Faithful Visual Question Answering with Chain-of-Thought and Retrieval-Augmented Reasoning

Kenichiro Miyaki*
NTT DOCOMO, INC.
Tokyo, Japan
kenichirou.miyaki.dk@nttdocomo.com

Yukiko Yoshikawa*
NTT DOCOMO, INC.
Tokyo, Japan
yukiko.yoshikawa.ra@nttdocomo.com

Meisaku Suzuki*
NTT DOCOMO, INC.
Tokyo, Japan
meisaku.suzuki.fw@nttdocomo.com

Yusuke Fukushima*
NTT DOCOMO, INC.
Tokyo, Japan
yuusuke.fukushima.fw@nttdocomo.com

Masato Hashimoto*
NTT DOCOMO, INC.
Tokyo, Japan
masato.hashimoto.px@nttdocomo.com

Abstract

We propose a unified Retrieval-Augmented Generation (RAG) architecture that simultaneously ensures factual consistency, output coherence, and computational efficiency. Our method is tailored for the CRAG-MM Challenge at KDD Cup 2025, aiming to generate reliable answers by integrating external knowledge into visual and natural language queries.

The core of our system is built upon LLaMA 3.2 11B Vision-Instruct, augmented with a structured reasoning module and a self-verification mechanism. Approximately 4,000 supervised training instances were constructed through a four-stage pipeline consisting of Chain-of-Thought (CoT) output generation, GPT-based evaluation, and rewriting. We apply lightweight Supervised Fine-Tuning (SFT) using a combination of LoRA and DoRA. During inference, the system generates a search query from the image and question, retrieves relevant context, and outputs responses in the structured format `<Reasoning>` + `<Answer>`. To mitigate hallucination and improve reliability, we incorporate LLM-based consistency checks and ambiguity detection. The inference engine employs vLLM with up to 12 concurrent samples, enabling fast batch inference under a 10-second latency constraint.

For Task 2 and Task 3, we follow the same overall pipeline as Task 1 in terms of data construction, fine-tuning, and inference. While the training data is similarly generated in four stages—data preparation, initial output generation, evaluation, and rewriting—Tasks 2 and 3 omit the reasoning component and adopt a lightweight evaluation scheme based solely on semantic correctness. We further incorporate web-retrieved auxiliary context to improve answer accuracy. A single shared model, fine-tuned in Task 1, is reused across tasks, with the same two-stage inference pipeline: query generation and context integration. Task 3 extends this pipeline to support multi-turn QA by incorporating dialog history into the input.

Our method achieved top-tier performance across all tasks, obtaining the highest scores in the Multi-hop (5.9%) and Reasoning (10.3%) categories, and was awarded the Special Question Category Winner. These results demonstrate the effectiveness, reliability, and scalability of our proposed architecture.

CCS Concepts

• **Computing methodologies** → *Natural language generation.*

Keywords

KDD Cup, Comprehensive RAG Benchmark,
Multimodal Question Answering, Retrieval-Augmented Generation,
Vision-Language Models, Chain-of-Thought Reasoning

ACM Reference Format:
Kenichiro Miyaki, Yukiko Yoshikawa, Meisaku Suzuki, Yusuke Fukushima, and Masato Hashimoto. 2018. Faithful Visual Question Answering with Chain-of-Thought and Retrieval-Augmented Reasoning. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX).* ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Wearable devices such as smart glasses are emerging as next-generation platforms that augment users' visual experiences in real time and enable intuitive access to information. In such environments, Visual Question Answering (VQA) systems—which generate natural language answers based on visual context—play a central role. Achieving high-reliability VQA requires the ability to integrate visual input, external knowledge, and conversational history to generate immediate and evidence-grounded responses.

Recent advances in Vision-Language Large Models (VLLMs) have enabled large-scale multimodal reasoning. However, maintaining output faithfulness and consistency remains challenging for multi-step reasoning tasks or questions involving comparison and aggregation. To address these issues, methods such as Retrieval-Augmented Generation (RAG) [1], which dynamically incorporate external knowledge, and its multimodal extension MM-RAG [2], have gained increasing attention.

