

Supplementary Materials: A Picture Is Worth a Graph: A Blueprint Debate Paradigm for Multimodal Reasoning

Anonymous Authors

1 EXPERIMENTAL DETAILS

1.1 Algorithm for BDoG

For a better understanding of BDoG, an algorithmic procedure has been formulated in Algorithm 1.

Algorithm 1 BDoG

Require: Input $S = (\text{question } Q, \text{image } I \text{ and context } C)$, Multimodal LLM agents $A = (a_0, a_1, \dots, a_n)$, Max debate round R_{max} .
Initialize blueprint $G_0 \leftarrow \text{Extract_Entity_Relation_Attribute}(a, S)$, proponent a_p , opponent a_o , and moderator a_m with different personalities, $G \leftarrow G_0$.
while $R \leq R_{max}$ **do**
 \triangleright Affirmative Graph Generation
 $G_p \leftarrow \text{Graph_Condensation}(a_p, G, S)$
 $G \leftarrow G_p$
 \triangleright Negative Graph Generation
 $G_o \leftarrow \text{Graph_Condensation}(a_o, G, S)$
 $G \leftarrow G_o$
 \triangleright Debate Termination
 if $G_p = G_o$ or $R = R_{max}$ **then**
 $G \leftarrow [G_p, G_o]$
 Answer (a_m, G, S)
 break
 end if
end while

As depicted in Figure 1, we introduce the *Blueprint debate-on-graph (BDoG)* paradigm modeled by a hierarchical tree topology. The topology is defined by the vertex set containing nodes and edge set representing connections. Within the BDoG framework, leaf nodes directly exchange proposed updates to the shared knowledge graph, while interior nodes aggregate information flows.

Graph condensation aims to learn a small, synthetic, and informative graph \mathcal{G}' from a large, original graph \mathcal{G} . We consider the graph condensation as a reasoning process that multimodal LLMs learn to revise previous errors and filter out irrelevant information. We formally define the condensation process as follows:

Entity update: Let \mathcal{V}^1 and \mathcal{V}^2 represent the sets of entities in graphs \mathcal{G}^1 and \mathcal{G}^2 respectively after a debate round.

The set of common agreed-upon entities \mathcal{V}^c is defined as:

$$\mathcal{V}^c = \mathcal{V}^1 \cap \mathcal{V}^2 \quad (1)$$

Relation update: Let \mathcal{E}^1 and \mathcal{E}^2 represent the sets of relations in graphs \mathcal{G}^1 and \mathcal{G}^2 .

The set of common agreed-upon relations \mathcal{E}^c is defined as:

$$\mathcal{E}^c = \mathcal{E}^1 \cap \mathcal{E}^2 \quad (2)$$

Graph pruning: After identifying the common agreed-upon entities (\mathcal{V}^c) and relations (\mathcal{E}^c), the LLM is prompted to discard any

entities and relations in graphs \mathcal{G}^1 and \mathcal{G}^2 that are deemed irrelevant to the discussion.

The LLM evaluates each entity $v \in \mathcal{V}^1 - \mathcal{V}^c$ and relation $e \in \mathcal{E}^1 - \mathcal{E}^c$ not in the agreed set, and determines whether to keep it based on its relevance to the problem context and debate. Any deemed irrelevant are removed. The same process occurs for pruning \mathcal{G}^2 . This prompts the LLM to actively discard unnecessary information based on learned relevance, rather than a rigid mathematical operation, better simulating the flexibility of human judgment.

1.2 Statistics of Datasets

Table 1 provides an overview of the size and diversity of datasets used in the paper, including the number of instances, subjects, categories, and the average question length. These statistics can help in understanding the complexity and challenges posed by these datasets for multimodal reasoning-based QA systems.

| Dataset | Instance | Subject | Category | Avg. Ques. |
|-----------------|----------|---------|----------|------------|
| SQA-Test [3] | 2017 | 3 | 65 | 9.3 |
| SQA-Dev [3] | 2097 | 3 | 66 | 9.6 |
| MMBench-Dev [2] | 4329 | 6 | 20 | 8.9 |

Table 1: The statistics of ScienceQA test and dev set and MMBench dev set. Avg. Ques. = average counts of tokens in questions.

