoning

Q: Please provide the correct option for the question in the image.
A: A. Father and daughter

(b) Embedded Text-based VQA

Q: What should a woman wearing black do if she wants to play the piano?
A. Turn back and walk to the piano
B. Turn right and walk to the piano
C. Turn left and walk to the piano
A: C. Turn left and walk to the piano

(c) Visual Planning

Figure 4. **Examples for three tasks we designed.** In this figure, we present examples for three tasks, namely Math Reasoning, Embedded Text-based VQA, and Visual Planning. Math reasoning requires recognition, counting, and reasoning abilities. Embedded Text-based VQA demands OCR and other abilities. Visual planning requires localization and planning capabilities.

3.2. VersaChallenge Benchmark

To evaluate the performance of MLLMs on complex tasks that require various capabilities, we introduce the *VersaChallenge* benchmark. The benchmark consists of 8 tasks, with 5 designed for general scenarios, and 3 designed for specific scenarios.

Tasks for general scenarios include commonsense reasoning, physical relation reasoning, physical property reasoning, future prediction, and functional reasoning, inspired by MME [4] and MMBench [11]. The requirements for these tasks are two-fold: 1) The questions necessitate visual information to address. 2) The questions require combining reasoning abilities with various visual abilities to solve. However, annotating visual questions that require various abilities encounters two challenges: 1) Collecting images for specific reasoning categories is challenging. 2) Annotating reasoning problems requires imagination, placing high demands on annotators. To address these challenges, we leverage GPT-4 [15] to assist in manual annotation. As shown in Figure 3, we collect images from existing public datasets, such as COCO, and prompt current MLLMs to generate dense captions for each image. Then we use GPT-4 to provide hints for human annotation, based on the dense captions and in-context examples that are designed for each task. Finally, the human annotators write questions that necessitate reasoning and visual abilities referring to the valuable hints provided by GPT-4. It is worth noting that while dense descriptions generated by current MLLMs may exhibit hallucination issues, GPT-4 only serves as a guide for annotators to avoid these illusions compromising the quality of annotations.

To further evaluate the MLLMs in solving complex questions that demand various abilities, we design three tasks: Math Reasoning, Embedded Text-based VQA, and Visual Planning. We provide one sample for each task in Figure 4. These tasks are designed with the distinctive feature that the required combination of abilities is rarely encountered in the training data. As a result, models face significant challenges when addressing such problems.

Math Reasoning. The objective of this task is to perform calculations based on the number of objects depicted in the image. We collect images from CountBench [17], each containing 2 to 10 objects. This task necessitates recognition, counting, and calculation abilities. MLLMs are expected to identify the objects, count the specific objects, and execute calculations in response to the questions.

Embedded Text-based VQA. In this task, we directly embed the questions and options into the images. Therefore, the images consist of two parts: the upper half displays a natural scene, while the lower half contains text, presenting the original question and its corresponding options. To tackle this task, MLLMs have to recognize the visual content and the accompanying text in the image, without providing additional information. The model must leverage its OCR ability to answer the questions potentially. For this task, we sample images and questions from the validation set of MMBench [11] to construct images.

Visual Planning. To evaluate the combination of localization and planning abilities of MLLMs, we design a simplified version of the visual planning task. In this task, the question entails a goal and is given several options related to the actions to achieve this goal. The images are sourced from MIT indoor scenes [19], and questions and options are labeled by human annotators.

Considering the impact of the instruction-following ability of current MLLMs, we follow MMBench [11] to formulate the questions as single-choice questions and adopt the CircularEval strategy.

4. Experiments

In this section, we begin with introducing our experimental setup first, including baselines, settings, and metrics. Subsequently, we showcase our *Task Navigator* on *VersaChallenge* to demonstrate the superiority of our method, utilizing both inner LLM and external LLM.

4.1. Experimental Setup

Baselines. We conduct comprehensive comparisons with current MLLMs, including: 1) Shikra [3], which focuses on unleashing MLLMs' referential dialogue ability and can handle location-related tasks. 2) Qwen-VL [1], which utilizes a large-scale web-crawled image-text pair dataset of approximately 1.5 billion pairs. 3) Lynx[33], which is pretrained on more than 120 million image-text pairs and finetuned using 32 datasets. 4) InternLM-XComposer [34], pretrained on 1.1 billion multilingual image-text pairs and finetuned on various public datasets. 5) mPLUG-Owl2 [31], designed with a modularized network to leverage modality collaboration. 6) LLaVA-v1.5 [9], built on LLaVA [10] with simple modifications, such as upscaling image resolutions, using an MLP, and incorporating task-oriented VQA data. 7) GPT4-V [13, 14] represents the most advanced MLLM to date.

