DIEM: Decomposition-Integration Enhancing Multimodal Insights

Xinyi Jiang[1] Guoming Wang[1,*] Junhao Guo[1]
Juncheng Li[1] Wenqiao Zhang[1] Rongxing Lu[2] Siliang Tang[1]

[1]Zhejiang University [2]University of New Brunswick

[1]{jiangxinyi, NB21013, 22351300, junchengli, wenqiaozhang, siliang}@zju.edu.cn [2]RLU1@unb.ca

Abstract

*In image question answering, due to the abundant and sometimes redundant information, precisely matching and integrating the information from both text and images is a challenge. In this paper, we propose the **D**ecomposition-**I**ntegration **E**nhancing **M**ultimodal Insight (**DIEM**) which initially decomposes the given question and image into multiple subquestions and several sub-images aiming to isolate specific elements for more focused analysis. We then integrate these sub-elements by matching each subquestion with its relevant sub-images, while also retaining the original image, to construct a comprehensive answer to the original question without losing sight of the overall context. This strategy mirrors the human cognitive process of simplifying complex problems into smaller components for individual analysis, followed by an integration of these insights. We implement DIEM on the LLaVA-v1.5 model, and evaluate its performance on ScienceQA and MM-Vet. Experimental results indicate that our method boosts accuracy in most question classes of the ScienceQA (+2.03% in average), especially in the image modality (+3.40%). On MM-Vet, our method achieves an improvement in MM-Vet scores, increasing from 31.1 to 32.4. These findings highlight DIEM's effectiveness in harmonizing the complexities of multimodal data, demonstrating its ability to enhance accuracy and depth in image question answering through its decomposition-integration process.*

1. Introduction

With the rapid advancement of Large Language Models (LLMs) [2, 6, 38], they have showed astonishing capabilities across various tasks, simultaneously catalyzing the development of Multimodal Large Language Models (MLLMs) [1, 36, 40, 47]. In particular, when it comes to complex reasoning tasks, a significant breakthrough method of generating intermediate reasoning steps before inferring answers, known as Chain-of-Thought (CoT) [42], has gained considerable attention. This approach has sparked the creation of numerous models aimed at improving the CoT approach [24, 34, 41, 46]. Importantly, CoT [42] have been extended to MLLMs, such as multimodal-CoT [52], which utilizes a two-stage framework to focus on basic principles before deriving the final answer from these principles. However, a limitation of CoT [42], particularly in its application to MLLMs, is its focus only on text processing, but overlooking the integration and critical analysis of visual data.

Compared to text, images often convey richer and more diverse details, providing an abundance of visual cues and background knowledge. However, the high dimensionality, complex structure, and presence of visual noise make extracting information from images more challenging than text [3, 9, 35]. These complexities may cause models to overlook crucial image details or focus on irrelevant content. Furthermore, the objects, relationships, and attributes within images may have intricate connections with key textual information. Such intricacies in both image and text highlight a need for a more holistic approach, ensuring that both textual and visual elements are effectively utilized in complex reasoning tasks.

To address this challenge, we drew inspiration from the human method of processing text and image information. When humans answer a question based on an image, the process typically starts with a quick review of the question's text and a rapid "pre-scan" of the image to grasp the main content and context of both the question and the image. Following this, we begin to think through the question step by step and during this process, our attention is usually drawn to specific areas closely related to the question. For instance, if the question mentions a particular object, we might pay special attention to the color, shape, or other details of that object within the image. This localized observation allows us to delve deeply into the specific details and content within the image. Moreover, we also consider the relationships between various elements in the image and their connection to the question, which often furnish us with

more accurate information.

To emulate the human approach of sequential information processing, we introduce DIEM, a novel multimodal decomposition-integration strategy. Firstly, we decompose the original question into subquestions by GPT-3.5-Turbo [29] and decompose image into several sub-images using Segment Anything model [17]. Subsequently, by matching each subquestion with the respective sub-image using CLIP model [33], we identify multiple image regions closely related to each subquestion, allowing the model to capture key textual and visual information with more precise focus. To ensure that the overall image context is retaine, the original image is also provided for each subquestion. After this granular analysis, we then concatenate each pair of subquestion and sub-answer, integrating the visual information, to answer the original question. To illustrate our approach more intuitively, Fig. 1 displays the framework of DIEM.

This decomposition-integration strategy in multimodal tasks enables a more precise extraction and utilization of the deep links between images and text, leading to a thorough understanding and accurate reasoning of the original query. The design of DIEM allows for future adjustments or extensions based on different questions or image content, like modifying the number of sub-images matched to each subquestion as needed. Furthermore, DIEM is a training-free, plug-and-play method, which means it can be easily integrated with existing MLLMs. This flexibility allows DIEM to enhance these models without the need for extensive retraining or complex integration processes.

We evaluate DIEM on the ScienceQA [25] and MM-Vet [49]. Using the LLaVA-1.5 model [22], which had not been specifically trained or fine-tuned, as our baseline, our DIEM enhanced the average accuracy by 2.03% with the image accuracy by 3.40% on ScienceQA [25], and improved MM-Vet scores [49] from 31.1 to 32.4. Our results indicate that decomposing questions into subquestions and matching them with corresponding sub-images decomposed from the original image, followed by integrating all these fragments of information, can lead to a more coherent and precise understanding in multimodal task.

