Lite-CoT: Lightweight Adapting Stage-wise Self-consistent Reasoning for ScienceQA

Anonymous ICME submission

***Abstract*—Existing Chain-of-Thought (CoT) methods struggle to efficiently integrate multi-modal information and maintain coherent reasoning, limiting performance on Science Question Answering (ScienceQA) tasks. To address this, we propose Lite-CoT, a lightweight adapting and self-consistent plan-and-solve framework guided by a semantic adapter and a stage-wise self-consistency strategy, enabling efficient and coherent reasoning across the entire CoT generation process. Unlike previous multi-path CoT methods that focus on diverse trajectories, Lite-CoT uses word-level consistency voting across candidate paths to construct reliable rationales. The semantic adapter bridges the semantic gap between frozen visual encoders and language models for precise multi-modal alignment. On the ScienceQA dataset, it achieves 93.86% accuracy with only 228M parameters, outperforming several much larger models while using just 1.75% of their parameters. On the A-OKVQA dataset, it attains 61.7% direct-answer accuracy, surpassing mainstream baselines with only 2.07% of the parameters of the reference models. Project page: https://Lite-CoT.github.io/.**

***Index Terms*—self-consistent reasoning, lightweight semantic adapter, CoT, ScienceQA**

I. Introduction

In recent years, Chain-of-Thought (CoT) prompting [1] has emerged as a key technique in Science Question Answering (ScienceQA), enabling models to perform step-by-step reasoning that improves both interpretability and prediction accuracy. However, despite its promise, CoT often suffers from inconsistent intermediate steps. Recent methods, such as Multimodal CoT [2] and LLaMA-Adapter [3], attempt to mitigate this issue, yet they still struggle to preserve the integrity of multi-modal information and to maintain coherent reasoning throughout the inference process. Consequently, these models tend to converge to suboptimal solutions, especially when dealing with ambiguous or uncertain queries.

A fundamental reason behind such suboptimal or incorrect outputs lies in the hallucination problem of Large Language Models (LLMs). Hallucination refers to the generation of fluent but factually inaccurate or fabricated content, often caused by an overreliance on prior knowledge rather than input-grounded evidence. Moreover, oversimplified or poorly aligned reasoning processes may fail to incorporate essential contextual details, further exacerbating the inconsistency and unreliability observed in multi-modal ScienceQA tasks. For example, as shown in Fig. 1, although the context clearly states that *"Coco pulls toward herself, while Rusty pulls away from Coco"*, Multimodal-CoT incorrectly selects *"(A) away from Coco"* in response to the question *"What is the direction of the opposing force when Rusty pulls the toy?"*. This reflects a semantic misalignment between the visual and textual inputs and reveals a misunderstanding of basic physical principles.

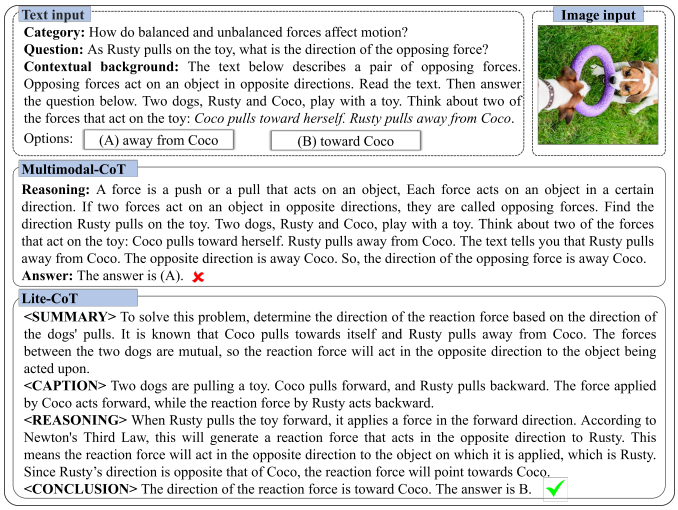

Fig. 1. An example compares Multimodal-CoT and the proposed Lite-CoT in generating rationales and answers. Despite the contextual cue that *"Coco pulls toward herself, while Rusty pulls away from Coco"*, Multimodal-CoT hallucinates and predict "(A) away from Coco" when asked about the direction of the opposing force on Rusty. In contrast, Lite-CoT avoids this error.

