VaccineRAG: Boosting Multimodal Large Language Models' Immunity to Harmful RAG Samples

Qixin Sun[1,2*], Ziqin Wang[1*], Hengyuan Zhao[1*], Yilin Li, Kaiyou Song, Si Liu[1], Xiaolin Hu[3], Qingpei Guo[†], Linjiang Huang[1†]

[1]School of Artificial Intelligence, Beihang University, Beijing 100191, China
[2]Sino-French Carbon Neutrality Research Center, École Centrale de Pékin, Beihang University, Beijing 100191, China
[3]QIYUAN LAB, Beijing 100095, China
{sunqx,wzqin,zh_hy}@buaa.edu.cn, yilinli@iscas.ac.cn, sky@gmail.com, liusi@buaa.edu.cn, xlhu@tsinghua.edu.cn, gqp.hust@gmail.com, ljhuang@buaa.edu.cn

Abstract

Retrieval Augmented Generation enhances the response accuracy of Large Language Models (LLMs) by integrating retrieval and generation modules with external knowledge, demonstrating particular strength in real-time queries and Visual Question Answering tasks. However, the effectiveness of RAG is frequently hindered by the precision of the retriever: many retrieved samples fed into the generation phase are irrelevant or misleading, posing a critical bottleneck to LLMs' performance. To address this challenge, we introduce **VaccineRAG**, a novel Chain-of-Thought-based retrieval-augmented generation dataset. On one hand, VaccineRAG employs a benchmark to evaluate models using data with varying positive/negative sample ratios, systematically exposing inherent weaknesses in current LLMs. On the other hand, it enhances models' sample-discrimination capabilities by prompting LLMs to generate explicit Chain-of-Thought (CoT) analysis for each sample before producing final answers. Furthermore, to enhance the model's ability to learn long-sequence complex CoT content, we propose **Partial-GRPO**. By modeling the outputs of LLMs as multiple components rather than a single whole, our model can make more informed preference selections for complex sequences, thereby enhancing its capacity to learn complex CoT. Comprehensive evaluations and ablation studies on VaccineRAG validate the effectiveness of the proposed scheme.

Code & Dataset —
https://github.com/qxsun02/VaccineRAG

1 Introduction

In the era of rapid information growth, maintaining the currency of large-scale models is a formidable challenge. The overwhelming and ever-increasing amount of online data necessitates continuous model updates to ensure their outputs remain accurate and relevant. Traditional model fine-tuning (Hu et al. 2022; Liu et al. 2022; Li and Liang 2021; Houlsby et al. 2019; Karimi Mahabadi, Henderson, and Ruder 2021; Chen et al. 2023) is resource-heavy and time-consuming, making frequent refreshes impractical. Retrieval-Augmented Generation (RAG) (Lewis et al. 2020a,b; Gao et al. 2023) tackles this challenge by retrieving query-relevant external knowledge from online or offline databases and feeding these results into LLMs for response generation. However, due to the uncertainty of external knowledge, a critical issue is posed in current RAG systems: their retrieval mechanisms often prioritize speed over accuracy when handling vast knowledge bases (Chen et al. 2024a). Consequently, the subsequent generation is easily misled by spurious evidence, especially content that is lexically similar but semantically incongruent, ultimately compromising the accuracy of generated responses.

To address this issue, prior works (Chen et al. 2024d; Lin et al. 2025; He et al. 2024; Jiang, Ma, and Chen 2024) have focused on improving the retriever or designing specialized retrieval pipelines, to enhance retrieval quality and reduce the number of harmful samples entering the generation phase. MM-Embed (Lin et al. 2025) proposes modality-aware hard negative mining to mitigate the modality bias exhibited by MLLM retrievers, and RagVL (Chen et al. 2024d) proposes a multimodal RAG framework using MLLM as a Re-ranker to pick positive retrieval samples. Although these retrieval-centric methods can reduce the incidence of harmful evidence, they implicitly assume that the retriever's precision remains stable after deployment. In practice, retrieval quality often degrades due to various factors, such as domain shifts caused by continuous updates to the knowledge base. Therefore, LLMs remain vulnerable to spurious passages, and their downstream performance suffers accordingly.