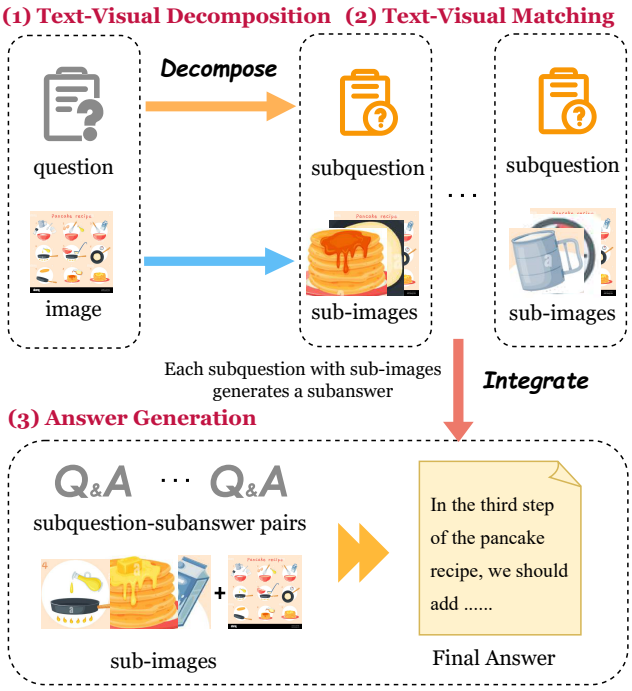

Figure 1. The framework of DIEM method. DIEM contains three stages: (1) Text-Visual Decomposition: Decompose the original question and image to derive a list of subquestions and a list of sub-images. (2) Text-Visual Matching: Determine several sub-images most relevant to each subquestion, making the visual information more targeted. (3) Answer Generation: After answering all subquestions, add subquestion-subanswer pairs to prompt the original question. Combined with the sub-images information corresponding to the original question, the correct option is finally chosen.

2. Related Work

2.1. CoT reasoning

As the data size grows and the parameters of the models become larger, Large Language Models(LLMs) exhibit remarkable emergent capabilities [8, 11, 31, 32]. There are many approaches that have shown to improve results with extended reasoning steps by the system, such as chain-of-thought [42], deductive verification [21], and self-verification [15, 26, 43]. Among them, chain-of-thought [42] is a prompting strategy that derives the final answer through a sequence of intermediate reasoning steps, mirroring human cognitive processes. It's been demonstrated to be highly effective for complex reasoning tasks. Inspired by the success of CoT, several studies [18, 27, 46, 51, 52] have ventured into extending the unimodal CoT to a multimodal version.

2.2. Multimodal Large Language Models

Multimodal Large Language Models (MLLMs) have made significant strides in recent years[12, 20, 48]. By integrating textual and visual modalities, MLLMs can transcend the limitations inherent in textual information [7, 14, 44, 50].

In current large-scale multimodal pre-training models, whether it's CLIP [33], UNITER [4], or ViLT [16], the embedding layers or the complexity of the visual features surpass that of the textual features. Multimodal models need to derive more knowledge from these visual characteristics. MiniGPT-4 [53] aligns a frozen visual encoder (BLIP-2 [19]) with a frozen LLMs (Vicuna [5]) using just a single projection layer. Initially trained with 5 million text-image pairs, it is subsequently fine-tuned with 3,500

high-quality datasets, rapidly equipping the language model with image understanding capabilities. A recent work, LLaVA [23], is a model capable of visual and textual multimodal transformations, consisting of a visual encoder and Vicuna v1.5 [5]. Through end-to-end training, LLaVA has achieved high performance in visual reasoning. Building on this, LLaVA-1.5 [22] has been further optimized and improved, surpassing several state-of-the-art (SOTA) models on various benchmarks. LLaVA [23] employs the pre-trained CLIP [33] VIT-L/14 as its encoder and is linked with the open-sourced LLaMA [39]. Unlike MiniGPT-4 [53], LLaVA [39] mainly applies instruction-tuning to the multimodal model. This represents the first attempt to extend instruction adjustment to the multimodal domain, using ChatGPT [28]/GPT-4 [30] to transform image-text pairs into appropriate instruction-following formats. By connecting the CLIP [33] visual encoder with the LLaMA [39] language decoder and undergoing end-to-end fine-tuning, it also achieves impressive performance.

There are also lots of related works in improving reasoning for multimodal tasks. Visual ChatGPT [23] primarily focuses on image generation and editing. KOSMOS-1 [13] and PaLM-E [10] demonstrate the zero-shot multimodal CoT capabilities with large-scale training. ViperGPT [37] instructs LLMs to generate Python code for a one-round query answering while MM-REACT [45] is a multi-round, dialogue-based system that may integrate the strong QA model as one of its vision experts.