| Setting | Value |
|------------------------------|----------------|
| LLM | Vicuna-13B |
| Vision Encoder | EVA CLIP-G/14 |
| Hardware Requirement | 2x A100 (40GB) |
| Truncation Mode | Left |
| Number of Beams | 5 |
| Temperature | 1.0 |
| Top-p | 0.9 |
| Data Type | float32 |
| Image Resolution | 224x224 |
| Maximum Input Length | 256 |
| Maximum Output Graph Length | 128 |
| Maximum Output Answer Length | 50 |
| Maximum Debate Round | 4 |
| Inference Time for SQA | 4.7 s/sample |
| Inference Time for MMBench | 5.4 s/sample |

Table 2: Detailed model and experiment settings for Instruct-BLIP used in this paper.

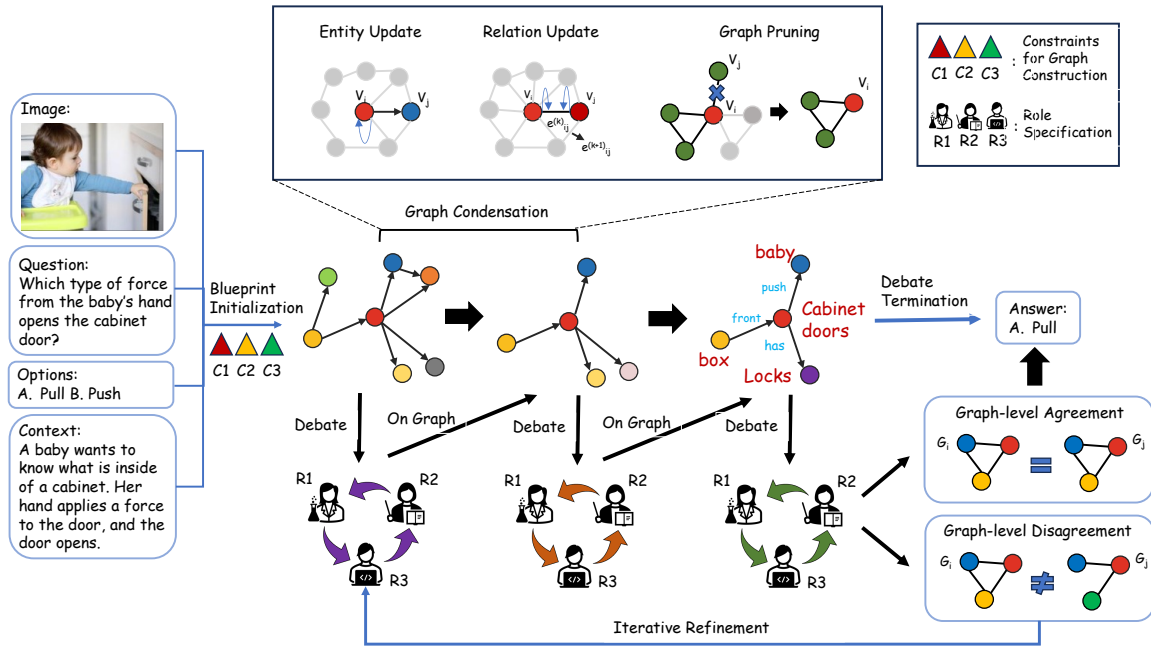


Figure 1: An overview of our Blueprint Debate-on-Graph (BDog) framework. It iteratively refines the blueprint with a multi-agent debate paradigm.

1.3 Model Deployment

The specifics of model deployment and hyperparameter configurations for the InstructBLIP system are detailed in Table 2. Experimental evaluations are conducted leveraging the computational resources of two NVIDIA A100 GPUs. Consistent with prior work [1], we employ the Vicuna-13B as the large language model and the EVA-CLIP as the vision encoding module. It is noteworthy that InstructBLIP imposes constraints on the overall input length; consequently, we set the output limits for graph generation and candidate answer production to 128 and 50 tokens, respectively. For outputs exceeding 256 tokens in length, we apply truncation techniques.

Furthermore, we report the inference time metrics for the ScienceQA (SQA) and MMBench datasets. Although the average question token count for MMBench is lower than SQA, the inference time required is higher. A plausible explanation for this discrepancy may be the inherently greater complexity of questions in the MMBench dataset, necessitating the generation of more intricate output graphs.