SURf introduces a self-refinement framework to enhance the robustness against irrelevant retrieval samples without improving the retriever (Sun et al. 2024). By jointly providing the LLMs both relevant and irrelevant retrieved samples and supervising it with the ground-truth answer, this approach suppresses the influence of harmful samples. However, it still suffers from two key drawbacks: **1) Lack of diagnostic signal**, which makes error attribution difficult and impedes the fine-grained optimization of the model's evidence-selection behavior. **2) Slow evidence alignment**, when dozens of retrieved samples are included, the model is forced to learn from a single scalar loss derived from the ground-truth answer. It may result in sparse gradients across

*Equal Contribution
†Corresponding author

all tokens, leading to slow and data-inefficient training.

In this paper, we pioneer leveraging the deep reasoning capabilities of LLMs through their Chain-of-Thought reasoning scheme to alleviate the aforementioned issue. Given a series of retrieval results containing positive and negative samples, LLMs are tasked with diagnosing the helpfulness of each retrieved sample, summarizing relevant ones, and generating the final answer step by step. To furnish dense supervision and thus richer gradient signals throughout the reasoning process, we introduce the first CoT-based Multi-Modal RAG dataset as shown in Figure 1, named **VaccineRAG**. Each instance pairs an initial question with its corresponding image and descriptive caption, alongside a *helpfulness* label, collectively prompting the model to generate a structured, multi-step chain-of-thought analysis. Consequently, we have constructed a dataset comprising 10k entries, with each entry containing an average of five retrieved samples.

We employ the training paradigms widely adopted in contemporary MLLMs, specifically SFT followed by RL, e.g. GRPO (Shao et al. 2024) for network optimization (Chen et al. 2024c; Bai et al. 2025). However, the vanilla GRPO treats rewards accrued by all tokens in the sequence uniformly. As illustrated in Figure 2, when conducting separate preference optimization for distinct segments of the CoT, vanilla GRPO is prone to advantage function misalignment: for instance, low-reward segments may reduce the loss magnitude of tokens in high-reward segments, thereby leading to slow convergence and performance degradation. To address this, we propose **Partial-GRPO**, in which the optimization process enables targeted gradient backpropagation tailored to tokens from distinct segments of the CoT, thereby significantly accelerating the convergence speed of post-optimization and improving the final model performance.

During the experimental phase, we selected three mainstream multimodal large models as the base models. We optimized them on our proposed VaccineRAG dataset using Partial GRPO, and compared the results with several constructed baselines (including SURf). The comparison verified that our proposed method can effectively counter harmful retrieved samples and improve generative robustness in multimodal RAG tasks. Furthermore, ablation studies on the components of our method demonstrated the usefulness of each structure. Examples of using our method for RAG can be found in the appendix.

2 Related work

Multimodal Retrieval-Augmented Generation.

Retrieval-augmented generation (RAG) was initially proposed to tackle knowledge-intensive tasks in natural language processing (NLP) (Lewis et al. 2020a,b; Gao et al. 2023). By integrating knowledge retrieved from external sources during generation, RAG has achieved notable success and widespread use in NLP (Asai et al. 2024; Ram et al. 2023; Gao et al. 2024; Lan et al. 2023; Chen et al. 2024b). Recently, RAG has also attracted growing attention in multimodal large models (Shohan et al. 2024; Yan and Xie 2024; Zhao et al. 2024; Xia et al. 2024, 2025). For example, in Visual Question Answering, Wiki-LLaVA (Caffagni et al. 2024) employs a hierarchical RAG mechanism to retrieve relevant Wikipedia passages, enhancing the model's ability to answer complex, domain-specific questions. In Captioning, RA-TX (Sarto et al. 2024) leverages captions from similar images to generate more accurate and context-aware descriptions. However, recent studies (Hu et al. 2025; Sun et al. 2024) have shown that low-quality retrieved samples may introduce noise and degrade generation quality, highlighting the need for methods that improve model robustness against such detrimental retrievals.