3. Methods

3.1. Overview

DIEM not only allows for a deep dive into the details of the textual and visual information but also ensures a comprehensive grasp of the overall context. On the whole, DIEM first decomposes both the question and image, then integrates them, so that the model can more meticulously combine key textual and visual information. It performs three core steps:

1. *Text-Visual Decomposition:* Given an input of a question and an image, generate a list of subquestions and a list of sub-images.

2. *Text-Visual Matching:* Given a subquestion and the sub-images list, generate a matching list that contains several sub-images most relevant to the subquestion.

3. *Answer Generation:* Given the subquestion list and the matching list of each subquestion, generate corresponding subanswers. And then given subquestion-subanswer pairs and options, generate the final answer.

The first step shows the decomposition strategy of DIEM while the last two steps indicate the integration strategy. In Fig. 2, we take a detailed example to show how our method works. We will elaborate on each part of our method in the following subsections.

3.2. Text-Visual Decomposition

3.2.1 Question Decomposition

In the Question Decomposition Stage, we prompt the LLM with a question q and instruct it to decompose the original question into a series of subquestions to be answered. So we derive an initial list of subquestions $Subquestion_q = [q_1, q_2, \dots]$. Since there's a dependency order among the subquestions, to distinguish their sequence, we attach a pair of label $<sub\_q\_i></sub\_q\_i>$ to each subquestion q_i to clarify the order and identity of each subquestion. Thus, the form of q_i is like: $q_i = <sub\_q\_i>content</sub\_q\_i>$.

Furthermore, given the contextual linkage between subquestions, each subquestion might need to reference the answer(s) from any preceding subquestion(s) (termed as subanswers, with subquestion q_i corresponding to subanswer a_i). We employ a specific label Ref to denote this reference. If subquestion q_i requires the subanswer q_j, then the sentence for q_i will include the Ref label, and:

$$Ref = <sub\_a\_j></sub\_a\_j>. \tag{1}$$

Considering that the content of the final subquestion might not necessarily align with the intended meaning of the original question, we append the original question to the end of the subquestion list to ensure that the final answer directly addresses the original inquiry. Consequently, the subquestion list derived from the Question Decomposition is:

$$Subquestion_q = [q_1, q_2, \dots, q_m, q], \tag{2}$$

where m represents the number of subquestions obtained after decomposing question q.

3.2.2 Image Decomposition

In the Image Decomposition Stage, we divide the original image g into n sub-image. Each sub-image g_i captures an object or a distinct segment from the original image. The value of n is not fixed, but is adaptive based on different images. We construct a sub-image list G, specifically:

$$G = [g_1, g_2, \dots, g_i, \dots, g_n], \quad g_i \subset g. \tag{3}$$

Such a procedure helps us pay closer attention to the individual details within the image. Because in some cases, an entire image might be loaded with abundant information, making it challenging to capture all key details from a singular, holistic perspective. By breaking it down into sub-images, we can focus more specifically on certain segments of the image, thereby capturing important features or nuances that might otherwise be overlooked.