Settings. As different MLLMs are trained with various settings, we utilize the default setting for each MLLM to perform inference. In the case of our *Task Navigator*, we employ LLaVA-v1.5 as the MLLM and Vicuna-v1.5 [38] as the inner LLM. It is important to note that LLaVA-v1.5 is built upon Vicuna-v1.5. We also introduce an external LLM, such as GPT-4 [15], to conduct experiments.

Metrics. The questions in our benchmark are single-choice questions, and we report the accuracy of each sub-task as well as the average accuracy across all tasks in our experiments. Following MMBench [11], we employ the CircularEval strategy for evaluation. Specifically, for each single-choice question with N choices, we present it to MLLMs N times with different choice orders. MLLMs are considered to have answered a question correctly only if they provide correct responses for all variants of the question. This approach helps reduce randomness in the evaluation of MLLMs.

Method	LLM	VersaChallenge								
		CR	PR	PP	FP	FR	MR	ETV	VP	Avg.
Shikra [3]	Vicuna-7B	11.49	3.37	5.33	6.10	10.86	2.43	0.00	0.00	3.11
Qwen-VL [1]	Qwen-7B	28.32	4.54	23.61	16.41	25.00	11.68	0.00	1.3	8.83
Lynx [33]	Vicuna-7B	10.40	2.24	10.66	3.81	8.69	1.81	0.00	1.97	3.65
InternLM-XComposer [34]	InternLM-7B	17.24	7.86	14.66	19.23	23.91	4.09	0.00	0.00	6.64
mPLUG-Owl2 [31]	LLaMA2-7B	26.16	16.09	33.33	28.68	10.86	11.87	0.58	0.04	11.17
LLaVA-v1.5 [9]	Vicuna-13B	12.79	14.77	9.33	7.63	8.69	14.02	0.87	6.6	7.59
GPT4-V [13, 14]	GPT4	42.74	12.50	46.48	54.17	44.19	19.13	59.88	6.94	38.31
GPT4-V(CoT) [13, 14]	GPT4	30.77	10.23	31.94	34.43	37.21	32.05	55.18	11.33	34.41
Task Navigator	Vicuna-13B	18.97	5.62	20.00	9.92	13.04	11.38	2.60	1.32	8.82
*Task Navigator**	Vicuna-13B	**46.55**	**30.34**	**49.33**	**48.85**	**39.13**	**37.89**	**21.38**	**23.03**	**34.01**

Table 1. **Comparisons of Task Navigator with existing MLLMs.** Current MLLMs, including Shikra [3], Qwen-VL [1], Lynx [33], InternLM-XComposer [34], mPLUG-Owl2 [31], and LLaVA-v1.5 [9] are tested on our VersaChallenge benchmark. We report the scores of each sub-task as well as the average scores across all tasks. These sub-tasks involve Commonsense Reasoning (CR), Physical Relation (PR), Physical Property (PP), Future Prediction (FP), Functional Reasoning (FR), Math Reasoning (MR), Embedded Text-based VQA (ETV), and Visual Planning (VP). * indicates the use of an external LLM for question decomposition.

4.2. Existing MLLMs with Task Navigator

Existing MLLMs are trained across various vision-and-language tasks, developing a lot of capabilities, from image captioning to VQA, object detection, and OCR. For instance, Shikra shows capability in referential dialogue, Qwen-VL, and LLaVA can perform strong OCR ability. However, we observe that these models face challenges when tackling complex tasks requiring a combination of multiple capabilities.