Chain of Thought (CoT) and Reinforcement Learning.

The CoT approach enhances the reasoning capability of LLMs by prompting them to sequentially derive intermediate steps rather than directly producing the final output, thereby improving their performance on complex problem-solving tasks (Wei et al. 2022). This enhancement significantly improves the reasoning capabilities, interpretability, controllability, and flexibility of large-scale models (Yu et al. 2023; Qiao et al. 2023; Lu et al. 2023). In the domain of multimodal large models, the CoT-based approach is frequently employed to address complex, multi-step problems across various domains (Zhang et al. 2023; Xu et al. 2024; Lu et al. 2022). The latest Deepseek-R1 (Shao et al. 2024) integrates reinforcement learning with the CoT approach, endowing the model with the capability of self-sampling and evolution. For the problem investigated in this study, which also involves reasoning in complex knowledge and tasks to determine the relationship between retrieved samples and the original question, we can consider employing reinforcement learning to enable the large model to perform step-by-step reasoning and achieve the desired outcomes.

3 Preliminary

A RAG system typically consists of two components: a retriever, responsible for retrieving samples relevant to the input from a database, and a generator, which takes the retrieved samples and the original input concatenated together to perform the generation. For an input x and a database $\mathcal{D} = \{d_i\}_{i=1}^{|\mathcal{D}|}$, the retriever typically employs maximum inner product search to match the top K samples from the database that have the highest similarity to the input embedding. The final set of retrieved samples is given by:

$$\mathcal{S}_{\text{retrieval}} = \underset{d \in \mathcal{D}}{\text{argTopK}} \left(\text{Emb}(x) \cdot \text{Emb}(d) \right), \tag{1}$$

where $\text{Emb}(\cdot)$ is the embedding function. The output y_{t+1} of the LLM (denoted as π) at time step $t+1$ is:

$$y_{t+1} = \pi \left(x; \mathcal{S}_{\text{retrieval}}; y_t \right). \tag{2}$$

However, given the large database size and the need to compute against input x for each sample, retrievers are typically designed to be very simple to ensure acceptable retrieval times, resulting in retrieved samples of inconsistent quality. Consequently, irrelevant samples or those lacking useful information may interfere with the model, leading to incorrect answers.