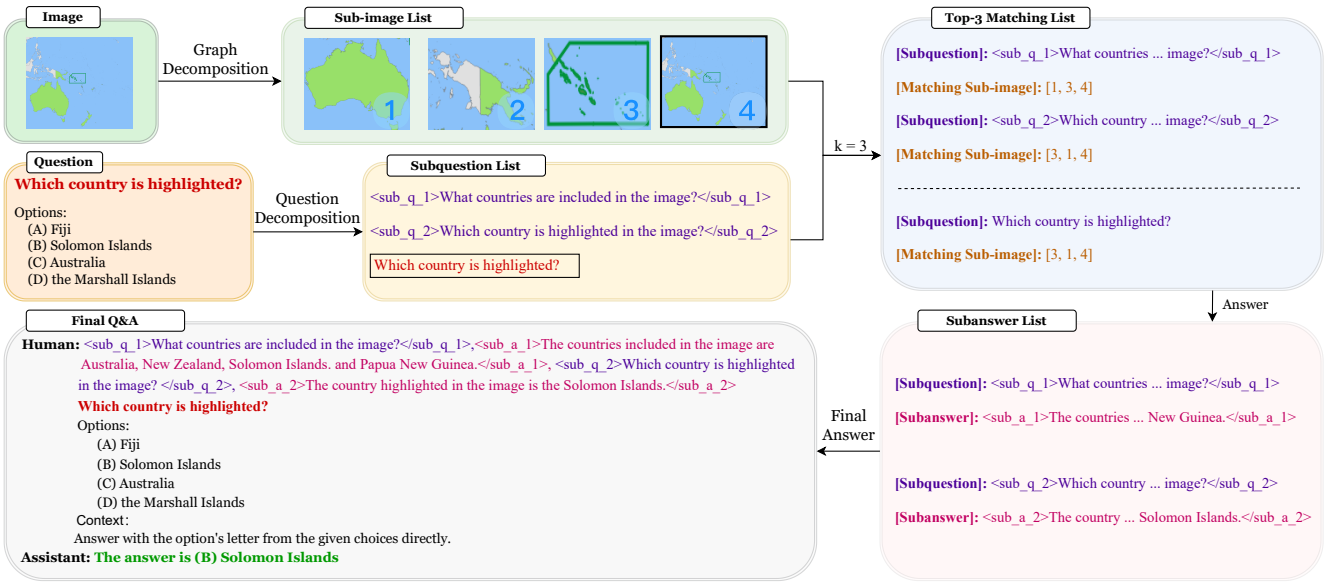

Figure 2. A specific example(omitting some formatting). First, the original question is decomposed into 2 subquestions and the original question is appended to the end of the subquestion list. Simultaneously, we segment the origional image into various sub-images, adding the origional image to the end of sub-image list as well. Then, we match each subquestion with 3 most relevat sub-images that must contain the origional image to ensure the integrity of the visual information. Based on the visual information of images in the matching list, we answer each subquestion, and derive 2 subanswers. In the end, we add 2 subquestion-subanswer pairs, options and context from the dataset to the prompt, combinig visual information from its matching list to derive the final answer.

3.3. Text-Visual Matching

To more accurately associate each subquestion with the relevant image content, we evaluate the similarity score S_{ij} between the features of subquestion q_i and sub-image g_j. The top k sub-images with the highest scores are considered to be most related to that subquestion. The set of images with the highest similarity scores for subquestion q_i constitutes the matching sub-image list $G_k(q_i)$:

$$\text{argtopk}(\{S_{i1}, S_{i2}, \ldots, S_{in}\}) \rightarrow \{j_1, j_2, \ldots, j_k\}, \quad (4)$$

$$G_k(q_i) = [g_{j_1}, g_{j_2}, \ldots, g_{j_k}], \quad (5)$$

The function $\text{argtopk}(\cdot)$ returns an indices set $[j_1, j_2, \ldots, j_k]$ corresponding to the indices of the top k elements in the similarity scores set. The sub-images indexed by these indices form $G_k(q_i)$.

Additionally, to preserve the overall context, we treat the original image g as a sub-image and include it in the sub-image list for each subquestion. This way, aside from conducting a localized visual analysis, we also maintain an understanding of the entire structure and content of the image. within the subquestion list $Subquestion_q$ for question $q$each subquestion q_i is associated with a list of sub-images $SubImg_{q_i}$ described as:

$$SubImg_{q_i} = G_k(q_i) \cup \{g\}. \quad (6)$$

It's worth noting that since the original question q is also included in the subquestion list $Subquestion_q q$ will also be associated with a corresponding list of sub-images $SubImg_q$.

3.4. Answer Generation

In the Answer Generation Stage, we combine the textual information of each subquestion q_i with the visual information from the corresponding matching sub-image list $SubImg_{q_i}$ and sequentially answer each subquestion without providing options. A reference tag Ref within a subquestion q_i indicates a dependency on another subanswer a_j. When such a reference is detected, we perform a substitution, replacing the tag Ref with the content of the relevant subanswer, updating it to ensure it does not reference any subanswer, as shown in Fig. 3, which follows this conditional logic:

$$q_i = \begin{cases} \text{Replace}(Ref, a_j), & \text{if } q_i \text{ contains } Ref \\ q_i, & \text{else} \end{cases}, \quad (7)$$

where $Replace(X, Y)$ denotes replacing X with Y.

A function $Answer(\cdot)$ representing the answer generation process, which requires the refined subquestion q_i and its matching sub-images list $SubImg_{q_i}$, is then applied to produce an answer a_i:

$$a_i = Answer(q_i, SubImg_{q_i}). \quad (8)$$

Combine the subquestion q_i and its corresponding subanswer a_i into a tuple, represented by $T_i = (q_i, a_i)$, where

$i = 1, 2, \ldots, m$.

When answering the last subquestion in the list, which is the original question q, we concatenate all the previous tuples before the question to guide the generation of the final answer. Additionally, visual features derived from the matching sub-image list corresponding to the original question q are incorporated. We then present options for the question, ultimately obtaining the answer as FinalAnswer, as determined by:

$$T^{'} = \bigoplus_{i=1}^{m} T_i, \tag{9}$$

$$FinalAnswer = Answer(T^{'} \oplus q, SubImg_q), \tag{10}$$

where $X \oplus Y$ represents the text concatenation operation with X preceding Y.

4. Experiments

4.1. Dataset

ScienceQA We evaluate our methods on the ScienceQA benchmark [25]. It is the first large-scale multimodal science question dataset that annotates the answers with detailed lectures and explanations. The dataset contains 21k multimodal multiple choice questions with rich domain diversity across 3 subjects, 26 topics, 127 categories, and 379 skills. The benchmark dataset is split into training, validation, and test splits with 12726, 4241, and 4241 examples, respectively. Each question is presented with a context in the form of natural language or an optional image.

In the test set of ScienceQA [25], there are 2178 multimodal questions and 2063 text-only questions. For the multimodal questions, we directly apply our method. For the text-only questions, we also carry out the Question Decomposition step, answering each subquestion in the list in sequence. Finally, the pairs of subquestions and subanswers are passed on to answer the original question. Fig. 3 shows an example of the text-only situation.