In Table 1, we present the quantitative results of current MLLMs on our benchmark. We report the accuracy of each sub-task as well as the average score across all tasks. This benchmark poses a significant challenge for existing MLLMs, with all MLLMs exhibiting relatively poor performance on the benchmark. We find that the most challenging tasks are math reasoning, embedded text-based VQA, and visual planning, all of which are specifically designed by us. These tasks demand the combination of various abilities, and the needed combinations are rarely present in the training datasets. This illustrates the limitation of MLLMs, they typically fail to spontaneously combine abilities they own to address intricate problems if such combinations are not present in their training data. We also observe that GPT4-V demonstrates superior performance on our benchmark, indicating that current open-source MLLMs still lag significantly behind GPT4-V. An interesting finding is that simply applying zero-shot CoT reasoning to GPT4-V can harm its performance. This could be due to CoT reasoning demanding a deep understanding of both text and visuals, highlighting that even GPT4-V struggles with multimodal CoT in challenging tasks.

Inner LLMs as Task Navigator. In Table 1, we implement the Task Navigator based on LLaVA-v1.5. Specifically, we adopt the LLaVA-v1.5 trained with LoRA [5], which has the Vicuna-13B as LLM. We decouple the Vicuna-13B component from the LLaVA-v1.5 to conduct question decomposition. Utilizing the Task Navigator, we observe an enhancement in performance across various sub-tasks including commonsense reasoning, physical properties reasoning, future prediction, and so on. The improvement in performance can be attributed to two main factors. The first is the process of question decomposition, where the LLM decomposes intricate issues into basic visual problems that MLLM can effectively tackle. By restructuring the problem into simpler sub-tasks, the model can focus on specific aspects of the problem, making it easier to analyze and solve. By continually seeking information through inquiries, the problem can eventually be solved. Secondly, when LLM is utilized to provide the final answer, it does not introduce visual tokens, which allows LLM to leverage its extensive knowledge to solve the problem.

However, we observe that for specific sub-tasks, such as math reasoning and visual planning, employing inner LLM for question decomposition does not yield improvement. Upon analysis, we identify that this is primarily due to the limited capability of LLM. When MLLM provides incorrect answers or exhibits hallucinations, it becomes challenging for inner LLM to detect and rectify these errors, leading to a decline in performance. To improve Inner LLM's question decomposition, we fine-tune the inner LLM and test the model. The results and analysis can be found in the supplementary materials.

External LLMs as Task Navigator. To further explore the potential of Task Navigator, we aim to leverage a powerful external LLM to investigate the extent of improvement that can be achieved in MLLM performance when question decomposition is effectively executed. In Table 1, we employ GPT-4 as an external LLM to perform question decomposition. The results demonstrate that with a robust LLM, our

Decomposition steps	1	2	3	4	5
Baseline	28.13	13.20	11.97	5.83	4.62
Task Navigator	59.33	32.03	28.21	11.65	12.31

Table 2. **Analysis of the number of question decomposition steps.** We demonstrate the accuracy for different numbers of decomposition steps.

Task Navigator can achieve substantial improvement across all complex tasks. We observe that for math reasoning and visual planning tasks, a strong LLM can yield significant improvement compared to the inner LLM. Moreover, compared with GPT4-V, our method attains comparable performance by integrating GPT4 with an open-source MLLM, demonstrating the effectiveness of our approach.

5. Ablation Study

In this section, we delve deeper into the *Task Navigator* by examining four key aspects, including the impact of question decomposition, the impact of question decomposition steps, the differences between dialogue decomposition and direct decomposition, and the generalization abilities of our method on MMBench.

5.1. Impact of Question Decomposition

We analyze the impact of question decomposition from two aspects: 1) whether the improvement on the benchmark primarily results from the incorporation of LLMs or question decomposition, and 2) the significance of the refinement process. In the second row of Table 3, we directly use LLaVA-v1.5 to extract the caption of the image, and input the caption and question into the LLM. In this case, it is equivalent to removing the intermediate step of question decomposition and directly using the LLM to answer the question. The results demonstrate that directly using LLM to answer questions without question decomposition does not yield good results, which proves that our question decomposition indeed brings benefits. In the third row, we directly ask LLM to decompose the question without the refinement process for analyzing the importance of refinement. Without the refinement process, the performance drops from 34.01 to 18.07.

