Improving Truthfulness in Multimodal RAG: A Dual-Adapter Vision Large Language Model Approach

Hiroki Yamamoto
yamamoto@acroquest.co.jp
Acroquest Technology Co., Ltd.
Japan

Shin Higuchi
higuchi@acroquest.co.jp
Acroquest Technology Co., Ltd.
Japan

Takashi Sasaki
sasaki@acroquest.co.jp
Acroquest Technology Co., Ltd.
Japan

Shun Yoshioka
s_yoshioka@acroquest.co.jp
Acroquest Technology Co., Ltd.
Japan

Abstract

Trustworthy multimodal question answering requires systems that can fuse visual evidence, external knowledge, and dialog history to give helpful answers for wide variety of situations while avoiding to give falsy answers when the data is insufficient or uncertain. The Meta Comprehensive RAG Multimodal (CRAG-MM) Challenge 2025 evaluates this setting across staged tasks combining wearable egocentric imagery (including Ray-Ban Meta smart-glasses), image and web retrieval, and multi-turn interaction. We present the AcroYAMALEX system, built on Llama 3.2 Vision Instruct with a two-stage adapter architecture: (i) a *Retrieval-Oriented* LoRA adapter that first produces a concise provisional answer, which we repurpose as a high-precision text query for downstream search; and (ii) an *Answer-Generation* LoRA adapter trained with uncertainty-aware relabelling so the model outputs "I don't know" instead of hallucinating under weak evidence. Retrieved web snippets are chunked and reranked with Qwen3-Reranker-0.6B to provide focused context before final answer generation. In Task 2 (multi-source RAG), our approach contributed to a **3rd-place** final ranking. These results suggest that coupling abstention training with deliberate query construction and neural reranking improves factual reliability in multimodal RAG systems.

Keywords

RAG, VLM, Hallucination

ACM Reference Format:
Hiroki Yamamoto, Shin Higuchi, Takashi Sasaki, and Shun Yoshioka. 2025. Improving Truthfulness in Multimodal RAG: A Dual-Adapter Vision Large Language Model Approach. In *Proceedings of johe (Conference acronym 'KDD)*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference acronym 'KDD, Toronto, Canada

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

1.1 Challenge Background

Large Language Models (LLMs) have advanced rapidly in recent years; nevertheless, hallucination—the tendency to produce factually incorrect or ungrounded content—remains a serious obstacle in trustworthy deployment. One promising direction for mitigating hallucinations is Retrieval-Augmented Generation (RAG) [7] , in which a system searches external knowledge sources and conditions its response on retrieved evidence. In 2024, the Meta Comprehensive RAG Challenge (CRAG 2024) [15] was organized to evaluate RAG system performance and hallucination robustness in a text-based question answering setting. By spanning diverse domains, question types, and factual dynamism, CRAG 2024 exposed key limitations of current RAG pipelines under realistic conditions.

Subsequent technological progress—in particular the maturation of Vision-Language Large Models (VLLMs) and the growing availability of wearable devices such as smart glasses—has made it increasingly realistic to extract knowledge multimodally from what a user sees and deliver context-aware assistance in real life situation. However, substantial challenges remain unresolved: robust image understanding (object recognition and scene context), Optical Character Recognition (OCR), long-tail knowledge completion, integration with external search, and ensuring factual consistency while suppressing hallucinations in generated answers. A standardized benchmark is needed to reliably evaluate such multi-modal RAG (MM-RAG) systems.

Responding to this need, the Meta CRAG-MM Challenge 2025 (CRAG-MM) [12] [13] extends CRAG into the visual domain. The competition provides a factuality-oriented visual question answering (VQA) benchmark tailored to wearable use cases, comprising roughly 5,000 images—about 3,000 of which are egocentric, first-person captures from Ray-Ban Meta smart glasses—paired with question–answer data. The dataset spans 13 domains, and includes four question types ranging from simple, image-answerable recognition queries to complex questions requiring external knowledge search, multi-source integration, and reasoning. In addition, both single-turn and multi-turn conversational formats are supported, enabling staged evaluation of MM-RAG system capabilities.