For the GeminiProVision and GPT-4V systems, we utilize their official APIs without employing any pre-processing or post-processing techniques.

1.4 Prompts

1.4.1 Role Specification. To simulate a debate process, we treat each multimodal LLM as an agent and specify distinct roles to them following previous studies. Although all the agents share similar actions defined by several instructions, formulate an analogy personality for each agent can encourage exchange of thoughts from diverse background and knowledge sources. Inspired by this, we explicitly model different roles as follows:

The diligent explainer:

Role = Scrutinize (Problem) + Explain (Details, Knowledge, Logic)
Goal = Benchmark (Solution)

You are Ben, a high school student with a track record of excellent grades, particularly in mathematics. Your friends admire your diligence and often seek your guidance in their studies. Your role is to scrutinize the problem at hand with your usual attention to detail, drawing from your vast knowledge of principles. Your clear and logical explanations are valuable, as they will serve as a benchmark for your friends to compare and refine their own solutions.

The creative problem-solver:

Role = Dissect (Problem) + Leverage (Creative Strategies)
Goal = Devise (Unique Solution)

You are Peter, a high school student recognized for your unique problem-solving abilities. Your peers often turn to you for assistance when they encounter challenging tasks, as they appreciate your knack for devising creative solutions. Today, your challenge is to dissect the given problem, leveraging your unique problem-solving strategies.

The attentive analyzer:

Role = Analyze (Problem) + Apply (Attentive Skills) + PieceTogether (Detailed Solution)
Goal = Catch (Missed Details)

You are Kitty, a high school student admired for your attentiveness and detail-oriented nature. Your friends often rely on you to catch details they might have missed in their work.

Your task is to carefully analyze the presented problem, apply your attentive skills, and piece together a detailed solution.

In these expressions, Scrutinize, Dissect, Analyze represent the core activities of examining the problem, Details, Knowledge, Logic, Creative Strategies, Attentive Skills represent individual traits, and the "+" signifies combining activities and goals.

1.4.2 Blueprint Extraction. Given a question Q and its correlated images I , BDoG formulates a blueprint \mathcal{G} encompassing coarse-grained descriptive information, thereby circumventing the necessity for manually annotated ground-truth scene graphs or image captions. To relate the visual and textual contexts, we initially investigate disaggregating the graph extraction process into three dissociated components: (1) Generation of captions C depicting I . (2) Detection of semantic associations Re between I and Q . (3) Aggregation of C and Re to compose \mathcal{G} representing the unified multimodal representation, where nodes signify detected visual objects and edges capture inter-object relations as informed by C along with image-question alignments from Re .

Although this approach shows potential, such a decomposed pipeline is susceptible to the error propagation problem. The quality of the generated graph will be influenced by introduced biases and erroneous captions pertaining to specific objects. To mitigate this issue, we aim to adopt a holistic perspective not only of the objects, which serve as primary units for visual reasoning, but also of their properties and interactions in a "big picture" manner visualized simultaneously. Specifically, we prompt the multimodal large language model to generate a graph \mathcal{G} which consists of objects \mathcal{G}_o , attributes \mathcal{G}_a and relationships \mathcal{G}_r that are relevant to answering the question Q .

Hint: {context} Question: {question} Options: {choices}
For the provided image and its associated question, generate a scene graph in JSON format that includes the following:

1. Objects that are relevant to answering the question.
2. Object attributes that are relevant to answering the question.
3. Object relationships that are relevant to answering the question.

1.4.3 Proponent and Opponent. The proponent and opponent are initialized by requiring them to adhere to the constraints for graph condensation, which involves updating, adding, and pruning nodes. The prompt for proponent is as the following:

You are a fellow debater from the AFFIRMATIVE side. For the provided image and its associated question, generate an updated graph from a different view based on the Debate Graph in JSON format that includes the following:

1. Objects that are more relevant to answering the question.
2. Object attributes that are more relevant to answering the question.
3. Object relationships that are more relevant to answering the question.
4. Delete the irrelevant objects, attributes and relationships

Hint: {context} Question: {question} Options: {choices} Debate Graph: {knowledge} Updated Graph:

Similarly, the prompt for opponent can be implemented as:

You are a fellow debater from the NEGATIVE side. For the provided image and its associated question, generate an updated graph from a different view based on the Debate Graph in JSON format that includes the following:

1. Objects that are more relevant to answering the question.
2. Object attributes that are more relevant to answering the question.
3. Object relationships that are more relevant to answering the question.
4. Delete the irrelevant objects, attributes and relationships

Hint: {context} Question: {question} Options: {choices} Debate Graph: {knowledge} Updated Graph:

It is important to note that the input debate graph for the proponent can be the updated graph from the opponent, and vice versa.