5.2. Impact of Question Decomposition Steps

Our *Task Navigator* performs question decomposition in a dialogic manner, the number of decomposition steps is an important factor in our method. In Table 2, we illustrate the accuracy across different decomposition steps. As the number of steps increases, there is a noticeable decrease in accuracy for both the baseline and *Task Navigator*. This suggests that questions necessitating more decomposition steps are inherently more complex, resulting in reduced accuracy with an increasing number of steps. The baseline method is directly using LLaVA-v1.5 to answer the question, while the Task Navigator entails using LLaVA-v1.5 combined with GPT4 as an external LLM. Compared to the baseline, our method demonstrates significant improvements in accuracy through its decomposition process.

Decomp.	Refine.	Cap.	VersaChallenge								
			CR	PR	PP	FP	FR	MR	ETV	VP	Avg.
✗	✗	✗	12.79	14.77	9.33	7.63	8.69	14.02	0.87	6.6	7.59
✗	✗	✔	24.71	4.49	24.00	37.40	28.26	13.57	0.00	0.00	9.46
✔	✗	✗	40.22	19.79	40.00	49.61	36.95	20.36	7.80	12.50	18.07
✔	✔	✗	**46.55**	**30.34**	**49.33**	**48.85**	**39.13**	**37.89**	**21.38**	**23.03**	**34.01**

Table 3. **Impact of Question Decomposition.** *Decomp.* denotes the utilization of question decomposition. *Refine.* means the incorporation of refinement processes within question decomposition. *Cap.* refers to the exclusive reliance on captions for answering questions.

Decomposition		VersaChallenge								
Dialogue	Direct	CR	PR	PP	FP	FR	MR	ETV	VP	Avg.
✗	✗	12.79	14.77	9.33	7.63	8.69	14.02	0.87	6.6	7.59
✗	✔	27.15	6.57	16.92	26.92	21.62	18.04	0.00	0.00	8.72
✔	✗	**46.55**	**30.34**	**49.33**	**48.85**	**39.13**	**37.89**	**21.38**	**23.03**	**34.01**

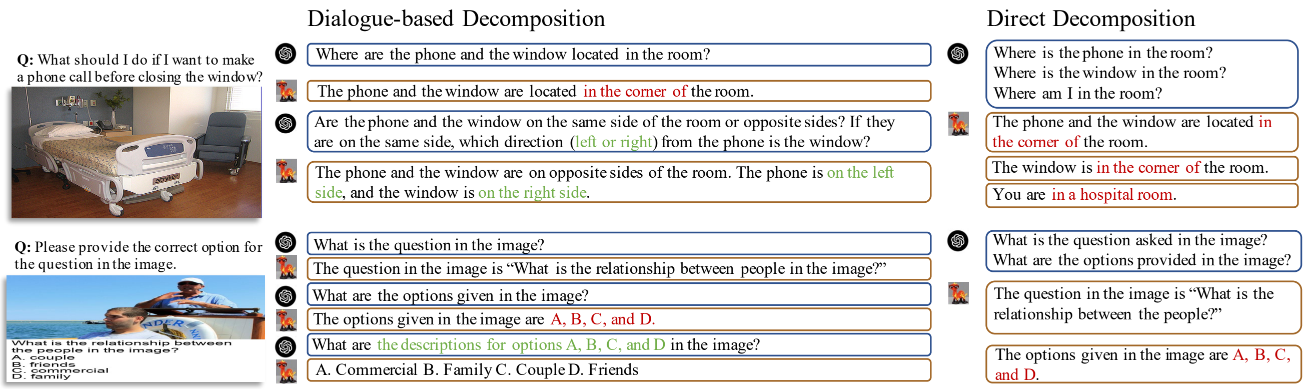
Table 4. **Comparisons of Dialogue Decomposition and Direct Decomposition.** We adopt LLaVA-v1.5 as MLLM and GPT-4 as external LLM for more reliable comparisons. The checkmark represents different decomposition methods.

5.3. Dialogue-based vs Direct Decomposition

Our *Task Navigator* has two implementations: dialogue-based question decomposition and direct question decomposition. In the dialogue-based approach, the decomposition process is structured as a dialogue, where the LLM poses one sub-question at a time. Based on the corresponding answer, it either inquires about the next sub-question or provides a final answer. Conversely, in the direct question decomposition approach, the LLM immediately decomposes the question into multiple sub-questions without awaiting responses. The final answer is then directly formulated based on these sub-questions and their respective answers.