1.2 Problem Statement

CRAG-MM Challenge has 3 tasks.

Task 1: Single-source Augmentation
Given a single image and a single-turn question as input, the system must generate an answer to the given question. As an external knowledge source, it may use *only* a mock image–retrieval API keyed by the input image.

Task 2: Multi-source Augmentation
Building on Task 1, the system must construct a retrieval-augmented generation (RAG) pipeline that also leverages a mock Web-search API in addition to the image API.

Task 3: Multi-turn Question Answering
The system must handle conversations of 2–6 turns, producing answers that consider the entire dialogue history within a RAG framework.

Evaluation Protocol
All three tasks share a common metric. For every response, a *Truthfulness Score* is assigned according to the rubric below; the final score is the average across all evaluation instances:

- **Perfect (1.0):** Fully correct answer.
- **Acceptable (0.5):** Useful answer with only minor, non-harmful errors.
- **Missing (0.0):** The system replies "I don't know."
- **Incorrect (−1.0):** Wrong or irrelevant answer.

In this paper, we describe the solutions our team, AcroYAMALEX, developed primarily for Task 2 and Task 3.

2 Related Work

Our approach lies at the intersection of two research streams: (1) multimodal RAG pipelines that transform visual inputs into retrieval-ready textual queries, and (2) techniques for mitigating hallucination and enabling explicit abstention ("I don't know") in large language and vision–language models. We first review frameworks that couple image understanding with external knowledge retrieval, highlighting how prior work structures the path from images to search queries. We then discuss studies on hallucination reduction and refusal strategies, clarifying how our dual-adapter design differs by isolating query generation from answer production and by training explicit abstention behavior.

2.1 Multimodal RAG Pipelines: From Images to Search Queries

As prior work that couples image understanding with external knowledge retrieval, REVEAL [6] incorporates retrieval into vision–language pretraining, proposing a framework that acquires external textual evidence to compensate for ambiguous elements in images. MM-ReAct [16] extends ReAct-style thought-and-action prompting, demonstrating a step-by-step procedure that invokes tools (e.g., search) using both images and text. VisProg [4] and ViperGPT [11] share the idea of treating visual reasoning as program generation and code execution, leveraging external resources in a staged reasoning process when necessary.

While these approaches share a high-level flow of "image → intermediate representation → retrieval → integrated answer," many stop at directly generating image captions or code fragments, without deeply analyzing the design of search query optimization (e.g., target granularity, top-K control, chunking strategies). In contrast, our work explicitly separates "image → search-query generation" into a dedicated LoRA adapter (the Retrieval-Oriented Adapter) and assigns complementary roles to the downstream answer-generation adapter.

2.2 Hallucination

Hallucination mitigation and answer abstention for LLMs/VLMs have been discussed in works such as SelfCheckGPT [9] and R-Tuning [17], but most studies focus primarily on text-only models. The winning solutions of CRAG2024 also addressed hallucination; notably, team db3 trained their model to output "I don't know" using LoRA. Building on these insights, our work differentiates itself by separating a retrieval-oriented adapter that extracts search-relevant information from images and an answer-generation adapter, and by explicitly training the model to output "I don't know" under uncertainty.

3 Solution Overview

Our solution couples a *risk-aware dual-adapter* version of Llama 3.2 Vision Instruct 11B with a lightweight retrieval stack. During training, we build a supervision set in which a GPT-4.0 mini verifier rewrites every erroneous prediction from the base model to "I don't know," then fine-tune two QLoRA adapters: one that turns an *(image, question)* pair into a concise search query and a second that generates the final answer—or "I don't know" when confidence is low. At inference time, the query adapter produces the search text, the official CRAGSearch API returns the top web snippets, a Qwen3-Reranker-0.6B re-orders them, and the answer adapter delivers a concise response while safely abstaining under uncertainty. This design earned 3rd place in CRAG-MM 2025 Task 2 with both high accuracy and a low penalty rate.