MM-Vet MM-Vet [49] is to evaluate Large Multimodal Models' ability on complicated multimodal tasks. It defines 16 emergent tasks of interest, integrated from the 6 defined core Vision-Language capabilities. The dataset contains 200 images, and 218 questions(all multimodal), all paired with their respective ground truths. Questions and expected responses in MM-Vet are designed to be open-ended to cover the diversereal-world scenarios. To better evaluate the responses, it leverage GPT-4 for evaluation. Each question is scored from incorrect (0 points) to correct (1 point), including defining different types of partially correct scores.

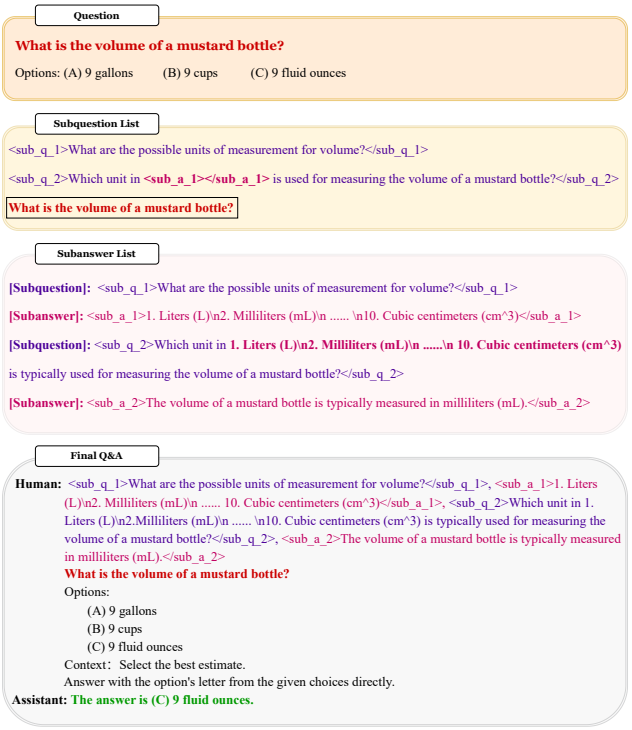

Figure 3. An example for the text-only questions. We only need to decompose the question, attaining a subquestion list. Then we sequentially answer the subquestions in the subquestion list. Since the second subquestion requires referencing the first subanswer, we replace the $<sub\_a\_1></sub\_a\_1>$ tag with the first subanswer. After replacing, the new subquestion is devoid of any references, and we then proceed to answer it.

4.2. Implementation

In our experiments, during the Question Decomposition Stage, we utilized GPT3.5-Turbo-4k [29] for subquestion decomposition and provided a few few-shot examples [34]. In the Image Decomposition Stage, we employed the Segment Anything model [17] for sub-image partitioning, where 811 questions resulted in only one sub-image, and 14 questions yielded up to 16 sub-images. In the Text-Visual Matching Stage, we used the CLIP [33] model to match each subquestion with its k most relevant sub-images. If the number of sub-images is less than k, we retain all of them in the sub-image lists. During the Answer Generation Stage, we used LLaVA-1.5-7b [22], a strong open multimodal model, as our fundamental model architecture. Our Standard data represents the performance of LLaVA-1.5-7b [22] on ScienceQA [25] and MM-Vet [49] without any training or fine-tuning. We applied our method directly on LLaVA-1.5-7b without any training or fine-tuning either. The rest of the settings are consistent with LLaVA-1.5 [22], including providing "Context" for questions and appending "Answer with the option's letter from the given choices di-

Method	Subject			Context Modality			Grade		Average
	SOC	NAT	LAN	TXT	IMG	NO	G1-6	G7-12	
Standard	69.18	65.76	**67.18**	71.83	65.70	**65.93**	70.52	60.25	66.85
Results on our own methods									
DIEM(SQ-only)	71.33	69.98	63.64	76.68	68.64	64.12	72.10	62.43	68.64
DIEM(SQ+SI, k=5)	**71.65**	**70.34**	63.64	**76.68**	**69.10**	64.12	**72.80**	**61.83**	**68.88**

Table 1. Main results on ScienceQA (accuracy %). NAT = natural science, SOC = social science, LAN = language science, TXT = text context, IMG = image context, NO = no context, G1-6 = grades 1-6, G7-12 = grades 7-12, SQ = subquestion, SI = sub-image. $k = 5$ means each subquestion is matched with the 5 most relevant sub-images on Text-Visual Matching stage. DIEM achieves improvements across most question classes, with the 3.4% increase in the IMG Modality that we particularly focused on.

Method	Rec	OCR	Know	Gen	Spat	Math	Total
Standard	36.1	**23.5**	17.4	22.2	25.9	**11.5**	31.1
Results on our own methods							
DIEM(SQ-only)	37.9	19.9	18.0	20.6	29.9	11.2	31.3
DIEM(SQ+SI, k=4)	**38.3**	22.5	**18.9**	**22.5**	**30.8**	11.2	**32.4**

Table 2. Main results on MM-Vet (points). Rec = recognition, OCR = optical character recognition, Know = knowledge, Gen = language generation, Spat = spatial awareness, SQ = subquestion, SI = sub-image. $k = 4$ means each subquestion is matched with the 4 most relevant sub-images. DIEM achieves improvements in most core visual-language capabilities, especially in Rec (+2.2) and Spat (+4.9).

rectly" to the prompt of the original question. Our tests were conducted on four NVIDIA GeForce RTX 3090 24G GPUs.