In Table 4, we present a comparison of the two implementations, analyzing their distinct characteristics. Our baseline is established using LLaVA-v1.5, and we integrate GPT-4 as an external LLM. This study aims to investigate the differing aspects of question decomposition, making it essential to use an LLM that is proficient in both decomposition methods. For this reason, employing a powerful and reliable LLM like GPT-4 is necessary.

We observe that the performance of direct decomposition is inferior to that of the dialogue-based decomposition. This difference in effectiveness can be attributed to two potential factors: 1) The limited informative content in some questions. Without adequate visual information, these questions may not furnish enough prior knowledge for the LLM to ef-


Dialogue-based Decomposition
Direct Decomposition
Stryker
What is the relationship between the people in the image?
A. couple
B. friends
C. commercial
D. family

Q: What should I do if I want to make a phone call before closing the window?

Dialogue-based Decomposition:

- Where are the phone and the window located in the room?
- The phone and the window are located in the corner of the room.
- Are the phone and the window on the same side of the room or opposite sides? If they are on the same side, which direction (left or right) from the phone is the window?
- The phone and the window are on opposite sides of the room. The phone is on the left side, and the window is on the right side.

Direct Decomposition:

- Where is the phone in the room?
Where is the window in the room?
Where am I in the room?
- The phone and the window are located in the corner of the room.
- The window is in the corner of the room.
- You are in a hospital room.

Q: Please provide the correct option for the question in the image.

Dialogue-based Decomposition:

- What is the question in the image?
- The question in the image is "What is the relationship between people in the image?"
- What are the options given in the image?
- The options given in the image are A, B, C, and D.
- What are the descriptions for options A, B, C, and D in the image?
- A. Commercial B. Family C. Couple D. Friends

Direct Decomposition:

- What is the question asked in the image?
What are the options provided in the image?
- The question in the image is "What is the relationship between the people?"
- The options given in the image are A, B, C, and D.

Figure 5. **Differences between dialogue decomposition and direct decomposition.** In this figure, we showcase two instances from visual planning and embedded text-based VQA using different decomposition methods. Dialogue decomposition can handle unclear answers from MLLM and further inquire for more information, but direct decomposition fails to do so. Red font signifies the absence of valid information, while green font indicates the presence of valid information.

fectively inquire about all the necessary details. 2) The possibility that the answers provided by the MLLM may lack relevant information. In Figure 5, we show two instances. We find that when asking about location and expecting specific relative positions like 'left' or 'right,', MLLM might respond with a more generic 'in the corner of', failing to provide useful information. Similarly, when inquiring about options in an image, MLLM may only mention the presence of letters 'A' and 'B' without providing the content of the options. These factors make it challenging for the direct decomposition approach to yield correct answers. In contrast, dialogue-based decomposition allows for the posing of follow-up questions in response to MLLM's ambiguous answers, facilitating more precise outcomes.

5.4. Generalization of Task Navigator on MMBench

To further explore the generalization capabilities of our *Task Navigator*, we conduct experiments on the reasoning tasks on MMBench, which include attribute reasoning, logic reasoning, and relation reasoning. This focus was chosen because perceptual tasks typically do not necessitate the integration of multiple abilities. While the reasoning tasks in MMBench usually require various abilities to address.

As shown in Table 5, we employ LLaVA-v1.5 as our baseline and reproduce the results on MMBench. We observe that equipping *Task Navigator* enhances the performance of LLaVA-v1.5 on MMBench for all reasoning tasks. Especially in challenging logical reasoning task, our approach has brought significant improvements to the baseline, elevating it from 35.83% to 48.33%. The results show that our method can generalize effectively to other tasks. Consequently, our method can be seamlessly adapted and extended to various domains, leveraging the inherent versatility of LLMs.

Method	Task Navigator	MMBench			
		AR	LR	RR	Avg.
LLaVA-v1.5	✗	64.17	35.83	59.13	44.64
LLaVA-v1.5	✔	65.67	48.33	65.21	60.77

Table 5. **Comparisons on the reasoning tasks of MMBench.** We reproduce the results of LLaVA-v1.5 with the default prompt provided py MMBench.