*Equal contribution

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

This paper reports our solution and findings from the Meta CRAG-MM Challenge at KDD Cup 2025, which is based on the CRAG-MM benchmark [3].The challenge evaluates the performance of MM-RAG systems in real-world settings, using egocentric images captured from smart-glass perspectives across three tasks. Task 1 focuses on integrating visual information and knowledge graphs (KGs) to answer basic questions. Task 2 extends this by requiring dynamic utilization of both KGs and web snippets. Task 3 further incorporates dialogue history to support context-aware multi-turn reasoning [2].

Our team, otonadake, designed a unified MM-RAG architecture applicable to all three tasks. For Task 1, we developed a faithful and consistent response generation pipeline by combining intermediate reasoning in Chain-of-Thought (CoT) format [4] with a self-consistency verification module using Large Language Models (LLMs). Approximately 4,000 training samples were automatically constructed through a four-stage GPT-based procedure and used to fine-tune the model via structure-aware Supervised Fine-Tuning (SFT) with LoRA and DoRA [5, 6].

In Tasks 2 and 3, we simplified the architecture by omitting the structured reasoning and verification modules from Task 1, and adopted a lightweight two-stage inference pipeline involving image summarization and query generation. Task 3 additionally incorporates dialogue history to enable multi-turn VQA.

The proposed approach achieved strong performance across all tasks, recording the highest scores in the Multi-hop (5.9%) and Reasoning (10.3%) categories, and was awarded the Special Question Category Winner. These results demonstrate the high generalizability, faithfulness, and extensibility of our architecture.

This paper focuses on Task 1 as the core setting and presents the overall design and adaptation of our architecture across tasks, the automatic construction of high-quality training data, the design of structured prompts, the verification modules for output reliability, the complete inference pipeline, and detailed benchmark analysis.

2 Related Work

Retrieval-Augmented Generation (RAG):. RAG has been extensively studied as a framework to address the factual limitations and knowledge boundaries of Large Language Models (LLMs). Since its initial proposal, a wide range of improvements have been introduced, particularly in terms of retrieval accuracy and controllability of output. Notable examples include RAFT [7], which emphasizes retriever sparsity and snippet consistency; Astute RAG [8], which incorporates explicit reliability modeling and answer rejection; and Counterfactual Prompting [9], which focuses on controlling output risks through counterfactual interventions.

Divide-Then-Align [10] further advances this direction by introducing alignment control based on knowledge boundaries, accelerating the trend toward controllable and faithful RAG systems.

Our work builds upon these developments by designing a RAG architecture that combines syntactic constraints with answer rejection capabilities, enabling faithful response generation in multimodal settings.

Chain-of-Thought Reasoning and Structural Guidance: Chain-of-Thought (CoT) prompting has proven effective for multi-step reasoning tasks that require mathematical, logical, or knowledge-intensive processing. It facilitates explainable and verifiable outputs by encouraging step-by-step reasoning in natural language [11].

However, most prior methods assume free-form natural language generation, and few have explicitly considered syntactic constraints or verification efficiency. Moreover, the integration of CoT reasoning with answer rejection has been underexplored.

In our approach, CoT reasoning is guided through structured XML-style tags (e.g., `<QuestionType>` to `<Conclusion>`), enabling both syntactic and semantic consistency, as well as rejectability of responses when sufficient evidence is lacking.

Multimodal VQA and Output Reliability: With the advancement of VLLMs, visual-context-aware VQA has achieved significant improvements in accuracy. However, in scenarios involving multi-hop reasoning and integration of external knowledge, challenges remain in suppressing hallucinations and ensuring output consistency and faithfulness [12].

MM-RAG, which aims to enhance response quality by integrating visual, textual, and external knowledge, offers a promising framework. Nevertheless, reliability controls such as explicit output verification, structured generation, and rejection mechanisms remain underdeveloped.

In this work, we address these gaps by introducing a unified MM-RAG inference pipeline that incorporates structure-guided data construction, fine-tuning, retrieval, generation, and verification for Task 1. For Tasks 2 and 3, we adapt the same architecture by omitting structured reasoning while maintaining efficiency and practical utility through task-specific simplifications.