1.4.4 Moderator. We contend that prior research has employed an iterative approach involving text summarization to refine output, which has led to the trivialization of opinions characterized by a decline in performance. In contrast to this approach, our methodology directly extracts the answer through the analysis of the final graphs generated by both the proponent and opponent arguments. The prompt is presented as follows:

Hint: {context} Affirmative Graph: {knowledge[0]} Negative Graph : {knowledge[1]}
Use the image and two debate graph as context and answer the following question: Question: {question} Options: {choices}
Answer with the option's letter from the given choices directly.

Notably, visual elements are seamlessly integrated throughout reasoning processes, ensuring that the textual output is grounded in the visual context.

2 QUALITATIVE EXAMPLES

Figure 2 depicts the running examples within a debate round that compares BDoG with MAD on ScienceQA-IMG and MMBench datasets. Our proposed *Blueprint debate on graph (BDoG)* is a more effective way to present information than the vanilla *Multi-agent debate (MAD)*. It is more structured, more visual, and more interactive. This makes it easier to follow the flow of the debate, to identify the key points that are being made, and to explore the information in more detail.

The first case utilizes the ScienceQA dataset, evaluating geographic knowledge and map interpretation skills. The correct response (identifying the southernmost state) necessitates comparing the locations of various states. In the multi-agent debate scenario, despite the affirmative agent's accurate prediction of the relative position, the negative side overwhelmingly opposes it. Due to the widespread issue of hallucination in large language models (LLMs), detecting such misinformation demands significant effort. Instead of engaging in debate solely on the final answer, our model facilitates debate at the fact level. By strategically modifying blueprint nodes, BDoG demonstrates that the final answer should be derived by comparing the attributes (latitudes) of each candidate. BDoG offers the additional advantage of facilitating easier monitoring of changes, resulting in a more reliable system.