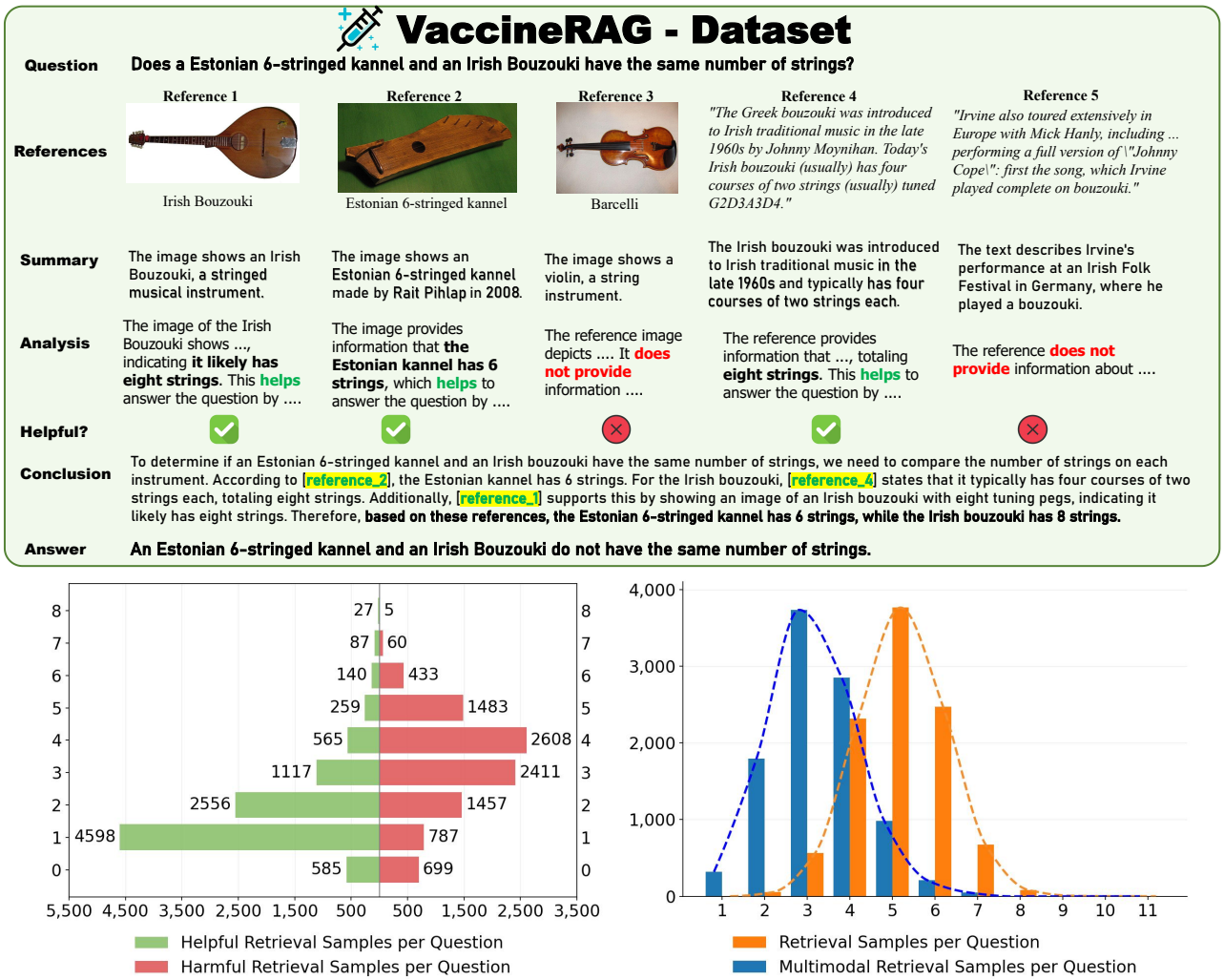

Figure 1: Top: Our multimodal RAG dataset VaccineRAG annotated with CoT. The reasoning process encompasses summarizing, analyzing each retrieved sample in conjunction with the original query, integrating the helpful samples, and ultimately generating the final answer. Down: Preliminary statistics of the dataset.

4 VaccineRAG

4.1 Data Construction

To teach MLLMs to analyze and utilize retrieval samples for multimodal RAG using the CoT approach, we leverage the WebQA dataset (Chang et al. 2022) as the basis for constructing a new dataset. Each sample in this dataset includes a text-based question, multiple retrieval results, and a label indicating their helpfulness. However, the dataset does not provide an explanation of the helpfulness of each retrieval sample, nor does it indicate how the final answer is derived from these retrieval samples. In fact, these two steps correspond to two levels of reasoning:

1) **Reasoning within retrieval samples**: This involves summarizing the content of a retrieval sample and analyzing its relevance to the original question in order to determine whether the sample is helpful.

2) **Cross-retrieval sample reasoning**: This involves integrating the helpful retrieval samples from the previous stage and using them to infer the answer to the original question.

To obtain these CoT annotations, we use SOTA commercial large model for annotation, followed by manual verification. Specifically, for each sample, we use GPT-4o to perform the following annotations: 1) Retrieval Sample Summarization: For all retrieval results in a sample, we input images (if available) to obtain their descriptions. 2) Helpfulness Analysis: Using the sample's helpfulness label and the original question as prompts, we derive an analysis to determine whether the sample is genuine. 3) Final Conclusion: After synthesizing the analysis of all retrieval results, we generate a conclusion that aligns with the final answer.