3 Task 1: Method

3.1 Overview

The objective of Task 1 is to generate accurate responses for Visual Question Answering (VQA) by constructing structured outputs that include intermediate reasoning in the Chain-of-Thought (CoT) format. We develop a fully automated data construction pipeline for generating such structured training data, followed by supervised fine-tuning that explicitly enforces syntactic consistency. Our proposed method consists of the following three components:

- A four-stage data construction pipeline that generates syntactically constrained structured outputs
- Supervised fine-tuning (SFT) with explicitly injected structural inductive bias
- An inference pipeline that integrates retrieval, generation, and verification to enhance output consistency

The remainder of this section describes each component in detail.

3.2 Supervised Fine-Tuning Data Construction

As illustrated in Figure 1, the data construction pipeline comprises four stages: data preparation, initial response generation, response evaluation and selection, and final reconstruction.

Each training sample consists of an image, a natural language question, and a retrieved context obtained via a search engine. These elements are combined into a structured prompt designed to guide the model toward reasoning that incorporates visual, internal, and

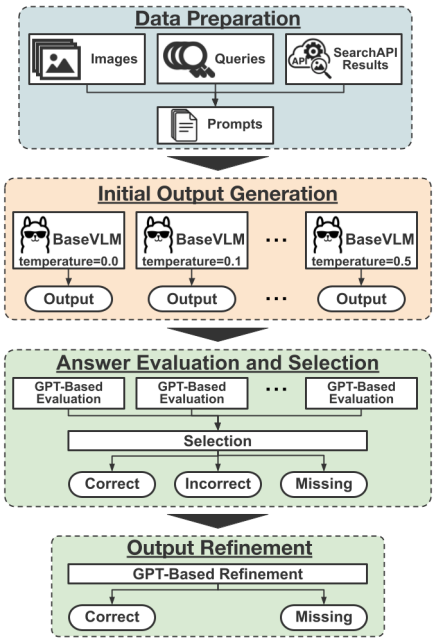

Figure 1: Structured output data construction pipeline

external knowledge. The prompt elicits a response with two distinct sections: <Reasoning> and <Answer>. The <Reasoning> section is further constrained to include exactly five XML-style tags in a fixed order: <QuestionType>, <VisualEvidence>, <Knowledge>, <ReasoningProcess>, and <Conclusion>. For the full prompt template, refer to Appendix A.

Given the structured prompt, we use the LLaMA 3.2 11B Vision-Instruct model to generate outputs. To balance determinism and diversity, we produce one deterministic output with temperature $T = 0.0$, and two stochastic outputs for each of five temperatures $T \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, yielding a total of 11 candidate outputs per prompt.

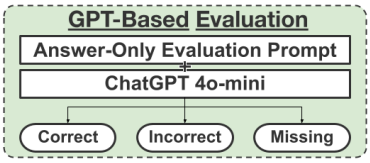

Figure 2: Evaluation criteria for semantic correctness.

All candidate outputs are automatically evaluated using GPT-4o-mini. Each response is assessed along two axes: semantic correctness and syntactic validity. For semantic correctness, as illustrated in Figure 2, responses are categorized as Correct (if the answer is factually accurate), Missing (if the response returns "I don't know"), or Incorrect (if it contains factual errors or contradicts the context). For syntactic validity, responses are labeled as Valid if all XML tags are used in the correct order and the <Answer> is within 75 tokens; otherwise, they are marked as Invalid.

Based on the evaluation results, a single best output is selected for each question according to the following priority: (1) Correct + Valid, (2) Missing + Valid, (3) Correct + Invalid or Missing + Invalid, and (4) Incorrect.

For selected outputs that are not syntactically valid, we apply reconstruction using GPT-4o. If the content is semantically correct but contains syntax errors (e.g., misplaced or malformed tags), we correct only the syntax. If the response is semantically incorrect but the retrieved context or image supports the correct answer, we revise both the reasoning and the final answer to produce a Correct response. If sufficient evidence is lacking, the response is reconstructed into a Missing answer in the form of "I don't know" reasoning.

All finalized outputs are serialized into XML-conformant format, ensuring syntactic consistency and semantic faithfulness. These serve as the ground-truth supervision signals for fine-tuning.