On the other hand, reasoning-oriented multimodal LLMs are rapidly scaling from billions to trillions of parameters, yet lightweight deployment remains essential in real-world applications. To this end, and following prior works such as Multimodal-CoT [2] and LLaMA-Adapter [3], this work focuses on developing a ScienceQA model with fewer than one billion parameters, aiming to maximize the potential of CoT reasoning under limited computational resources. Specifically, inspired by the LLaVA-CoT [4] that introduces a multi-stage reasoning procedure, Lite-CoT is designed to efficiently develop a stage-wise self-consistency reasoning mechanism to dynamically select the most reliable intermediate outcomes for generating optimal rationale. The other key component of Lite-CoT is its lightweight semantic adapter, which effectively bridges frozen visual encoders and language models. The adapter enables flexible and fine-grained multimodal feature fusion, thereby achieving stronger cross-modal semantic alignment. Specifically, this paper makes three key **contributions**:

(1) This paper introduces Lite-CoT, an efficient and lightweight framework that integrates a semantic adapter with stage-wise self-consistent plan-and-solve reasoning to achieve optimal rationale generation.

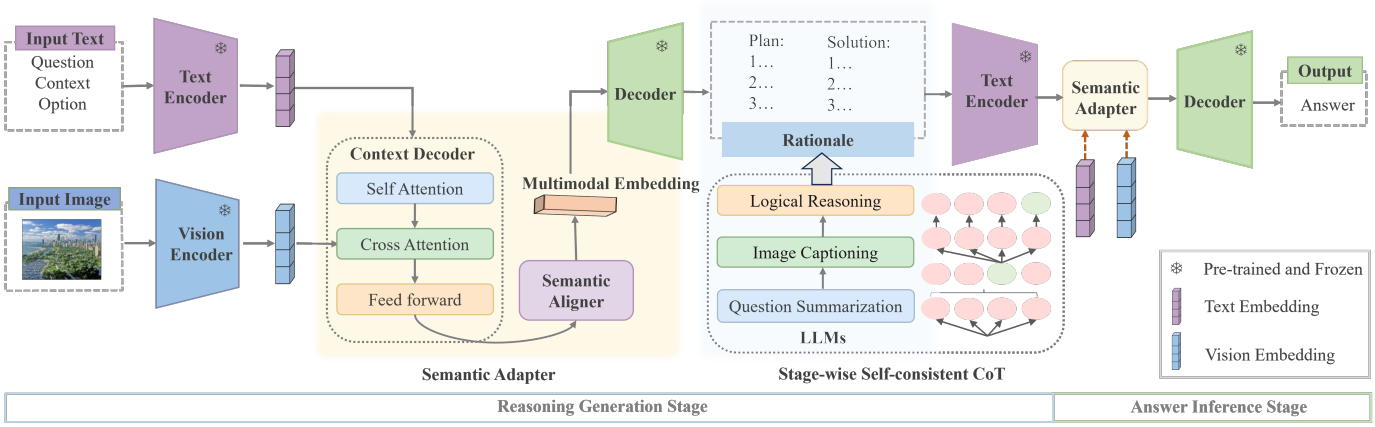

Fig. 2. Overview of the Lite-CoT framework. The framework aligns text and vision via a lightweight semantic adapter and employs a stage-wise self-consistent CoT mechanism to select optimal rationales. A two-stage fine-tuning scheme first generates structured reasoning and then infers the final answer.

(2) The semantic adapter provides fine-grained alignment of visual and textual features to enhance multi-modal representation learning. During reasoning, it guides stage-wise self-consistency by selecting trustworthy intermediate reasoning paths, during answer inference, it helps integrate intermediate results to produce accurate outputs.

(3) Lite-CoT delivers strong results with a compact model. It achieves 93.86% accuracy on ScienceQA using only 228M parameters (1.75% of the baseline size) and attains competitive accuracy on A-OKVQA with merely 2.07% of the baseline parameters, outperforming substantially larger models.

II. Related Work

Multi-modal Reasoning. In multi-modal reasoning, various frameworks [1] have been developed to enhance cross-modal integration. Early methods such as ViLBERT [5] use dual-stream architectures with cross-modal attention but suffer from frozen encoders that limit deep fusion. Multimodal-CoT [2] incorporates explicit reasoning chains via a two-stage training process, though at high annotation cost. More recently, Honeybee's C-Abstractor [3] leverages deformable convolutions and geometric priors to strengthen symbolic reasoning.