4.2 Manual Verification

We also performed manual verification of the information generated by GPT-4o to ensure annotation quality. For example, for the Helpfulness Analysis annotation, we compared the Helpfulness generated by GPT-4o with the Help-

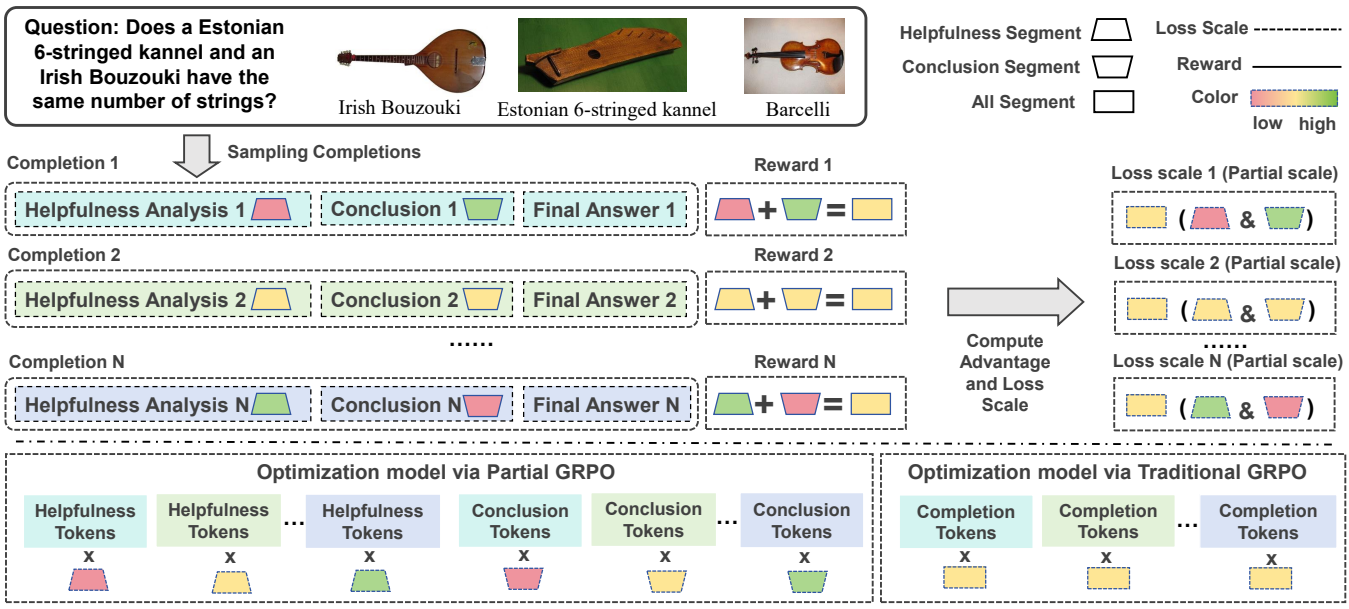

Figure 2: Partial GRPO calculates the objective function by multiplying the rewards applied to different parts of the completion with the importance sampling ratio of the related tokens, thus providing more fine-grained reward signals. The figure shows that traditional GRPO optimization cannot distinguish a better completion, whereas partial GRPO can identify sections that are not globally but locally improved and encourage the model to generate them. The Format Reward is omitted in the figure because its calculation method is the same as in traditional GRPO.

fulness annotated in the original WebQA. If they matched, we had strong confidence in the correctness of the annotation. If there was a discrepancy, manual verification was conducted, and corrections or exclusions were made accordingly. We provided a standardized template to link the above information together, forming a natural language-based CoT for model training.

4.3 Dataset Statistic

In total, we constructed approximately 10,000 samples. The dataset samples, some statistical details are illustrated in the Figure 1. Statistically, this dataset comprises approximately 10k entries, with each entry containing an average of 5.05 ± 1.06 retrieved samples. Among these, the number of image-type references averages 3.32 ± 1.08, while the count of helpful retrieved samples is 1.85 ± 1.36. Notably, 84% of the questions are answerable, whereas the remaining questions cannot be addressed due to insufficient information.