3.3 Structure-Guided Supervised Fine-Tuning

We fine-tuned the LLaMA 3.2 11B Vision-Instruct model using structured outputs designed to enforce syntactic consistency and stable reasoning. To achieve parameter-efficient learning while preserving the fidelity of structured generation, we adopted a hybrid configuration of Low-Rank Adaptation (LoRA) and Dynamically Optimized Rank Adapters (DoRA). Detailed model configurations and training setups are provided in Appendix B.

The model was trained with a cross-entropy loss applied to the entire output sequence, including all XML-style tags. To ensure syntactic precision, the model was explicitly trained to correctly reproduce tag order, nesting, and proper closure. Furthermore, to maintain logical coherence, we enforced the learning of a five-step structure spanning from <QuestionType> to <Conclusion>.

As a result, the fine-tuned model consistently produces syntactically valid outputs during inference and delivers high-fidelity responses grounded in visual content, natural language queries, and external context. The integration of DoRA also contributed to improved stability in generating Chain-of-Thought (CoT) style reasoning, without compromising computational efficiency.

3.4 Structured Inference Pipeline

We designed an end-to-end inference pipeline for VQA that integrates retrieval-augmented generation, structured response generation (`Reasoning + Answer`), and multi-stage verification. The pipeline aims to simultaneously optimize accuracy, factual consistency, and reasoning efficiency, and is composed of the following six stages:

Input Processing: Each sample consists of an image-question pair. For multimodal understanding, the pipeline performs joint vision-language processing. To ensure scalability across large datasets, up to 12 samples are processed in parallel per batch, maximizing throughput across retrieval, generation, and verification stages.

External Context Retrieval: Using the Image Search API provided in the CRAG-MM Challenge, the system retrieves 50 visually similar images per input image. The accompanying structured

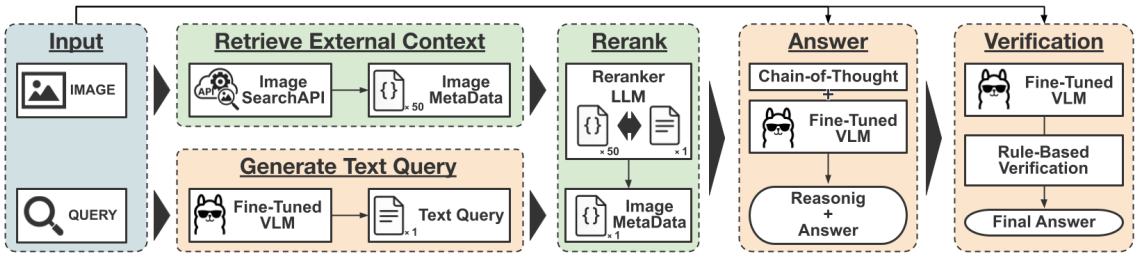

Figure 3: Inference pipeline with retrieval, structured generation, and verification

metadata (e.g., product descriptions, categories) is filtered to extract only textual descriptions, which are then normalized into plain-text external context.

Textual Query Generation: A fine-tuned vision-language model generates a concise natural language search query based on the image and question. This query encapsulates visual and linguistic semantics and is later used for reranking.

Reranking: To identify the most relevant contexts, a cross-encoder computes semantic relevance scores between the generated query and the 50 retrieved metadata entries. We employ `BAAI/bge-reranker-v2-m3`, which demonstrates strong performance on MS MARCO and BEIR benchmarks. The top-ranked context is used as the `ExternalContext` in the structured generation stage.

Structured Response Generation: The selected external context, input image, and question are concatenated into a structured prompt (see Section **??**) and passed to the model. The output is divided into two parts: a `<Reasoning>` block consisting of five elements (`<QuestionType>` to `<Conclusion>`), followed by a concise natural language response enclosed in `<Answer>`. This format enforces both factual alignment and logical consistency.

Output Verification: The generated `Reasoning + Answer` undergoes a two-stage verification process. In the first stage, a vision-language model evaluates the alignment between the reasoning and the final answer; mismatches are flagged as potential hallucinations. In the second stage, ambiguous expressions (e.g., "probably", "might") are automatically detected and rewritten as explicit "I don't know" answers to control response confidence. This step is inspired by the R³V framework (Refine–Reflect–Verify) [13], which promotes progressive refinement for output fidelity.