4 Training

4.1 Overview

Figure.1 is training step overview. The goal of our training pipeline is to fine-tune Llama 3.2 Vision Instruct 11B [1] so that the model outputs "I don't know" whenever it is uncertain or likely to make an error. During dataset construction, we first run inference with the base model; inference result whose confidence falls below a threshold are automatically relabelled with "I don't know". Subsequently, we apply Low-Rank Adaptation (LoRA) [5] (When training, we use QLoRA [3]) to create two separate adapters on top of the shared base model.

(1) Retrieval-Oriented Adapter: generates a preliminary answer or query that conditions downstream retrieval.
(2) Answer-Generation Adapter: produces the final answer; samples labelled "I don't know" are learned verbatim so the model can abstain under high uncertainty.

4.2 Dataset Creation

The competition metric imposes a severe penalty for incorrect answers, it is crucial to return "I don't know" on ambiguous items.

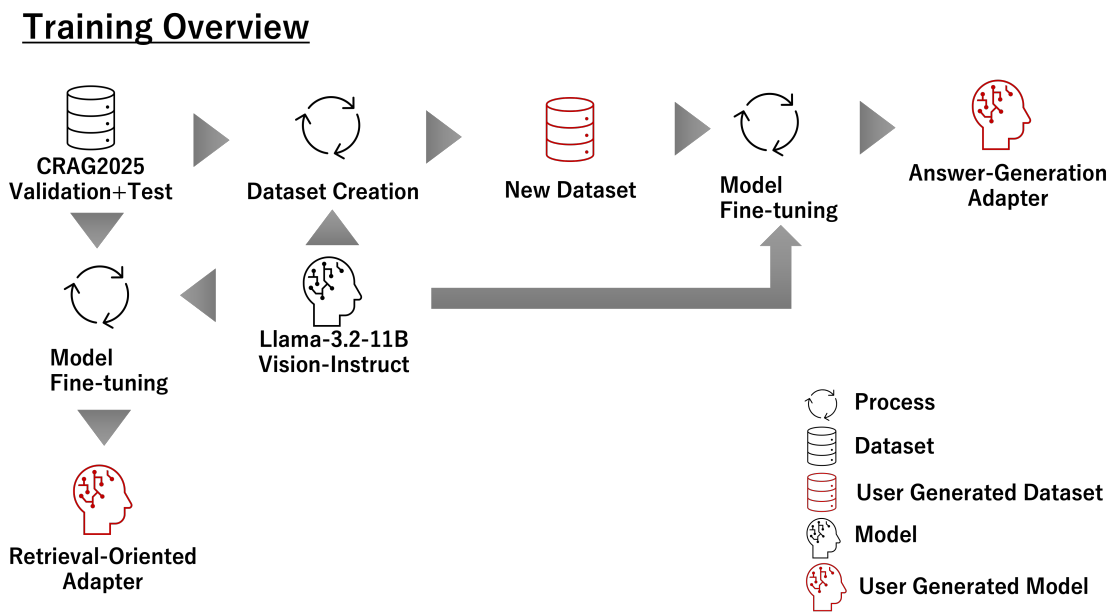

Figure 1: Training Overview

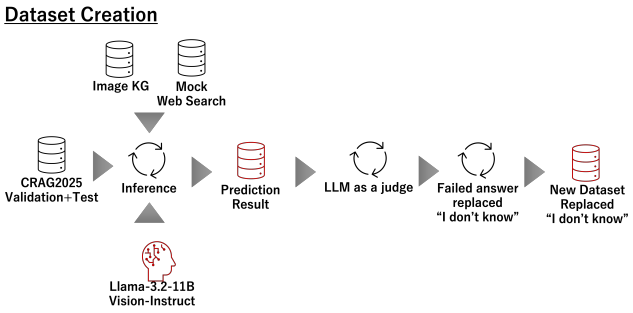

Figure 2: Dataset Creation Flow

Figure.2 is dataset creation process overview. We reconstructed the dataset by following the methodology of the CRAG 2024 winning team [14] . We build training data and construct labels in two steps:

(1) Inference with Retrieval: We run Llama 3.2 11B Vision Instruct while injecting the outputs of our retrieval pipeline, mirroring the RAG setting at evaluation time.
(2) LLM-as-a-Judge Relabelling: A GPT-4.0 mini [10] verifier reviews each prediction and replaces incorrect answers with "I don't know".

The resulting dataset is then used for fine-tuning as described below.

4.3 Model Fine-tuning

Using the curated dataset,we train two LoRA adapters:

Retrieval-Oriented Adapter. Given an image and a question, the adapter predicts a concise answer that serves as a high-precision retrieval query. We train this module with supervised fine-tuning (SFT) on the ground-truth answers.

Answer-Generation Adapter. For the final answer we perform SFT using the labels that include "I don't know". Answer-Generation Adapter directly output "I don't know". This enables the model to abstain whenever the question or the retrieved evidence remains unreliable, thereby mitigating the penalty for wrong answers.

Together, these two adapters allow the system to retrieve relevant evidence effectively while remaining robust under uncertainty.

5 Inference

5.1 Overview

At test time we transform each *(image, question)* pair into a final answer through a four-stage pipeline, illustrated in Figure.3

(1) **Image–to–Query Conversion.** A Retrieval-Oriented Adapter trained with LoRA first converts the image into a succinct textual description that captures the visual evidence needed to answer the question.
(2) **Web-Scale Retrieval.** The generated description, optionally concatenated with the user's question and dialogue history, is passed to the official CRAGSearch API, which returns the top-k web snippets likely to contain the answer. We set $k = 20$ in all experiments.
(3) **Passage Reranking.** Retrieved snippets are used. Qwen3-Reranker-0.6B [18] scores each passage against the (question, image-summary) pair, and we keep the top-X passages; X is a tunable hyper-parameter.
(4) **Answer Generation.** Finally, the selected passages, the original question, the image, and the chat history are fed