4.3. Results and Discussions

Main Results We compared the performance of DIEM with the Standard method, which neither decomposes the question nor the image, across different classes on both benchmarks. Main results on ScienceQA [25] is shown in Tab. 1. The average accuracy of our baseline, LLaVA-v1.5, is 66.85%. When we only decompose the question into subquestions(SQ-only), the average accuracy increases to 68.64%. When we combine both subquestions and sub-images (SQ+SI) to reasoning, the average accuracy reaches 68.88%, bringing improvements in almost all subjects, context modalities, and grades. Since DIEM pays attention on processing images, we should pay more attention to its performance in the IMG Modality. DIEM's accuracy in the IMG Modality is 69.10%(+3.4%) and compared to subquestion-only, incorporating sub-images to aid in subquestion reasoning further improves accuracy by 0.46%, bringing enhancements across all subjects.

Main results on MM-Vet [49] are shown in Tab. 2. It demonstrates that DIEM has achieved improvements across multiple core visual-language capabilities, particularly in the areas of Recognition and Spatial Awareness. This indicates that our decomposition-integration strategy enables the model to focus more on recognizing specific objects or features within an image, rather than being distracted by other irrelevant visual information. It also improves the understanding of the spatial relationships between objects in the image, helping the model to more accurately parse the positional relationships among these objects, thereby enhancing its ability to comprehend spatial layouts.

Overall, our DIEM method, through its more detailed and precise decomposition and matching mechanism, enhances the performance of the model in image ressoning problems, and is training-free, plug-and-play.

The Influence of k To investigate the impact of the number of sub-images k on answering questions, we took different k and compared the results on ScienceQA [25] and MM-Vet [49] relatively. The results are shown in Fig. 4. On ScienceQA [25], we divided it into two scenarios: decomposing and not decomposing the original questions. In the case of decomposing the questions(SQ), the performance of the model first rises with the increase of k. At $k = 5$, the model's average accuracy (Ave-Acc) and image accuracy (IMG-Acc) reach their peaks, which are 68.88% and 69.10%, respectively. However, at $k = 6$, although the average accuracy and image accuracy are still high, they begin to decline. When we only decompose the image without decomposing the question(No SQ), the performance of the model becomes relatively stable. Especially at $k = 4, 5, 6$, both the average accuracy and the image accuracy are very similar, and they are still better than the baseline. Similar experimental results are also reflected on MM-Vet [25]. This proves the beneficial effect of decomposing sub-images on performance improvement. However, when

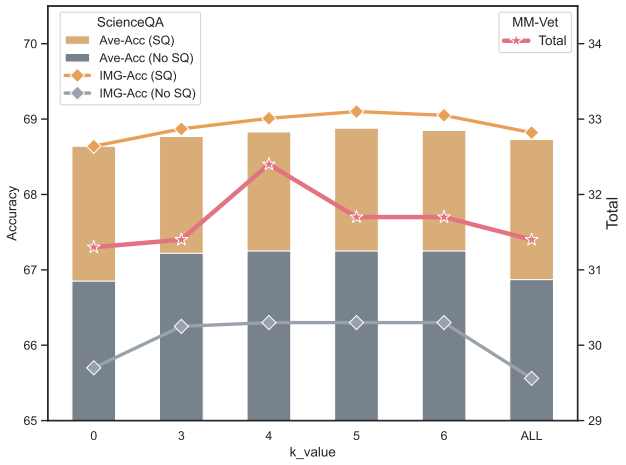

Figure 4. The influence of different numbers of subgraphs k most relevant to the subproblem on the results on ScienceQA and MM-Vet. Ave-Acc = average accuracy, IMG-Acc = image accuracy, Total = total scores. $k = 0$ indicates that the image is not segmented, while $k = ALL$ implies the use of all sub-images. The accuracy and scores initially increase with the number of sub-images, and the improvements then gradually slows down. After reaching a certain point, it tends to stabilize and then starts to decline.

using all sub-images, the accuracy will drop significantly. We think that this is because many images in the dataset contain only one object or even no obvious object, leading to ineffective decomposition of sub-images. Therefore, when the number of sub-images exceeds a certain threshold, it may introduce too much noise or redundant information, leading to a performance decline.

Interaction Between SQ and SI In multimodal tasks, the connection between text and images is crucial. As can be seen from Fig. 4, with the increase of k, the improvement of image Accuracy brought by decomposing both the question and the image (SQ) is greater than that of decomposing the image-only(No SQ). Moreover, the image accuracy of simultaneously decomposing both the question and the image begins to exceed the average accuracy when $k = 3$, while the image accuracy of only decomposing the question is below the average. This indicates a positive interaction between subquestions and sub-images. Decomposing the image alone can already help the model to focus and filter out unnecessary noise, bringing performance improvement. However, when we decompose the question into subquestions, each subquestion is more specific and can be more easily matched with the relevant sub-images. The model, through such precise matching, can better grasp the subtle relationships between text and images. By reducing information redundancy and processing information more focusedly, the accuracy is significantly enhanced.