Structured Representation and Inference. To support complex reasoning, structured representations [6] are increasingly used to organize intermediate steps and improve interpretability. Scene-graph–based methods [7] enhance spatial reasoning but have limited scalability. Instruction-tuned models like GPT-4o [8] generate structured rationales for 89.7% of multi-step questions. Enhanced CoT approaches such as Tree of Thoughts (ToT) and Graph of Thoughts (GoT) have been proposed, but they remain computationally expensive due to extensive branching and exploration. In particular, LLaVA-CoT [4] employs stage-wise beam search for scalable and interpretable reasoning in vision–language tasks.

Reasoning Optimization for ScienceQA. Recent efforts have focused on improving reasoning efficiency in ScienceQA by leveraging prompt engineering and lightweight model architectures, aiming to reduce computational overhead while maintaining strong performance [9]. Prompt-based methods [1], [9] introduces step-by-step reasoning but often suffers from inconsistency in intermediate steps. Multi-path decoding approaches such as SC-CoT [10] and MC-CoT [11] reduce error propagation via majority voting over candidate trajectories and extend voting to both rationales and answers for small models. Modular architectures like BLIP-2 [12] show that frozen vision-language backbones with lightweight adapters achieve competitive performance with fewer trainable parameters.

Limitations of Existing Approaches. Current work relies on reasoning chains without intermediate verification, making them prone to cumulative errors. Cross-modal alignment remains challenging, and pseudo-alignment can occur when visual cues are inconsistently used. Structured reasoning methods such as ToT and GoT also incur higher computational costs. These limitations motivate Lite-CoT, which integrates stage-wise verification with a lightweight semantic adapter to achieve fine-grained cross-modal alignment efficiently.

III. Methodology

As shown in Fig. 2, Lite-CoT adopts a stage-wise self-consistent plan-and-solve framework with a lightweight semantic adapter. Text and image embeddings from pre-trained encoders are fused via a context decoder and an alignment module to maintain cross-modal consistency. The LLMs then employ a stage-wise self-consistency mechanism to select the most reliable intermediate reasoning steps and generate an optimized structured reasoning process. Finally, the answer inference stage produces the final conclusion and answer.

A. Pre-training Semantic Adapter

The semantic adapter is a trainable module bridging frozen image encoders and LLMs. It includes a context decoder, with self-attention and cross-attention, to capture textual context and guide attention to relevant visual cues, and a semantic aligner that refines visual–textual representations. During pre-training, the adapter is paired with frozen encoders and trained on image–text data, where the decoder integrates both modalities through a Transformer structure.

Given an input image I_v, a pre-trained visual encoder is employed to extract visual features $f_v = \mathcal{E}_V(I_v)$. For the input question Q, context C, and candidate options P, the textual input is defined as $I_t = \{Q, C, P\}$. A text encoder is used to obtain textual features $f_t = \mathcal{E}_T(I_t)$. The context decoder then projects and fuses textual embeddings with visual features via self-attention and cross-attention, generating fused representations formulated as:

$$f_{\text{CD}} = \mathcal{D}_{\text{Context}}\big((f_t, f_v), \theta\big), \tag{1}$$

where θ denotes the trainable parameters of the context decoder $\mathcal{D}_{\text{Context}}$.

Subsequently, a semantic aligner is applied to refine and align the fused representation, enhancing cross-modal consistency. The final multi-modal features are given by:

$$f_{\text{SA}} = \mathcal{A}_{\text{Semantic}}(f_t, f_{\text{CD}}) + f_{\text{CD}}. \tag{2}$$

The refined features f_{SA} are then passed to the decoder of a language model to generate the output sequence $\mathbf{O}$.

The semantic adapter is pre-trained using an autoregressive objective. The probability of generating a target sequence $\mathbf{O} = (O_1, \ldots, O_L)$ of length L is defined as:

$$\mathcal{P}_\phi(\mathbf{O} \mid I_v, I_t) = \prod_{i=1}^{L} \mathcal{P}_\phi(O_i \mid I_v, I_t, O_{<i}), \tag{3}$$

where ϕ denotes the trainable parameters of the semantic adapter, and $O_{<i}$ represents the previously generated tokens.

B. Stage-wise Self-consistent Plan-and-Solve

Unlike representative CoT strategies where traditional approaches either generate a single and non-validated path or apply validation only at the final answer stage across multiple paths, our proposed approach performs stage-wise validation during reasoning generation. It progressively selects the most consistent words or steps to reduce error propagation.