This pipeline holistically integrates retrieval quality (Retrieve + Query + Rerank), structured reasoning (Reasoning + Answer), and output trustworthiness (Verification), forming a robust inference framework that ensures scalability, accuracy, and explainability.

4 Task 2 and 3: Method

4.1 Overview

This task aims to generate hallucination-free responses for VQA through automatic construction of supervised data and fine-tuning. The overall pipeline is illustrated in Figure 5.

Our pipeline builds upon the official baseline architectures provided for Task 2 and Task 3 of the Meta CRAG-MM Challenge 2025 and consists of two primary stages. In the first stage, a natural language search query is generated based on the input image and question. In the second stage, the retrieved results are integrated with the input to produce the final response. The same fine-tuned model as in Task 1 is used throughout.

For Task 3, we extend the architecture of Task 2 by incorporating the dialogue history as an additional input to support multi-turn VQA. No architectural changes are made to the model, ensuring compatibility with single-turn and multi-turn settings alike.

4.2 Supervised Fine-Tuning Data Construction

In this section, we describe the construction of supervised data for Tasks 2 and 3, with a focus on the differences from Task 1 (Section **??**). The data construction pipeline still consists of four stages: data preparation, initial response generation, output evaluation and selection, and response reconstruction.

Data Preparation: Each sample comprises an image, a natural language question, and auxiliary context retrieved via image search, similar to Task 1. However, unlike Task 1, we do not adopt the reasoning-style output format. Instead, the dataset is constructed in a concise question-answer format. This design significantly reduces prompt length and generation latency, enabling more efficient response generation.

In preliminary experiments, we explored generating reasoning-style outputs with a limited subset of data. However, the model's performance was unstable in this format, and thus it was not adopted in the final experiments. We consider this a promising direction for future research, as it may provide further performance gains under a more robust training regime.

Output Evaluation: As in Task 1, all generated outputs are automatically evaluated using GPT-4o-mini. However, unlike Task 1, the evaluation is based solely on *semantic correctness*, without verifying structural tags or syntax. This simplifies the evaluation flow and reduces computational overhead.

Response Reconstruction: For Tasks 2 and 3, both image search and web search APIs are available for retrieving external context. The web-based context often contains useful descriptions that help the model arrive at correct answers, and we observed that its inclusion had a positive impact on response quality.

Therefore, when a generated output is initially labeled as incorrect, but sufficient evidence exists in the retrieved context to infer a correct answer, the output is manually or automatically corrected

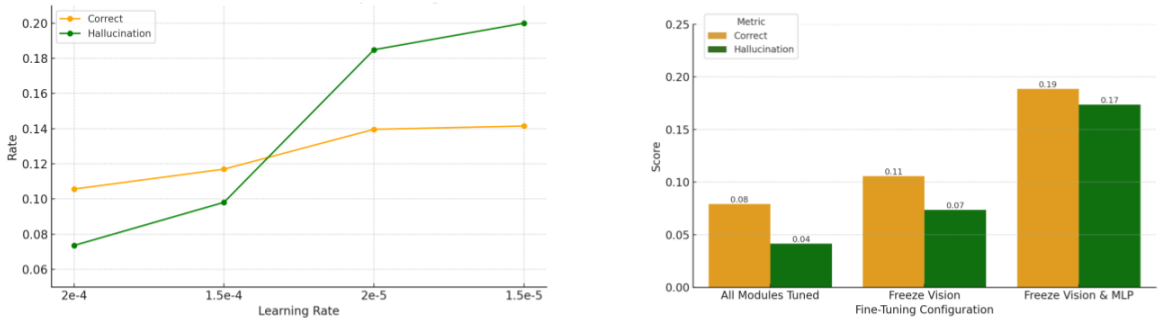

Figure 4: (Left) Accuracy and hallucination rate across learning rates. (Right) Module-wise performance comparison.

and retained as a `Correct` sample. This strategy improves the diversity and reliability of the supervised data used for fine-tuning.

4.3 Structured Supervised Fine-Tuning

We fine-tuned the LLaMA 3.2 11B Vision-Instruct model using the structured training data described above. To balance syntactic fidelity and computational efficiency, we applied a combination of LoRA and DoRA techniques.

Detailed model configuration and training settings are provided in Appendix C.