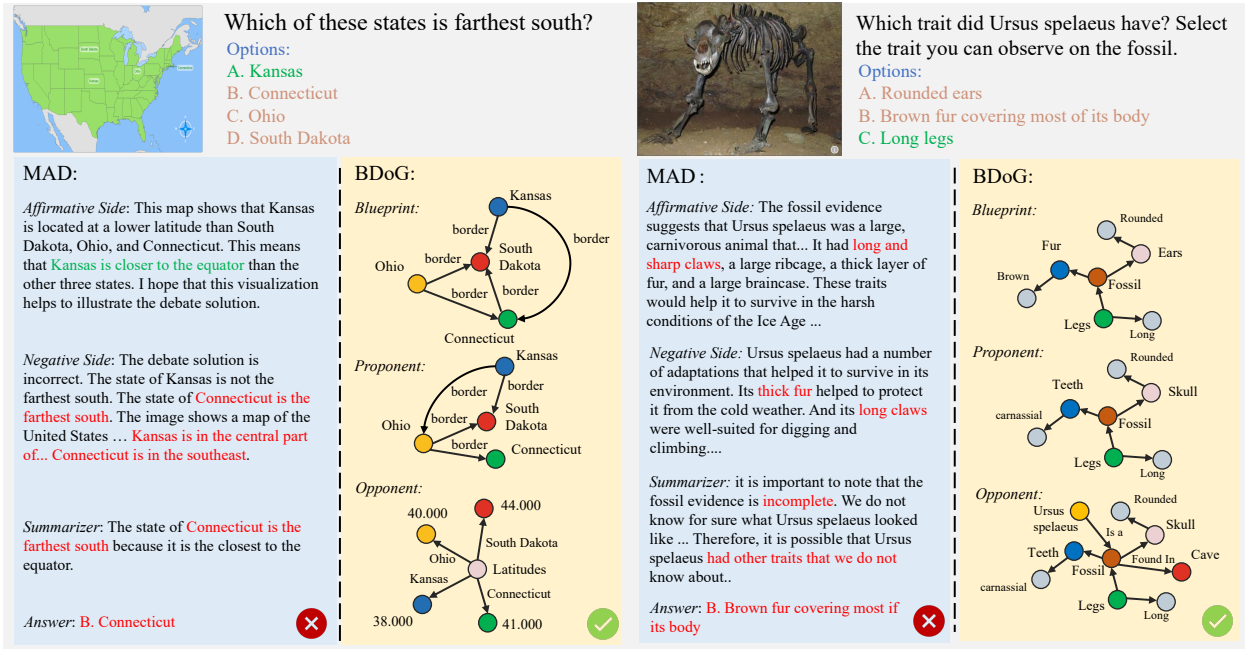


Figure 2: Intra-round case study comparing the proposed Blueprint Debate-on-Graph (BDoG) and vallina Multi-agent Debate (MAD) on ScienceQA-IMG (left) and MMBench (right) datasets. Green color indicates the correct answer/rationale and Red means incorrect/irrelevant predictions.

In the second case, drawn from the MMBench dataset, the task requires reasoning about attributes to answer the posed question. While the MAD method generates rationales incorporating extensive inherent knowledge and imagination, this leads to the erroneous inference of a live Ursus, distinct from the fossil depicted in the image. Notably, the question demands direct observation of the image, from which the correct answer – "Long legs" – can be readily inferred. The extraneous information generated by MAD misguides the model towards irrelevant concepts. Conversely, our BDoG method commences by analyzing the image, the associated question, and the candidate options. This effectively restricts the scope of analysis to the attributes of the fossil. Subsequently, BDoG incorporates additional observed features and refines potentially inaccurate nodes, ultimately leading to the accurate prediction.

3 OTHER ESSENTIALS OF THE MODEL

3.1 Monitoring the Debate Progress

Figures 3 and 4 depict the evolution of debate graphs for the ScienceQA-IMG dev set and MMBench-Dev set, respectively. These figures illustrate the dynamic changes in attributes, entities, and relations across debate rounds. Notably, the proposed BDoG framework outputs debate graphs in JSON format, facilitating further analysis. By comparing updated debate graphs with their original counterparts, we can track the modifications introduced during the debate process. This analysis underpins a key strength of BDoG: its ability to quantify the debate process through the lens of graph changes. This approach effectively demonstrates the value of dynamically

adjusting the initial graph based on the evolving discussion. Furthermore, the results presented in Figures 3 and 4 align with our hypothesis that disagreements and errors diminish as the debate progresses.

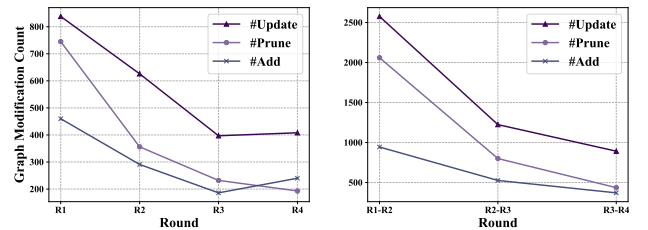


Figure 3: Statistics of intra-round (left) and inter-round (right) Blueprint condensation of BDoG with GeminiProVision for ScienceQA-IMG dev set. #Update: number of updated attributes; #Prune: number of pruned entities/relations; #Add: number of newly-added entities/relations.

3.2 Effect of Blueprint Quality

We conduct a human evaluation to assess the impact of blueprint quality, as illustrated in Figure 5. A random sample of 200 predictions from the MMBench dataset is selected for evaluation. Due to the absence of a standardized metric for evaluating generated graph quality, three annotators are tasked with classifying each blueprint as either high or low quality. The results reveal that 56%