5 Methodology

In this section, we introduce our training scheme, consisting of two successive stages: SFT-based model warming up for output knowledge alignment and Partial-GRPO-based model preference optimization.

5.1 Stage I: Warming Up

To enable the base model to adapt to the task and the basic format of the output, thereby preventing the generation of outputs that fail to meet the fundamental formatting requirements during the next stage, we first optimize the base model using Supervised Fine-Tuning. To mitigate the risk of excessive training, which could otherwise lead to a loss of sampling diversity in the subsequent stage, we just utilize 15% of the data and train within a few steps.

5.2 Stage II: Preference Optimization

Vanilla-GRPO Given an input q, traditional GRPO samples a group of G completions $\{o_i\}_{i=1}^{G}$ from the old policy π_{old}. Let the partial importance sampling ratio for a certain completion o_i in the interval $[x_1, x_2](x_1 \geq 1, x_2 \leq |o_i|)$ be

$$\omega_{x_1}^{x_2}(o_i) = \frac{1}{x_2 - x_1} \sum_{t=x_1}^{x_2} \frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q, o_{i,<t})}. \quad (3)$$

Then, traditional GRPO optimizes the policy model by maximizing the given objective:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim P(Q),\, \{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)} \left[\frac{1}{G} \sum_{i=1}^{G} \omega_1^{|o_i|}(o_i) \hat{A}_i \right], \quad (4)$$

To improve readability, we omitted the KL divergence term and the clipping objective function in the formula. Here, $\hat{A}_i$ represents the relative advantage of the i-th completion, which can be calculated by normalizing the advantages across a group of completions.

Partial-GRPO Directly calculating the total reward function through a weighted sum and applying it to evaluate the entire completion, although reasonable, wastes the information about the quality of different parts in the CoT. In our

Partial-GRPO, there are mainly two core designs: 1) Reward Function and 2) Gradient backpropagation.

Reward functions. 1) Format Reward. The format reward r^f is designed to encourage the model to generate completions that adhere to the dataset template. A reward of 1 is assigned to completions that conform to the required format, while a reward of 0 is given otherwise. The scope of this reward function is the whole completion.

$$r^f = \begin{cases} 1, & \text{if the completion adheres to the format} \\ 0, & \text{otherwise} \end{cases}. \quad (5)$$

2) Helpfulness Reward. The helpfulness reward r^h is designed to evaluate whether the model correctly analyzes the helpfulness of each retrieved sample. For an input containing n retrieved samples, we extract the keywords ("helpful"/"unhelpful") from the model's analysis of each retrieved sample to determine the model's judgment on its helpfulness. Let the ground truth of the helpfulness for the j-th retrieved sample be denoted as H_j^{gt}, and the model's judgment as H_j. The sub-reward function for each retrieved sample, $r_{h,j}$, is defined as:

$$r_j^h = \begin{cases} 1, & \text{if } H_j = H_j^{gt} \\ 0, & \text{otherwise} \end{cases}. \quad (6)$$

Then, we can get the overall helpfulness reward function r^h as below

$$r^h = \frac{1}{n}\sum_{j=1}^{n} r_j^h. \quad (7)$$

The scope of this reward function is the helpfulness analysis section.

3) Conclusion Reward. The conclusion reward r^c is utilized to verify whether the model correctly cites the appropriate retrieved samples in the summary section. Specifically, it evaluates whether the model cites the samples that were previously determined as "helpful" in the reference analysis section, while refraining from citing those deemed "unhelpful". For the j-th retrieved sample, if it is cited in the conclusion section of the completion, C_j is assigned a value of true; otherwise, it is assigned false. The sub-reward function for this retrieved sample is defined as:

$$r_j^c = \begin{cases} 1, & \text{if } C_j = H_j \\ 0, & \text{otherwise} \end{cases}. \quad (8)$$

Similarly, the overall conclusion reward function is defined as:

$$r^c = \frac{1}{n}\sum_{j=1}^{n} r_j^c. \quad (9)$$

The scope of this reward function is the conclusion section.