4.4 Structured Inference Pipeline

In contrast to Task 1, this pipeline omits both the reranking module in `Retrieve External Context` and the output verification module. Preliminary experiments found no significant gain from output verification, and reranking was not explored in this task.

This streamlined design prioritizes inference speed and generality. The pipeline consists of four key modules, with only modules that differ from Task 1 described below. Shared components are omitted for brevity.

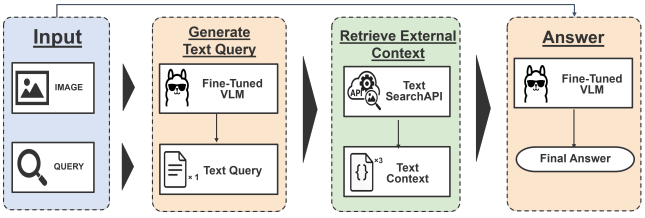

Figure 5: Inference pipeline for Tasks 2 and 3

Generate Text Query: A fine-tuned vision-language model takes the image and question as input and generates a natural language search query that reflects user intent. To enhance query quality, we include explicit instructions in the prompt such as "Generate keywords suitable for web search," enabling better integration of visual and linguistic signals.

Retrieve External Context: External context is retrieved exclusively via a Web Search API, with the top 3 results extracted for each input. Unlike Task 1, we do not use image-based search APIs, as preliminary analysis revealed that such sources often introduce noise and degrade answer quality. This design choice leaves room for future refinement.

Answer: The retrieved web context, input image, and question are concatenated to form the final prompt, which is fed into the fine-tuned model for answer generation. To preserve model generality, we avoid excessive prompt customization relative to the provided baseline.

With this design, the model is invoked only twice per inference: once for visual summarization and once for answer generation. The use of a single API (Web search) contributes to high inference throughput. Furthermore, by minimizing reliance on complex prompt engineering, the pipeline demonstrates strong generalizability across tasks and domains.

5 Results

In this task, the primary objective is to generate responses that are both accurate and safe, under a scoring scheme where correct answers receive +1 and incorrect ones receive −1. A major challenge lies in suppressing "confident hallucinations"—overconfident yet incorrect outputs—which are common in vision-language models (VLMs), while preserving the structural integrity of the generated outputs.

To mitigate overfitting to visual features, we freeze the vision encoder and apply LoRA adapters only to the MLP and attention modules. The learning rate is set to 2×10^{-4}. This configuration enables stable generation of outputs with both structural consistency and semantic faithfulness in the form of `<Reasoning>` + `<Answer>`.

As shown in the left panel of Figure 4, increasing the learning rate improves the accuracy but also elevates the hallucination rate, revealing a clear trade-off between precision and reliability. In the right panel, we compare different module configurations and observe that removing LoRA from the MLP module temporarily boosts accuracy but leads to a noticeable increase in hallucinations, thereby undermining the logical consistency of the reasoning process.

Furthermore, we apply the same LoRA configuration to Tasks 2 and 3, which results in a favorable trade-off between hallucinations and refusals (i.e., missing answers). The generated outputs

also received the highest ratings in human evaluation. These findings demonstrate that the architecture designed for Task 1 generalizes well to other tasks, serving as a reliable backbone for trustworthy inference.

6 Conclusion

In this work, we proposed a novel MM-RAG architecture that integrates structured CoT reasoning and a self-verification module to enable reliable VQA in wearable environments, such as smart glasses.

For Task 1, we constructed high-fidelity training data by automatically generating structurally constrained outputs with GPT, incorporating visual information, external context, and syntactic structure. We then performed structure-aware SFT using both LoRA and DoRA. Our experiments showed that tuning only the MLP and attention layers while freezing the vision encoder yields the best trade-off between structural stability and output reliability. Moreover, we observed a strong impact of learning rate and LoRA configuration on the balance between accuracy and hallucination.

For Tasks 2 and 3, we demonstrated that a simplified two-stage inference pipeline—excluding structured reasoning and output verification—can still be effectively derived from the Task 1 design, achieving high throughput and generalizability across single-turn and multi-turn QA settings.