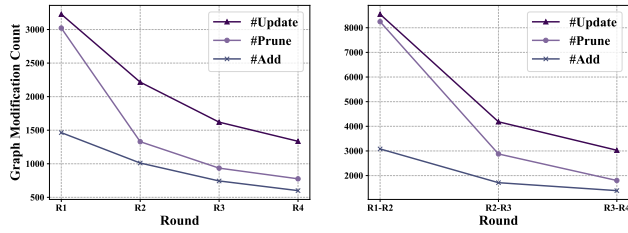


Figure 4: Statistics of intra-round (left) and inter-round (right) Blueprint condensation of BDoG with GeminiPro-Vision for MMBench-dev set. #Update: number of updated attributes; #Prune: number of pruned entities/relations; #Add: number of newly-added entities/relations.

of the initial blueprints were classified as low-quality. This finding aligns with expectations, given the complexity of the questions and the potential for the MLLM to generate coarse and imprecise direct answers.

To address the limitations of low-quality blueprints, we propose BDoG, a novel approach that iteratively refines the blueprint to enhance its conciseness and ultimately converge towards a correct answer. As demonstrated in the left panel of Figure 5, the final correctness rate is strongly correlated with blueprint quality. Notably, for questions with high-quality initial blueprints, the final correctness rate reaches 93.2%, highlighting the importance of a well-constrained initial graph. Furthermore, it is noteworthy that 67.8% of instances with low-quality blueprints ultimately result in correct predictions, demonstrating the effectiveness of BDoG's iterative refinement capabilities.

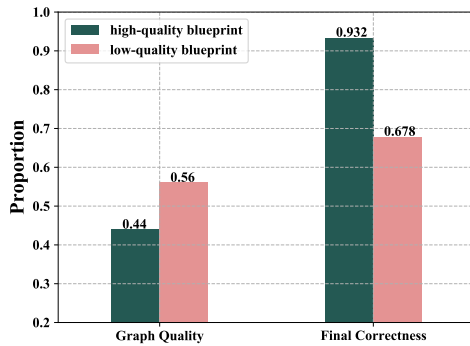


Figure 5: Human evaluation on the effect of blueprint quality for the GeminiProVision model.

3.3 Effect of Introduced Options

We further investigate the impact of incorporating options within a debate round. For a moderately parameterized multimodal LLM such as InstructBLIP-13B, introducing options can introduce noise and degrade final prediction performance. This is likely due to the model's inherent under-confidence in its judgments, leading to

options becoming a distraction that hinders effective reasoning. By ablating the options from the debate round, the performance of the InstructBLIP-13B model improves by 2.4% to 3.4%. However, this improvement is relatively minor for the larger GeminiProVision model. Such multimodal LLMs with over 175B parameters exhibit greater robustness to distractions from options, likely due to their increased capacity and sophistication.

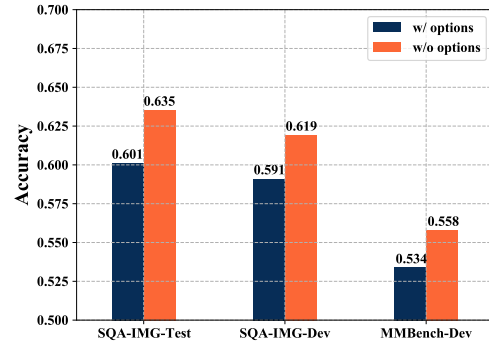


Figure 6: Ablation study on the effect of introducing candidate answers (options) within debate rounds for the InstructBLIP-13B model.

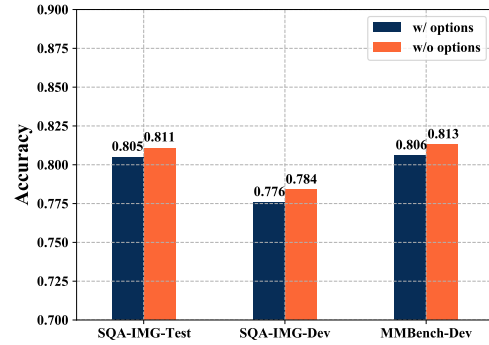


Figure 7: Ablation study on the effect of introducing candidate answers (options) within debate rounds for the GeminiProVision model.

REFERENCES

- [1] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Albert Li, Pascale Fung, and Steven C. H. Hoi. 2023. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. *ArXiv abs/2305.06500* (2023).
- [2] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2023. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281* (2023).
- [3] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems* 35 (2022), 2507–2521.