1) Stage-wise Self-Consistent Mechanism: At each stage, the model generates multiple candidate outputs, and a voting mechanism selects the most consistent intermediate result, which is then used as input for the next stage. During the reasoning generation process, multiple reasoning trajectories R^i are generated for the given text-image pair (I_v, I_t) through N_r rounds of sampling from the model, as formally defined:

$$R^i \sim \mathcal{P}(R \mid I_v, I_t), \quad i = 1, 2, \ldots, N_r. \tag{4}$$

The most consistent reasoning is obtained by progressively selecting the most consistent words across the reasoning trajectories. The best reasoning R^* is chosen as:

$$R^* = \text{Vote}(\{R_i^{(j)}\}), \tag{5}$$

where the word with the highest frequency at position j in the N_r generated rationales is selected to form the optimal reasoning token R_j^*.

In the conclusion phase, multiple answers are generated based on the best reasoning process R^* and the input text-image pair (I_v, I_t). Different answers are generated by sampling N_a times from the model A_i, as defined:

$$A_i \sim P(A \mid I_v, I_t, R^*), \quad i = 1, 2, \ldots, N_a. \tag{6}$$

Finally, the answer is selected by majority voting, as defined:

$$A^* = \text{Vote}(\{A_i\}). \tag{7}$$

Among the N_a generated candidate answers, the one with the highest occurrence frequency is chosen as the most reliable and consistent answer.

2) Voting Strategy: Unlike the Multimodal-CoT [2] model, which directly employs the cross-entropy function to compute the model loss, a combined loss voting strategy is adopted to further enhance the inference process. Specifically, the proposed method first calculates both the average prediction vector $\overline{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$ and the weighted average vector $\hat{Y} = \sum_{i=1}^{n} w_i Y_i$, where the weight $w_i = \frac{1}{1+\sigma_i}$, and σ_i denotes the standard deviation of the i-th prediction. The average vector $\overline{Y}$ captures the overall prediction trend and helps suppress the impact of random fluctuations. To estimate the statistical characteristics of the predictions, the model is sampled N_s times and the variance σ of the j-th dimension of the outputs is computed as follows:

$$\sigma_i^2 = \frac{1}{N_a}\sum_{i=1}^{N_a}(Y_i - \mu)^2, \quad \mu = \frac{1}{N_s}\sum_{i=1}^{N_s} Y_i, \tag{8}$$

where μ denotes the mean of the j-th dimension of the outputs. A smaller σ_i indicates more stable predictions and is thus assigned a higher weight, while a larger σ_i reflects greater uncertainty and is given a lower weight. The weight vector $w_i = \frac{1}{1+\sigma_i}$ dynamically adjusts each prediction's contribution based on its confidence. As σ_i decreases, w_i increases, emphasizing more reliable outputs. This confidence-aware weighting enhances model robustness by prioritizing consistent predictions. The final output is computed as a linear combination of the average prediction $\overline{Y}$ and the weighted average prediction $\hat{Y}$, given by:

$$Y^* = \alpha\hat{Y} + (1 - \alpha)\overline{Y}, \tag{9}$$

where α is a balance parameter.

Finally, the cross-entropy loss is computed between the combined prediction Y^* and the ground-truth label A:

$$\mathcal{L} = \text{CrossEntropy}(Y^*, A) \tag{10}$$

C. Two-stage Fine-Tuning Process

The fine-tuning process has two stages, reasoning generation and answer inference. As shown in Fig. 2, both stages share the same architecture but differ in input–output formats. The visual encoder remains frozen, while the language model is fine-tuned. In the reasoning stage, the model produces explicit reasoning chains from multi-modal inputs. In the inference

TABLE I
ACCURACY COMPARISON ACROSS EIGHT QUESTION CATEGORIES ON THE SCIENCEQA DATASET. BOLD AND UNDERLINED VALUES INDICATE **BEST** AND SECOND-BEST RESULTS, RESPECTIVELY.