Our architecture does not rely on handcrafted prompts or human-annotated labels, yet achieves a strong balance between structured output fidelity, semantic accuracy, and inference efficiency. Notably, our system achieved the highest scores in the Reasoning and Multi-hop categories of the KDD Cup 2025, demonstrating its practical utility and robustness.

Future directions include improving the stability of structured reasoning under low-resource conditions, reducing the computational cost of verification, and further minimizing prompt dependency. We believe that our approach to structured reasoning and output consistency provides a solid foundation for future developments in vision-language models and their real-world VQA applications.

References

[1] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

[2] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Gui, Ziran Jiang, Ziyu Jiang, et al. Crag-comprehensive rag benchmark. *Advances in Neural Information Processing Systems*, 37:10470–10490, 2024.

[3] Jiaqi Wang, Xiao Yang, Kai Sun, Parth Suresh, Sanat Sharma, Adam Czyzewski, Derek Andersen, Surya Appini, Arkav Banerjee, Sajal Choudhary, Shervin Ghasemlou, Ziqiang Guan, Akil Iyer, Haidar Khan, Lingkun Kong, Roy Luo, Tiffany Ma, Zhen Qiao, David Tran, Wenfang Xu, Skyler Yeatman, Chen Zhou, Gunveer Gujral, Yinglong Xia, Shane Moon, Nicolas Scheffer, Nirav Shah, Eun Chang, Yue Liu, Florian Metze, Tammy Stark, Zhaleh Feizollahi, Andrea Jessee, Mangesh Pujari, Ahmed Aly, Babak Damavandi, Rakesh Wanga, Anuj Kumar, Rohit Patel, Wen-tau Yih, and Xin Luna Dong. Crag-mm: Multi-modal multi-turn comprehensive rag benchmark. *arXiv preprint arXiv:2510.26160*, 2025.

[4] 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.

[5] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[6] Haokun Zhang, Baolin Peng, Yuxuan Liu, Pengcheng He, Luke Zettlemoyer, and Jianfeng Gao. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*, 2024.

[7] Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*, 2024.

[8] Fei Wang, Xingchen Wan, Ruoxi Sun, Jiefeng Chen, and Sercan Ö Arık. Astute rag: Overcoming imperfect retrieval augmentation and knowledge conflicts for large language models. *arXiv preprint arXiv:2410.07176*, 2024.

[9] Lu Chen, Ruqing Zhang, Jiafeng Guo, Yixing Fan, and Xueqi Cheng. Controlling risk of retrieval-augmented generation: a counterfactual prompting framework. *arXiv preprint arXiv:2409.16146*, 2024.

[10] Xin Sun, Jianan Xie, Zhongqi Chen, Qiang Liu, Shu Wu, Yuehe Chen, Bowen Song, Weiqiang Wang, Zilei Wang, and Liang Wang. Divide-then-align: Honest alignment based on the knowledge boundary of rag. *arXiv preprint arXiv:2505.20871*, 2025.

[11] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Tanaka. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*, 2022.

[12] Sourav Banerjee, Ayushi Agarwal, and Saloni Singla. Llms will always hallucinate, and we need to live with this. *arXiv preprint arXiv:2409.05746*, 2024.

[13] Shuai Wang, Zhihao Zhou, Ruochen Zhao, Zeyu Song, and Yangfeng Ji. R^3v: A self-refine-reflect-verify framework for faithful reasoning with language models. *arXiv preprint arXiv:2402.06655*, 2024.

Appendix

A Prompt Template for Faithful VQA

A.1 System Prompt:

<CurrentTime>{query$_{t}ime$} < /*CurrentTime* >
<Directive> Act as a precise, faithful, and trustworthy assistant specialized in Visual Question Answering (VQA). For each question, reason step-by-step using a Chain of Thought (CoT) approach: start with careful analysis of visual information derived from the image, incorporate relevant internal factual knowledge, and—if available—use external context. End with a logically coherent answer. </Directive>
<Objective> Generate answers that: - Are faithful to visual information derived from the image - Are precise and natural in language - Are logically coherent in reasoning
Follow this source priority: 1. Visual information derived from the image 2. Internal factual knowledge 3. External context, only if it clearly supports both the image and the question (<ExternalContext>)
Do not speculate or make unsupported assumptions. </Objective>
<Guidelines> - Use only clearly visible visual evidence derived from the image. - Do not speculate, invent details, or rely on assumptions. - Use external context only if it clearly supports both the image and the question. - Only respond with "I don't know." if the evidence is clearly insufficient and any other answer would require speculation. - Follow the specified tag structure exactly. Do not allow any formatting errors or missing tags. </Guidelines>