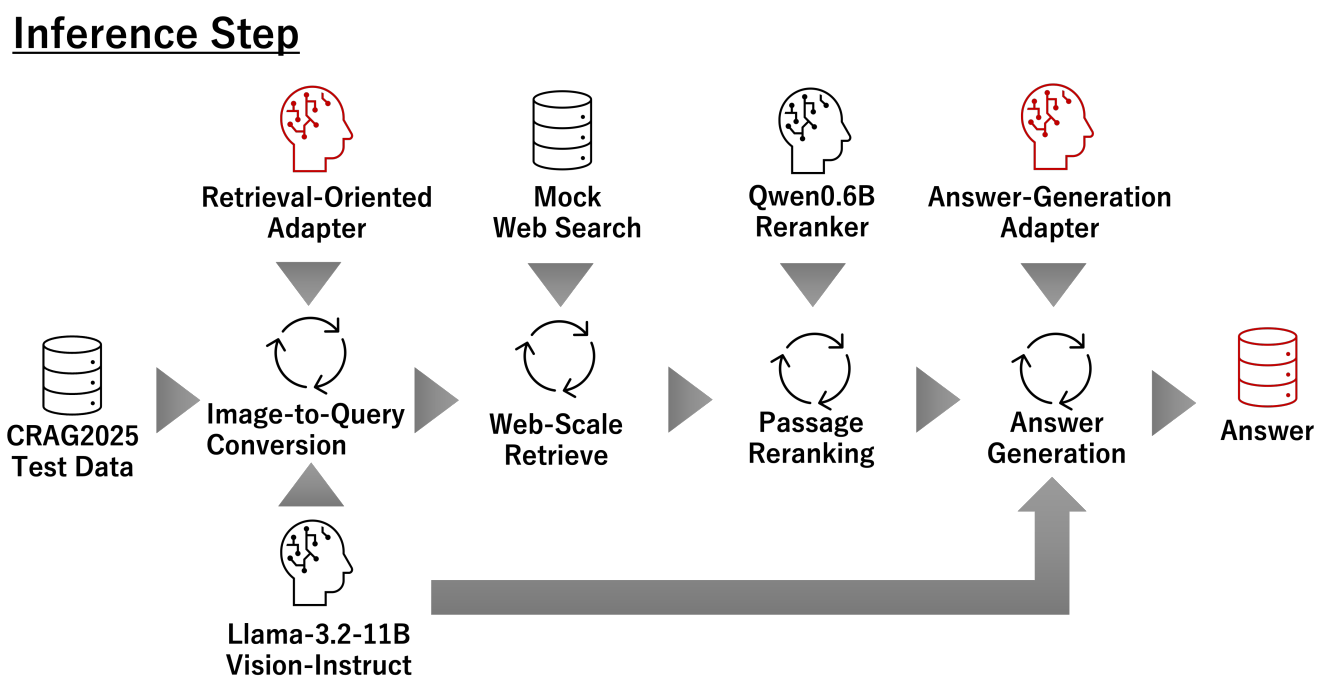

Figure 3: Inference Overview

into Llama 3.2 Vision Instruct equipped with an Answer-Generation Adapter. The model outputs a concise answer or "*I don't know*" when its confidence falls below a threshold.

This design cleanly separates evidence gathering from reasoning: visual grounding happens once in Step 1, text retrieval is handled by lightweight neural and symbolic components in Steps 2–3, and multimodal reasoning is reserved for Step 4, greatly reducing inference latency while preserving answer quality.

5.2 Image–to–Query Conversion

The competition provides two mock interfaces: an Image-KG module that accepts images and a Web-search module that accepts text. Because these interfaces cannot ingest the question and its corresponding image simultaneously, we must first extract the textual query needed for search directly from the image. Using the previously built Retrieval-Oriented Adapter LoRA module together with the prompt below, we generate the search-ready text from each image.

```
{"role": "system", "content": "You are a helpful assistant that accurately describes images. " "Your responses are subsequently used to perform a web search to " "retrieve the relevant information about the image. " "Please extract the key information from the image needed to answer the question."}
{"role": "user", "content": ["type": "image"]}
{"role": "user", "content": ["type": "text", "text": {query}]}
```

5.3 Image–to–Query Conversion

The search is performed using the CRAG search class provided by the competition hosts. Because the class can accept either text or an image as input and return relevant information, we supplied text queries and retrieved the top 30 related results. input text format is {question} {answer}

5.4 Web-Scale Retrieval.

We perform reranking on the retrieved search results to provide higher-quality inputs to the VLM. We use the page_snippet attribute as the reranking text. The reranking model is Qwen3-Reranker-0.6B [18]. The prompt fed to the reranker is as follows:

```
<Instruct>: Given a Question and Image Summary in Query, retrieve relevant passages that answer the question
<Query>: questions about the provided image summary.
Question: {query}
Image Summary  {answer}
Conversation: {message str}
<Document>: search text
```

5.5 Answer Generation

We input the top 5 passages returned by the reranker into Llama 3.2 Vision Instruct together with an Answer-Generation Adapter for inference. If the model's confidence is low, it refrains from generating a specific answer and outputs "I don't know."

```
{"role": "system", "content": "You are a helpful assistant that truthfully answers user questions about the provided image.Keep your response concise and to the point. If you don't know the answer, respond with 'I don't know'."}
{"role": "user", "content": ["type": "image"]}
...
{history}
...
{"role": "user", "content": rag_context}
{"role": "user", "content": query}
```

rag_context is created from the data retrieved by RAG; generated by concatenating retrieved documents as a newline-separated list of entries in the following format.

```
[Info {rank} | {score:.3f}] {snippet}
```

6 Experiment

6.1 Dataset

We used the publicly released CRAG 2025 v0.1.2 dataset from Hugging Face as our training data. The dataset includes a split labeled validation and test; we reconstructed dataset using our proposed method and used the resulting data for training.