Method	Size	NAT	SOC	LAN	TXT	IMG	NO	G1-6	G7-12	Avg.
Human [1]	-	90.23	84.97	87.48	89.60	87.50	88.10	91.59	82.42	88.40
MCAN [13]	95M	56.08	46.23	58.09	59.43	51.17	55.40	51.65	59.72	54.54
Top-Down [14]	70M	59.50	54.33	61.82	62.90	54.88	59.79	57.27	62.16	59.02
BAN [15]	112M	60.88	46.57	66.64	62.61	52.60	65.51	56.83	63.94	59.37
DFAF [16]	74M	64.03	48.82	63.55	65.88	54.49	64.11	57.12	67.17	60.72
ViLT [17]	113M	60.48	63.89	60.27	63.20	61.38	57.00	60.72	61.90	61.14
Patch-TRM [18]	90M	65.19	46.79	65.55	66.96	55.28	64.95	58.04	67.50	61.42
VisualBERT [19]	111M	59.33	69.18	61.18	62.71	62.17	58.54	62.96	59.92	61.87
UnifiedQA [20]	223M	68.16	69.18	74.91	63.78	61.38	77.84	72.98	65.00	70.12
GPT-3 [21]	175B	74.64	69.74	76.00	74.44	67.28	77.42	76.80	68.89	73.97
GPT-3 (CoT) [1]	175B	75.44	70.87	78.09	74.68	67.43	79.93	78.23	69.68	75.17
GPT-4 (CoT) [22]	>1T	84.06	73.45	87.36	81.87	70.75	90.73	84.69	79.10	82.69
Chameleon [22]	>1T	89.83	74.13	89.82	88.27	77.64	92.13	88.03	83.72	86.54
Multimodal-CoT$_{base}$ [2]	223M	87.52	77.17	85.82	87.88	82.90	86.83	84.65	85.37	84.91
Multimodal-CoT$_{large}$ [2]	738M	<u>95.91</u>	82.00	90.82	<u>95.26</u>	88.80	92.89	92.44	90.31	91.68
LLaMA-Adapter [3]	7B	84.37	88.30	84.36	83.72	80.32	86.90	85.83	84.05	85.19
LLaVA [23]	13B	90.36	<u>95.95</u>	88.00	89.49	88.00	90.66	90.93	90.90	90.92
LLaVA+GPT-4 (Judge) [23]	13B	91.56	**96.74**	91.09	90.62	88.99	93.52	92.73	92.16	92.53
LaVIN-13B [2]	13B	90.32	94.38	87.73	89.44	87.65	90.31	91.19	89.26	90.50
LLaMA-SciTune [3]	13B	89.30	95.61	87.00	93.08	86.67	91.75	84.37	91.30	90.03
PILL [24]	7B	90.36	95.84	89.27	89.39	88.65	91.71	92.11	89.65	91.23
Honeybee [25]	13B	95.20	96.29	91.18	94.48	93.75	<u>93.17</u>	<u>95.04</u>	93.21	<u>94.39</u>
Lite-CoT$_{Base}$ (**Ours**)	228M	94.84	94.21	<u>91.59</u>	94.91	<u>94.92</u>	92.10	93.73	<u>94.10</u>	93.86
Lite-CoT$_{Large}$ (**Ours**)	746M	**97.77**	93.55	**92.77**	**97.60**	**95.46**	**94.00**	**95.39**	**95.94**	**95.59**

stage, the generated reasoning is concatenated with the text input to predict the final answer.

In the reasoning generation stage, the model $\mathcal{M}_{\text{rational}}$ is trained to generate a reasoning process R conditioned on the multi-modal input I, defined as:

$$R = \mathcal{M}_{\text{rational}}(I). \tag{11}$$

Here, $I = \{I_v, I_t\}$ represents the multi-modal input. The reasoning process R is generated in an autoregressive manner, with its probability defined as:

$$\mathcal{P}(R \mid I_v, I_t) = \prod_{i=1}^{L} \mathcal{P}_\psi(R_i \mid I_v, I_t, R_{<i}), \tag{12}$$

where ψ denotes the trainable parameters of $\mathcal{M}_{\text{rational}}$, L is the length of the target reasoning sequence, and $R_{<i}$ refers to the previously generated reasoning tokens.

In the answer inference stage, a separate model $\mathcal{M}_{\text{answer}}$ is trained to predict the final answer A based on the original multi-modal input and the generated reasoning process. This can be formally expressed as:

$$A = \mathcal{M}_{\text{answer}}(I'), \tag{13}$$

where $I' = \{I'_t, I_v\}$ denotes the augmented multi-modal input, $I'_t = I_t \oplus R$ is the concatenation of the original textual input with the generated reasoning sequence. The conditional probability of predicting the answer is defined as:

$$\mathcal{P}(A \mid I_v, I_t) = \prod_{i=1}^{L} \mathcal{P}\psi'(A_i \mid I_v, I_t, A_{<i}), \tag{14}$$

where ψ' is the trainable parameters of the answer inference model, and $A_{<i}$ refers to the previously generated answer tokens. This two-stage pipeline promotes explicit reasoning, improving interpretability and accuracy.