4.4. Further Exploration

Necessity of Integration DIEM method adds the original question to the end of the subquestion list and includes the original image in the sub-image list for each subquestion. After sequentially answering all subquestions, all pairs of subquestion-subanswer are used to prompt the final answer to the original question. We focused on image accuracy on ScienceQA [25] and total scores on MM-Vet [49] when the subquestion list does not include the original question and the original image is not included in sub-image lists, under the condition of $k = 4$, while keeping other conditions unchanged. We did not conduct tests without original questions on MM-Vet [49], as it contains numerous simple questions that cannot be decomposed into sub-questions. The results are shown in Tab. 3. We can see that only by considering both the original question and the original image did we observe the maximum increase in performance, underscoring the importance of integrating the original question and original image in our method.

Dataset	OQ	OI	IMG-Acc	Total
ScienceQA	-	✓	66.61	-
	✓	-	68.50	-
	✓	✓	**69.01**	-
MM-Vet	✓	-	-	31.3
	✓	✓	-	**32.4**

Table 3. OQ = origional question, OI = original image. The performance declines when neither the original question nor the original image is considered after decomposition, highlighting the importance of the decomposition-integration strategy.

Subquestion Prompt Strategies On ScienceQA [25], when answering the original question on the end of the subquestion list, like the baseline method, we provide a "Context" and "Options" after the original question, where the "Context" is derived from the "hint" in the ScienceQA dataset. Hence, we sought to explore whether "Context" and "Options" should also be provided when prompting to subquestions. The results are shown in Tab. 4.

The results indicate that when only the "Context" is provided for the subquestions, the accuracy reaches its peak at 68.83%. However, when we provide options for the subquestions, the accuracy drops by 0.64%. This demonstrates that when solving problems, the "Context" consistently offers good guidance for the model at various stages. Since a subquestion is merely a part or detail of the original question, when we provide options for the subquestion to generate its answer, the resulting answer from the subquestion might be biased. When integrating all the subquestions and their answers to answer the final question, these biased answers might mislead, causing the answer to the original

question to deviate from the correct direction.

Method	CON	OPT	Ave-Acc	IMG-Acc
DIEM(k=4)	-	-	68.29	68.50
	✓	-	**68.83**	**69.01**
	✓	✓	68.19	67.22

Table 4. CON = context, OPT = option. We conducted three different tests: providing context and options to subquestions, providing only context without options, and not providing either. The results showed that giving context to subquestions has a positive impact, while providing options leads to a decrease in the model performance.

4.5. Error Analysis

The answers to the subquestions play a crucial role in responding to the origional question. Incorrect subanswers can lead to certain misleading conclusions for the original question. To better understand the behavior of DIEM and promote future work, we manually and randomly selected 50 multimodal incorrect answers generated by our method when $k = 5$ on ScienceQA [25] to see where the problems occurred.

We categorize them into two main groups, and further subdivide each category based on the different situations we observed. After each category, we have indicated the number of errors and provided our reasoning for classifying errors in that category. The results are as follows:

- **Partial Subanswers Incorrect (39 / 50):**
  - **Inappropriate Question Decomposition (5):**
    - Explanation: The decomposed subquestions are unrelated to the original question or deviate from the intended meaning of the original question.
  - **Inappropriate Image Decomposition (15):**
    - Explanation:
      1. Decomposing images from categories like flowcharts incompletely will adeversely impacts their strong comprehensive, results in the loss of textual information and further complicating the image-text matching step.
      2. The original image either lacks any distinct objects or contains only one object. Decomposing such images actually generates ineffective information.
  - **Inappropriate Image-Text Matching (10):**
    - Explanation: Due to different numbers of most relevant sub-images, some questions get matched with irrelevant sub-images, while others do not match all relevant sub-images.
  - **Subquestion Answering Failed (9):**
    - Explanation: Due to the limitations of the LLM, it couldn't generate comprehensible answers.
- **Subanswers Correct (11 / 50):**
  - **The answer's meaning is correct, but options weren't selected (2):**
    - Explanation: The final answer did not directly specify which option should be chosen. Instead, it generated a sentence whose meaning actually indicated the correct choice.
  - **Original Question Answered Incorrectly (9):**
    - Explanation: Even though we determined the subanswers to be correct, the final answer to the main question was still incorrect.

Furthermore, we observed numerous instances where the subanswers were incorrect, but the final answer was correct. This indicates that DIEM is robust to some extent.

5. Limitations and Future Work

For the decomposition of sub-questions, we solely relied on the GPT3.5-Turbo-4k [29] model. The shorter length limitation means that the few-shot prompt's performance is not ideal. In the future, we plan to employ larger language models for decomposing sub-questions and provide better few-shot prompt to ensure the quality of subquestions. We also suggest that decompose questions in conjunction with images might yield better results.

Moreover, upon decomposition, textual information might be lost, and strong inter-relationships between elements in some types of images might be severed, even though the original image is provided. If the original image is too simple, the image-text matching process might seem unnecessary. Moving forward, we aim to explore how to more accurately exclude distractions from intricate diagrams and extract vital information. We also believe that, for more complex images and questions, especially those containing multiple objects, our decomposition-integration strategy will yield better results.