6.2 Training Parameters

Table 1: Training Parameters

Parameter name	Value
Model	Llama 3.2 11B Vision Instruct
Batch size	1
Gradient accumulation	16
Epochs	3
Optimizer	AdamW [8]
Learning rate	0.01
Weight decay	0.01
α (LoRA)	16
Rank (LoRA)	8
Target modules (LoRA)	`down_proj, o_proj, k_proj, q_proj, gate_proj, up_proj v_proj`

Table 1 lists the primary parameters used during training, including the optimizer, model configuration, and other associated settings.

6.3 Inference Parameters

Table 2 summarizes the search parameters that affect accuracy during inference.

6.4 Ablation Study

6.4.1 Evaluation Metrics. For each question, we assign one of the four previously defined labels—Perfect (1.0), Acceptable (0.5), Missing (0.0), or Incorrect (−1.0)—and aggregate these to obtain the evaluation metrics Accuracy, Hallucination, Missing, and the overall Truthfulness Score. In the ablation study, however, we rely solely on automatic evaluation; consequently, the Acceptable category, which requires subjective judgment in manual evaluation, is not included in the final scores.

Table 2: Retrieval & Reranking Parameters

Parameter name	Value
Reranking model	Qwen3-Reranker-0.6B
Search candidates (k)	30
RAG context after reranking (k_{final})	5

Truthfulness Score The overall metric: each response is scored as Perfect (1.0), Acceptable (0.5), Missing (0.0), or Incorrect (−1.0); the final score is the mean over all responses.

Missing Proportion of responses where the system explicitly replied "I don't know" (Missing / Total).

Hallucination Among Incorrect responses, the fraction that contain content not supported by either the retrieved evidence or the image.

Accuracy Proportion of responses judged Perfect (Perfect / Total).

6.4.2 Search K. We investigated how the number of results retrieved by the Mock Web Search and their ranking depth affect performance. TableX reports outcomes when varying the number of retrieved documents $K \in 10, 20, 30, 40$, while fixing the re-ranking cutoff at $K_{\text{rerank}} = 5$. As shown in Table.3, overall metrics did not change drastically across different K values. Increasing K reduced the Missing rate and tended to raise Accuracy. This suggests that retrieving more high-confidence passages led to more attempted answers and, consequently, more correct ones. However, the Hallucination Rate also increased, so the final Truthfulness Score did not improve substantially. For the final submission, considering the evaluation on the private test set, we chose $K = 30$.

Table 3: Evaluation by Search K

K	Truthfulness Score	Hallucination Rate	Accuracy	Missing
10	0.074	0.144	0.218	0.638
20	0.085	0.143	0.229	0.628
30	0.078	0.148	0.226	0.626
40	0.081	0.154	0.236	0.610

6.4.3 Rerank K. We compared how many reranked passages to feed into the model relative to the number initially retrieved. With Search K fixed at 30, we evaluated Rerank K = 3, 4, and 5. Table 4 is evaluation result. Increasing the number of ranked passages increases the input information and thus improves accuracy. However, a memory error occurred at K = 6, so we adopted K = 5 for this work. Without reranker is best in local evaluation, but Missing is high, I can't choice it for final human evaluation

6.4.4 Reranker Model. We evaluated reranker performance. Prior reports suggest that Qwen3-Reranker-0.6B generally achieves higher accuracy than bge-reranker-v2-m3 [2] . Accordingly, we compared

Table 4: Rerank K comparison

Rerank K	Truthfulness Score	Hallucination Rate	Accuracy	Missing
0(w/o reranker)	0.081	0.084	0.165	0.749
3	0.070	0.149	0.219	0.632
4	0.075	0.147	0.222	0.631
5	0.078	0.148	0.226	0.626

these models in the context of this competition to test whether the more accurate reranker also delivers higher end-to-end performance.