IV. EXPERIMENTS

This study conducts experiments on two benchmarks for complex reasoning in visual question answering, ScienceQA [1] and A-OKVQA [26], and evaluates model performance using accuracy. More details about datasets and metrics can be seen in Appendix.

A. Experimental Setup

1) Datasets Re-annotation: To build the training corpus, the ScienceQA and A-OKVQA datasets are re-annotated using GPT-4o to generate structured four-stage rationales covering Summary, Caption, Reasoning, and Conclusion. Stage-wise explanations are produced with prompting templates and validated automatically. Samples with missing stages or inconsistent conclusions are removed, resulting in a clean and unified corpus for multi-stage reasoning training. Detailed prompting templates and examples are provided in Appendix.

2) Implementation Details: The semantic adapter is pre-trained alongside frozen CLIP image encoder [27] and a T5-based UnifiedQA language model [20], while the adapter itself is trained from scratch. We use a batch size of 16 for the base model and 8 for the large model, a learning rate of 5×10^{-5}, and a maximum sequence length of 512. The coefficient α

TABLE II
COMPARISONS ON THE A-OKVQA DATASET. BOLD AND UNDERLINED VALUES INDICATE **BEST** AND SECOND-BEST RESULTS.

Method	Model Size	Direct-answer	Multi-choice
Pythia [28]	70M	25.2	49.0
ViLBERT [5]	300M	30.6	49.1
KRISP [6]	200M	33.7	51.9
GPV-2 [29]	300M	48.6	60.3
BLIP-2 [12]	11B	53.2	**70.2**
PaLM-CoT [1]	540B	41.5	48.1
PICa [30]	175B	42.4	46.1
IPVR [9]	66B	46.4	48.6
Multimodal-CoT$_{Base}$ [2]	223M	-	50.6
Lite-CoT$_{Base}$ (**Ours**)	228M	**61.7**	66.0

TABLE III
IMPACT OF REASONING PATH QUALITY ON THE SCIENCEQA DATASET.

Model	Rouge-L (%)	Average Accuracy (%)
Multimodal-CoT$_{Base}$	96.97	84.91
Lite-CoT$_{Base}$	98.23	93.86
Lite-CoT$_{Large}$	98.86	95.59

TABLE IV
ABLATION STUDY ON THE SEMANTIC ADAPTER.

Method	Average Accuracy (%)	Δ (%)
w/ semantic adapter	93.86	-
w/ gated fusion	90.68	↓ 3.18

TABLE V
RESULTS UNDER VARYING NUMBERS OF REASONING PATHS ON THE SCIENCEQA DATASET.

Model	No. of Paths	Average Accuracy (%)
Multimodal-CoT$_{Large}$	0	91.68
Lite-CoT$_{Large}$	5	95.45
Lite-CoT$_{Large}$	10	95.59

TABLE VI
IMPACT OF STRUCTURED LABELS ON THE SCIENCEQA DATASET.

Method	Average Accuracy (%)	Δ (%)
w/ structured labels	93.86	-
w/o structured labels	87.05	↓ 6.81

is set to 0.5. All experiments are conducted on four NVIDIA Tesla V100 GPUs with 32GB of memory each.

B. Results and Analysis

Results on the ScienceQA Dataset. As shown in Table I, Lite-CoT$_{\text{Base}}$ uses only 228M parameters yet achieves an average accuracy of 93.86%. It not only surpasses LaVIN-13B [2] and LLaVa-13B [23], both of which require extensive fine-tuning of large language models, but also operates with just 1.75% of their parameter size. Furthermore, Lite-CoT$_{\text{Large}}$ achieves a 3.06% improvement in accuracy over the fine-tuned LLaVa+GPT-4 [23]. It also outperforms Honeybee [25], a multi-modal LLM method, by 1.2%. Compared to multimodal-CoT$_{\text{Base}}$ [2], Lite-CoT$_{\text{Base}}$ shows an 8.95% improvement, while Lite-CoT$_{\text{Large}}$ outperforms Multimodal-CoT$_{\text{Large}}$ by 3.91%.