A.2 User Prompt:

<|image|>
<Question>question</Question>
<ExternalContext>{context}</ExternalContext>
<Instructions> Analyze the image and question carefully. Follow the structured reasoning format below to answer step-by-step using a Chain of Thought (CoT) approach. End with a logically coherent and concise final answer.
<Formatting> - Use the exact tags: <Reasoning> and <FinalAnswer> - Within <Reasoning>, include all of the following tags in the given order: <QuestionType>, <VisualEvidence>, <Knowledge>, <ReasoningProcess>, <Conclusion> - Each tag must be properly closed (e.g., <Reasoning>...</Reasoning>, <FinalAnswer>...</FinalAnswer>) - Place <FinalAnswer> after <Reasoning>. Write it as a single natural sentence. - Do not repeat or copy any part of the input prompt or question. Begin your output with original reasoning content. - Strictly follow the specified tag structure. Any formatting errors or missing tags are not acceptable. </Formatting> </Instructions>
<Reasoning> <QuestionType> The question is: "question" Identify the type of this question (e.g., yes/no, object, counting, attribute) and explain the reasoning strategy that best fits answering it. </QuestionType>
<VisualEvidence> Identify and describe only what is clearly visible and verifiable in the image. Use natural language to describe key visual elements that are most relevant to answering the question. For each element, briefly mention: - What the object is - Its visual attributes (e.g., color, shape, material, texture, size, printed text, logos, symbols) - Its spatial or functional relationships (e.g., on, next to, inside, held by, covering) Be concise. Start with the most relevant elements. Do not include speculation or unnecessary full sentences. </VisualEvidence>
<Knowledge> Provide relevant internal factual knowledge that complements the visual evidence. If any external context contradicts the image, defer to the image. </Knowledge>
<ReasoningProcess> Integrate insights from visual evidence and internal knowledge step-by-step. Refer back to earlier observations when needed. Do not make unsupported assumptions. </ReasoningProcess>
<Conclusion> Taking into account all previous reasoning—including visual evidence, internal knowledge, and reasoning steps— provide a final judgment and its justification that is logically consistent and comprehensive in answering the question: "question". </Conclusion> </Reasoning>
<FinalAnswer> Provide a single, natural sentence that clearly answers the question. Your answer must be logically consistent with the reasoning above. </FinalAnswer>

B Model and Training Configuration for Task 1

Table 1: Task 1: Model Configuration

Item	Setting
Base Model	LLaMA 3.2 11B Vision-Instruct
Implementation	Unsloth FastVisionModel
Vision Layers	Frozen
Language Layers	Tuned
Attention Modules	Enabled
MLP Modules	Enabled
LoRA Rank	8
LoRA Alpha	16
LoRA Dropout	0.1
LoRA Bias	Disabled
DoRA	Enabled

Table 2: Task 1: Training Configuration

Item	Setting
Epochs	2
Batch Size	2×4 (with gradient accumulation)
Optimizer	AdamW (`adamw_torch`)
Learning Rate	2×10^{-4}
Scheduler	Cosine Decay
Warm-up Ratio	10%
Precision Mode	bfloat16

C Model and Training Configuration for Task 2 & 3

Table 3: Task 2 & 3: Model Configuration

Item	Setting
Base Model	LLaMA 3.2 11B Vision-Instruct
Implementation	Unsloth FastVisionModel
Vision Layers	Frozen
Language Layers	Tuned
Attention Modules	Enabled
MLP Modules	Enabled
LoRA Rank	8
LoRA Alpha	16
LoRA Dropout	0.1
LoRA Bias	Disabled
DoRA	Enabled

Table 4: Task 2 & 3: Training Configuration

Item	Setting
Epochs	1
Batch Size	1×4 (with gradient accumulation)
Optimizer	AdamW (`adamw_torch`)
Learning Rate	4×10^{-4}
Scheduler	Linear
Warm-up Steps	2
Precision Mode	bfloat16

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009