6. Conclusion

We introduce DIEM, a multimodal reasoning method that proposes a decomposition-integration strategy. Our DIEM method first decomposes, then integrates, and focusing on individual parts before the whole, which inherently endows our method with training-free and plug-and-play characteristics. Experimental results have proven the effectiveness of our method, increasing image accuracy by 3.40%, average accuracy by 2.03% on the ScienceQA benchmark and the total score is improved by 1.3 points on the MM-Vet benchmark. This enhancement in performance not only demonstrates DIEM's ability to enhance multimodal comprehension, but also highlights its effectiveness in ensuring a more detailed and accurate synthesis of insights. Our error analysis demonstrates the potential of DIEM in handling complex problems and multi-object images that require in-depth analysis and understanding.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 1

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

[3] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International conference on machine learning*, pages 1691–1703. PMLR, 2020. 1

[4] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer, 2020. 2

[5] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2023. 2, 3

[6] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022. 1

[7] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 2

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[10] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023. 3

[11] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020. 2

[12] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023. 2

[13] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023. 3

[14] Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Daniel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, et al. Opt-iml: Scaling language model instruction meta learning through the lens of generalization. *arXiv preprint arXiv:2212.12017*, 2022. 2

[15] Weisen Jiang, Han Shi, Longhui Yu, Zhengying Liu, Yu Zhang, Zhenguo Li, and James T Kwok. Backward reasoning in large language models for verification. *arXiv preprint arXiv:2308.07758*, 2023. 2

[16] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021. 2

[17] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 2, 5

[18] 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

[19] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2

[20] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 2

[21] Zhan Ling, Yunhao Fang, Xuanlin Li, Zhiao Huang, Mingu Lee, Roland Memisevic, and Hao Su. Deductive verification of chain-of-thought reasoning. *arXiv preprint arXiv:2306.03872*, 2023. 2

[22] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 2, 3, 5

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 3

[24] Jieyi Long. Large language model guided tree-of-thought. *arXiv preprint arXiv:2305.08291*, 2023. 1

[25] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022. 2, 5, 6, 7, 8

[26] Ning Miao, Yee Whye Teh, and Tom Rainforth. Selfcheck: Using llms to zero-shot check their own step-by-step reasoning. *arXiv preprint arXiv:2308.00436*, 2023. 2

[27] Xuefei Ning, Zinan Lin, Zixuan Zhou, Huazhong Yang, and Yu Wang. Skeleton-of-thought: Large language models can do parallel decoding. *arXiv preprint arXiv:2307.15337*, 2023. 2

[28] OpenAI. Chatgpt. https://openai.com/blog/chatgpt, 2023. 3

[29] OpenAI. Gpt-3.5-turbo. https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates, 2023. 2, 5, 8

[30] OpenAI. Gpt-4 technical report, 2023. 3

[31] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022. 2

[32] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023. 2

[33] 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, 3, 5

[34] Ansh Radhakrishnan, Karina Nguyen, Anna Chen, Carol Chen, Carson Denison, Danny Hernandez, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, et al. Question decomposition improves the faithfulness of model-generated reasoning. *arXiv preprint arXiv:2307.11768*, 2023. 1, 5

[35] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 1

[36] Daniel Rose, Vaishnavi Himakunthala, Andy Ouyang, Ryan He, Alex Mei, Yujie Lu, Michael Saxon, Chinmay Sonar, Diba Mirza, and William Yang Wang. Visual chain of thought: Bridging logical gaps with multimodal infillings. *arXiv preprint arXiv:2305.02317*, 2023. 1

[37] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. *arXiv preprint arXiv:2303.08128*, 2023. 3

[38] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022. 1

[39] 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. 3

[40] 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. *arXiv preprint arXiv:2305.11175*, 2023. 1

[41] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022. 1

[42] 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. 1, 2

[43] Yixuan Weng, Minjun Zhu, Shizhu He, Kang Liu, and Jun Zhao. Large language models are reasoners with self-verification. *arXiv preprint arXiv:2212.09561*, 2022. 2

[44] Zhiyang Xu, Ying Shen, and Lifu Huang. Multiinstruct: Improving multi-modal zero-shot learning via instruction tuning. *arXiv preprint arXiv:2212.10773*, 2022. 2

[45] 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. 3

[46] Yao Yao, Zuchao Li, and Hai Zhao. Beyond chain-of-thought, effective graph-of-thought reasoning in large language models. *arXiv preprint arXiv:2305.16582*, 2023. 1, 2

[47] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023. 1

[48] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*, 2023. 2

[49] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023. 2, 5, 6, 7

[50] Xiaoman Zhang, Chaoyi Wu, Ziheng Zhao, Weixiong Lin, Ya Zhang, Yanfeng Wang, and Weidi Xie. Pmc-vqa: Visual instruction tuning for medical visual question answering. *arXiv preprint arXiv:2305.10415*, 2023. 2

[51] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*, 2022. 2

[52] 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. 1, 2

[53] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2, 3