Results on the A-OKVQA Dataset. As shown in Table II, Lite-CoT$_{\text{Base}}$ delivers competitive performance on both direct-answer and multiple-choice settings. For direct-answer evaluation, it attains an accuracy of 61.7%, surpassing the second-best BLIP-2, despite its over 11 billion parameters, by a substantial margin of 8.5%. In the multiple-choice task, it achieves the second-highest accuracy of 66.0% while using only 2.07% of the parameters of the top-performing BLIP-2 model. These results clearly demonstrate the superior parameter efficiency of Lite-CoT.

C. Ablation studies

1) Study on Reasoning Path Quality: The impact of reasoning path quality on the ScienceQA dataset is summarized in Table III. Rouge-L [31] is used to measure the similarity between generated rationales and reference explanations. Compared to Multimodal-CoT$_{\text{Base}}$, Lite-CoT$_{\text{Base}}$ achieves a 1.26% gain in Rouge-L and an 8.95% increase in answer accuracy. Although Lite-CoT$_{\text{Large}}$ surpasses the base version by only 0.63% in Rouge-L, its accuracy improves by approximately 2%. These observations indicate that the proposed stage-wise self-consistent strategy indeed improves the quality of rationales, and that even incremental enhancements in rationale quality can lead to meaningful gains in accuracy.

2) Study on Semantic Adapter: To validate the semantic adapter, we replace it with a gated fusion module and conduct a comparison on the ScienceQA dataset using Lite-CoT$_{\text{Base}}$. As shown in Table IV, the gated fusion lowers accuracy by 3.18%, demonstrating the advantage of semantic adapter in cross-modal alignment.

3) Impact of Multi-path: To examine how the number of intermediate reasoning paths affects model performance, two configurations are compared, generating 5 paths, which is the default setting, and generating 10 paths to assess potential benefits from higher diversity. As reported in Table V, increasing the number of paths yields only a slight improvement of 0.14%, from 95.35% to 95.59%. The small margin suggests that a larger pool of paths does not necessarily enhance performance, as many paths may be redundant or implausible. Moreover, producing 10 paths nearly doubles computational requirements due to additional generation and verification. Therefore, using 5 paths offers a more favorable balance between accuracy and efficiency.

4) Impact of Structured Labels: The contribution of structured labels—<SUMMARY>, <CAPTION>, <REASONING>, and <CONCLUSION>—is evaluated by comparing the full model with a variant trained without these markers. As shown in Table VI, removing the labels reduces accuracy

by 6.81%, indicating that explicit stage boundaries provide effective supervision and improve reasoning quality.

In addition, the Appendix provides the theoretical basis for the multi-path reasoning and voting strategy, as well as an analysis of challenging examples and a visual comparison of model accuracy versus model size. In addition, it offers a comparison of the enhanced CoT method and an analysis of computational complexity and efficiency.

V. Conclusions

This paper presented Lite-CoT, a lightweight framework aimed at improving robustness, interpretability, and efficiency in ScienceQA tasks. The approach incorporated stage-wise self-consistency voting to mitigate hallucinations and reasoning drift commonly observed in end-to-end generation. Additionally, a semantic adapter was introduced to enhance visual-text alignment, thereby strengthening multimodal representation learning. Experiments show that Lite-CoT achieves accuracy comparable to, and in some cases exceeding, substantially larger multimodal models while using only a fraction of their parameters, indicating that compact models with structured reasoning remain competitive in settings requiring computational efficiency. Evaluations on A-OKVQA demonstrate initial transferability, though broader scalability and cross-domain generalization require further study. The framework provides a technically grounded approach for efficient and interpretable multimodal reasoning.

References

[1] Pan Lu, Swaroop Mishra, Tony 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," in *NeurIPS*, 2022, pp. 1–15.

[2] Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola, "Multimodal chain-of-thought reasoning in language models," *TMLR*, vol. 4, no. 5, pp. 1–25, 2024.

[3] Sameera Horawalavithana, Sai Munikoti, Ian Stewart, Henry Kvinge, and Karl Pazdernik, "SCITUNE: Aligning large language models with human-curated scientific multimodal instructions," in *NLP4Science*, 2024, pp. 58–72.

[4] Guowei Xu, Peng Jin, Hao Li, Yibing Song, Lichao Sun, and Li Yuan, "Llava-cot: Let vision language models reason step-by-step," 2024.