6. Conclusion

In this paper, we focus on enhancing existing MLLMs to effectively address complex tasks that demand various capabilities. While current MLLMs have shown promising results in various downstream tasks, they often struggle with complex tasks that necessitate the integration of multiple abilities. Inspired by the way humans tackle intricate questions, we propose a novel framework, *Task Navigator*. In our framework, LLMs serve as navigators to iteratively decompose complex questions into sub-questions and refine them, which are solved by MLLMs to obtain the corresponding sub-answers, until LLMs have gathered enough information to provide the answer to the initial question. Our framework effectively extends MLLMs to address complex tasks, thereby broadening MLLMs' applicability. We also introduce *VersaChallenge*, a benchmark specifically designed for intricate tasks, which consists of eight complex tasks. Extensive experiments demonstrate that *Task Navigator* effectively enhances the capabilities of MLLMs in solving complex tasks requiring diverse abilities.

Acknowledgments: This work was in part supported by the National Natural Science Foundation of China under grants 62032006 and 62021001.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 1, 3, 5, 6

[2] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021. 3

[3] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023. 1, 2, 5, 6

[4] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 3, 5

[5] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021. 6

[6] Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. Decomposed prompting: A modular approach for solving complex tasks. In *ICLR*, 2023. 2

[7] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022. 2

[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1

[9] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 3, 5, 6

[10] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 1, 2, 5

[11] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhnag, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? *arXiv:2307.06281*, 2023. 3, 5

[12] Yadong Lu, Chunyuan Li, Haotian Liu, Jianwei Yang, Jianfeng Gao, and Yelong Shen. An empirical study of scaling instruct-tuned large multimodal models. *arXiv preprint arXiv:2309.09958*, 2023. 3

[13] OpenAI. Gpt-4v(ision) system card. 2023. 5, 6

[14] OpenAI. Gpt-4v(ision) technical work and authors. https://cdn.openai.com/contributions/gpt-4v.pdf, 2023. 5, 6

[15] OpenAI. Gpt-4 technical report, 2023. 1, 5

[16] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011. 3

[17] Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. In *ICCV*, pages 3170–3180, 2023. 5

[18] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Qixiang Ye, and Furu Wei. Grounding multimodal large language models to the world. In *ICLR*, 2023. 1, 2

[19] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *2009 IEEE conference on computer vision and pattern recognition*, pages 413–420. IEEE, 2009. 5

[20] 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. 2

[21] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. Learning to retrieve prompts for in-context learning. In *NAACL*, pages 2655–2671, 2022. 2

[22] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 3

[23] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 3

[24] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1, 3

[25] 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. 1, 3

[26] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. In *NeurIPS*, 2024. 1

[27] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. 2

[28] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023. 2

[29] Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023. 2

[30] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[31] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *arXiv preprint arXiv:2311.04257*, 2023. 5, 6

[32] Zhenfei Yin, Jiong Wang, Jianjian Cao, Zhelun Shi, Dingning Liu, Mukai Li, Lu Sheng, Lei Bai, Xiaoshui Huang, Zhiyong Wang, et al. Lamm: Language-assisted multimodal instruction-tuning dataset, framework, and benchmark. In *NeurIPS*, 2023. 3

[33] Yan Zeng, Hanbo Zhang, Jiani Zheng, Jiangnan Xia, Guoqiang Wei, Yang Wei, Yuchen Zhang, and Tao Kong. What matters in training a gpt4-style language model with multimodal inputs? *arXiv preprint arXiv:2307.02469*, 2023. 5, 6

[34] Pan Zhang, Xiaoyi Dong Bin Wang, Yuhang Cao, Chao Xu, Linke Ouyang, Zhiyuan Zhao, Shuangrui Ding, Songyang Zhang, Haodong Duan, Hang Yan, et al. Internlm-xcomposer: A vision-language large model for advanced text-image comprehension and composition. *arXiv preprint arXiv:2309.15112*, 2023. 3, 5, 6

[35] Yanzhe Zhang, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong Sun. Llavar: Enhanced visual instruction tuning for text-rich image understanding. *arXiv preprint arXiv:2306.17107*, 2023. 1, 2

[36] Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2023. 3

[37] Ge Zheng, Bin Yang, Jiajin Tang, Hong-Yu Zhou, and Sibei Yang. Ddcot: Duty-distinct chain-of-thought prompting for multimodal reasoning in language models. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[38] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023. 1, 5

[39] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*, 2024. 1, 2