Table 5: Reranker Model Evaluation

Reranker model	Truthfulness Score	Missing	Hallucination Rate	Accuracy
BAAI/bge-reranker-v2-m3	0.043	0.692	0.133	0.175
Qwen3-Reranker-0.6B	0.057	0.665	0.139	0.196

6.4.5 Image Knowledge Graph. In this competition, we can leverage the Image Knowledge Graph (Image KG). Thus, we evaluate how performance changes when the Image KG is used. We retrieve results from both the Image KG and the Mock Web Search, then apply reranking. For the Image KG, the obtained JSON—containing the extracted attributes—is split into chunks of 600 characters with a stride of 450.

The results are reported in Table 6. While the validation set showed a slight improvement in accuracy, the private evaluation score decreased. Therefore, we decided not to use the Image KG during inference.

Table 6: Baseline vs. +Image KG

Setting	Truthfulness Score	Hallucination Rate	Accuracy	Missing
Baseline	0.078	0.148	0.226	0.626
+Image KG	0.087	0.141	0.229	0.630

6.4.6 Dual Adapter. We propose Dual Adapter in this paper. Single Adapter is only used Answer Generation Adapter using Retrieve and Answer Generation. Dual Adapter is separete with Answer Generation Adapter and Retrieve. Table 7 is result. In Leaderboard, Single Adapter is better, but Dual Adapter Solution is better than Single in final human evaluation.

Table 7: Single Adapter vs Dual Adapter

Setting	Truthfulness Score	Hallucination Rate	Accuracy	Missing
Single Adapter	0.057	0.139	0.196	0.665
Dual Adapter	0.012	0.232	0.244	0.524

7 Final Result

As shown in Table 8, the competition selected two submissions per team; final standings were based on the better score after organizers performed manual adjudication to correct errors and award Acceptable credit beyond the automatic evaluation. Our team (AcroYAMALEX) finished third in Task 2. We used only Answer-Generation Adapter in Task1, then We got 12th in Leaderboard. In Task3, We got 6th in Leaderboard

Table 8: Task 2 Final Result

Team name	Score
Team_NVIDIA	23.3%
db3	22.1%
AcroYAMALEX	**21.4%**

8 Conclusion

In this study, we propose a multimodal RAG pipeline that augments Llama 3.2 11B Vision Instruct with a two-stage LoRA adapter design: (i) a Retrieval-Oriented Adapter, which converts visual content into high-precision search queries, and (ii) an Answer-Generation Adapter, which autonomously abstains by outputting "I don't know" when uncertainty is detected. By applying Qwen3-Reranker-0.6B for passage re-ranking to filter search evidence and efficiently fusing image, retrieval, and dialogue history, our system secured 3rd place in Task 2 of the CRAG-MM 2025 challenge.

References

[1] Meta AI. 2024. Llama 3.2: Revolutionizing Edge AI and Vision with Open, Customizable Models. https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/. Accessed: 2025-07-17.

[2] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. arXiv:2402.03216 [cs.CL]

[3] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. arXiv:2305.14314 [cs.LG] https://arxiv.org/abs/2305.14314

[4] Tanmay Gupta and Aniruddha Kembhavi. 2022. Visual Programming: Compositional visual reasoning without training. arXiv:2211.11559 [cs.CV] https://arxiv.org/abs/2211.11559

[5] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. *CoRR* abs/2106.09685 (2021). arXiv:2106.09685 https://arxiv.org/abs/2106.09685

[6] Ziniu Hu, Ahmet Iscen, Chen Sun, Zirui Wang, Kai-Wei Chang, Yizhou Sun, Cordelia Schmid, David A. Ross, and Alireza Fathi. 2023. REVEAL: Retrieval-Augmented Visual-Language Pre-Training with Multi-Source Multimodal Knowledge Memory. arXiv:2212.05221 [cs.CV] https://arxiv.org/abs/2212.05221

[7] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*. Curran Associates, Inc., 9459–9474. https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html Paper URL includes links to PDF and supplementary material.