[5] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *NeurIPS*, 2019, pp. 13–23.

[6] Kenneth Marino, Xinlei Chen, Devi Parikh, Abhinav Gupta, and Marcus Rohrbach, "Krisp: Integrating implicit and symbolic knowledge for open-domain knowledge-based vqa," in *CVPR*, 2021, pp. 14111–14121.

[7] Weixin Liang, Yanran Jiang, and Zhiyuan Liu, "Graphvqa: Language-guided graph neural networks for scene graph question answering," in *NAACL-HLT*, 2021, pp. 79–86.

[8] OpenAI, "Gpt-4 technical report," Technical report, OpenAI, 2023, Available at https://arxiv.org/abs/2303.08774.

[9] Zhenfang Chen, Qinhong Zhou, Yikang Shen, Yining Hong, Hao Zhang, and Chuang Gan, "See, think, confirm: Interactive prompting between vision and language models for knowledge-based visual reasoning," 2023.

[10] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou, "Self-consistency improves chain of thought reasoning in language models," in *ICLR*, 2023, pp. 1–24.

[11] Cheng Tan, Jingxuan Wei, Zhangyang Gao, Linzhuang Sun, Siyuan Li, Ruifeng Guo, Bihui Yu, and Stan Z. Li, "Boosting the power of small multimodal reasoning models to match larger models with self-consistency training," in *ECCV*, 2024, pp. 305–322.

[12] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *ICML*, 2023, pp. 19730–19742.

[13] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian, "Deep modular co-attention networks for visual question answering," in *CVPR*, 2019, pp. 6281–6290.

[14] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *CVPR*, 2018, pp. 6077–6086.

[15] Jun-Ho Kim, Jaehyeon Jun, and Byoung-Tak Zhang, "Bilinear attention networks," in *NeurIPS*, 2018, pp. 1571–1581.

[16] Peng Gao, Zhengkai Jiang, Haoxuan You, Pan Lu, Steven C. H. Hoi, Xiaogang Wang, and Hongsheng Li, "Dynamic fusion with intra- and inter-modality attention flow for visual question answering," in *CVPR*, 2019, pp. 6639–6648.

[17] Wonjae Kim, Bokyung Son, and Ildoo Kim, "Vilt: Vision-and-language transformer without convolution or region supervision," in *ICML*, 2021, pp. 5583–5594.

[18] Pan Lu, Liang Qiu, Jiaqi Chen, Tanglin Xia, Yizhou Zhao, Wenting Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu, "Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning," in *NeurIPS Datasets and Benchmarks*, 2021, pp. 1–14.

[19] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang, "What does bert with vision look at?," in *ACL*, 2020, pp. 5265–5275.

[20] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *JMLR*, vol. 21, no. 140, pp. 1–140, 2020.

[21] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei, "Language models are few-shot learners," in *NeurIPS*, 2020, pp. 1877–1901.

[22] Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao, "Chameleon: Plug-and-play compositional reasoning with large language models," in *NeurIPS*, 2023, pp. 1–32.

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee, "Visual instruction tuning," in *NeurIPS*, 2023, pp. 1–25.

[24] Yuyu Yin, Fangyuan Zhang, Zhengyuan Wu, Qibo Qiu, Tingting Liang, and Xin Zhang, "Pill: Plug into llm with adapter expert and attention gate," *Applied Soft Computing*, vol. 165, no. 11, pp. 112115, 2024.

[25] Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh, "Honeybee: Locality-enhanced projector for multimodal llm," in *CVPR*, 2024, pp. 13817–13827.

[26] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi, "A-okvqa: A benchmark for visual question answering using world knowledge," in *ECCV*, 2022, pp. 146–162.

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever, "Learning transferable visual models from natural language supervision," in *ICML*, 2021, pp. 8748–8763.

[28] Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal, "Pythia: A suite for analyzing large language models across training and scaling," in *ICML*, 2023, pp. 2397–2430.

[29] Amita Kamath, Christopher Clark, Tanmay Gupta, and Aniruddha Kembhavi, "Webly supervised concept expansion for general purpose vision models," in *ECCV*, 2022, pp. 662–681.

[30] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang, "An empirical study of gpt-3 for few-shot knowledge-based vqa," in *AAAI*, 2022, pp. 3081–3089.

[31] Chin-Yew Lin, "ROUGE: A package for automatic evaluation of summaries," in *ACL*, 2004, pp. 74–81.