[8] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=Bkg6RiCqY7

[9] Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. arXiv:2303.08896 [cs.CL] https://arxiv.org/abs/2303.08896

[10] OpenAI. 2024. *GPT-4o mini: Advancing Cost-Efficient Intelligence*. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/ Model capabilities, pricing, and evaluation summary. Accessed: 2025-07-18.

[11] Dídac Surís, Sachit Menon, and Carl Vondrick. 2023. ViperGPT: Visual Inference via Python Execution for Reasoning. arXiv:2303.08128 [cs.CV] https://arxiv.org/abs/2303.08128

[12] CRAG-MM Team. 2025. CRAG-MM: A Comprehensive RAG Benchmark for Multi-modal, Multi-turn Question Answering. https://www.aicrowd.com/challenges/meta-crag-mm-challenge-2025

[13] 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. 2025. CRAG-MM: Multimodal Multi-turn Comprehensive RAG Benchmark. arXiv:2510.26160 [cs.CV] https://arxiv.org/abs/2510.26160

[14] Yikuan Xia, Jiazun Chen, and Jun Gao. 2024. Winning Solution For Meta KDD Cup' 24. arXiv:2410.00005 [cs.IR] https://arxiv.org/abs/2410.00005

[15] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Ethan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, Lei Chen, Nicolas Scheffer, Yue Liu, Nirav Shah, Rakesh Wanga, Anuj Kumar, Wen-tau Yih, and Xin Luna Dong. 2024. CRAG - Comprehensive RAG Benchmark. In *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (Eds.), Vol. 37. Curran Associates, Inc., 10470–10490. https://proceedings.neurips.cc/paper_files/paper/2024/file/1435d2d0fca85a84d83ddcb754f58c29-Paper-Datasets_and_Benchmarks_Track.pdf

[16] Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. 2023. MM-REACT: Prompting ChatGPT for Multimodal Reasoning and Action. arXiv:2303.11381 [cs.CV] https://arxiv.org/abs/2303.11381

[17] Hanning Zhang, Shizhe Diao, Yong Lin, Yi R. Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2024. R-Tuning: Instructing Large Language Models to Say 'I Don't Know'. arXiv:2311.09677 [cs.CL] https://arxiv.org/abs/2311.09677

[18] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176* (2025).

A LLM as a Judge Prompt

{"role": "system", "content": """You are an expert evaluator for question answering systems. Your task is to determine if a prediction correctly answers a question based on the ground truth.
Rules: 1. The prediction is correct if it captures all the key information from the ground truth. 2. The prediction is correct even if phrased differently as long as the meaning is the same. 3. The prediction is incorrect if it contains incorrect information or is missing essential details. 4. If the user clearly states "I don't know", count it as a "miss", not a hallucination.
Output a JSON object with a single field 'accuracy' whose value is true or false."""},

B Inference Implementation

The high-speed inference library vLLM supports loading multiple LoRA adapters, but the Llama 3.2 11B Vision Instruct model itself cannot dynamically switch among more than one LoRA module. To work around this limitation, we implemented a solution that toggles between two LoRA adapters using the Hugging Face Transformers library.

C Hardware

Table 9: Training Environment

Component	Specification
CPU	13th Gen Intel Core i9-13900KF
GPUs	2 × NVIDIA RTX 3090
Memory	128 GB

Received 2025 August 2025; revised June 2025; accepted June 2025