Gradient backpropagation. Given the i-th sampled completion o_i, it is divided into three parts: helpfulness analysis, conclusion, and final answer, which are denoted as o_i^h, o_i^c, and o_i^f, respectively.

$$\mathcal{J}_{\text{P-GRPO}}(\theta) = \mathbb{E}_{q\sim P(Q),\,\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)} \left[\frac{1}{G}\sum_{i=1}^{G}\hat{\boldsymbol{R}}_i\boldsymbol{\Omega}_i\right], \quad (10)$$

Among them, $\hat{\boldsymbol{R}}$ and $\boldsymbol{\Omega}_i$ are respectively the three normalized rewards of the i-th completion and the partial importance sampling ratios corresponding to the relative tokens (or "scope") of each reward.

$$\hat{\boldsymbol{R}}_i = \begin{bmatrix} \hat{r}^h(o_i) \\ \hat{r}^c(o_i) \\ \hat{r}^f(o_i) \end{bmatrix}^\top, \boldsymbol{\Omega}_i = \begin{bmatrix} \omega_1^{|o_i^h|}(o_i) \\ \omega_{|o_i^h|+1}^{|o_i^h|+|o_i^c|}(o_i) \\ \omega_1^{|o_i|}(o_i) \end{bmatrix}, \quad (11)$$

where $\hat{r}^h(o_i)$ is the helpfulness reward of the i-th completion after normalization across a group of completions, and $\hat{r}^c(o_i)$ and $\hat{r}^f(o_i)$ are defined similarly.

6 Experiments

6.1 Setups and Implementation Details.

Experiments Setup. To evaluate the MLLM's immunity against harmful retrieval samples, we selected 858 answerable questions from the validation set of WebQA. We designed two strategies for testing on answerable questions: 1) **Polluted generation**: For each answerable question, harmful retrieval samples are incrementally added to the helpful retrieval samples. This strategy assesses the impact of additional harmful samples on the generation when helpful samples are retrieved. 2) **TopK generation**: Using the retriever, a fixed number of K samples are retrieved from the database for each answerable question to facilitate generation. This strategy directly evaluates the final effectiveness of the RAG framework. For experiments related to unanswerable questions, please refer to the supplementary materials.

Implementation Details. We used 8×96GB NVIDIA H20 GPUs for both training and inference. For additional details, please refer to the supplementary materials.

6.2 Metrics

Accuracy (ACC). Given that the employed VQA dataset is open-ended in nature, we utilized the closed-source model GPT-4o to evaluate the accuracy of the answers. Specifically, based on the triad of question, ground truth, and the model's generated answer, each response was scored span from 0 to 5. Subsequently, these scores were normalized to a range of 0 to 100 for comprehensive assessment.

Through experiments, we found that using this method for evaluation aligns much more closely with human assessments compared to the method used in WebQA. This is because the questions in WebQA are open-ended, allowing for multiple possible answers, which leads to misjudgments in the native evaluation method used in WebQA. Moreover, for the same input, the scores obtained using this evaluation method are consistent across multiple assessments. Therefore, we believe this scoring approach is reasonable. In the supplementary materials, we report the agreement rates between different evaluation methods and human evaluations.

Model	Type	ACC_{+0}	ACC_{+1}	ACC_{+2}	ACC_{+3}	ACC_{+4}	ACC_{+5}	$MA_5 \uparrow$	$ADR_5 \downarrow$
Qwen2-VL	Zero-shot	46.81	44.15	41.03	39.53	37.93	37.34	41.13	34.05
	SURf	49.46	47.39	44.48	44.66	43.59	42.12	45.28	25.08
	SURf+CoT	50.35	46.92	47.44	48.04	47.34	45.41	47.58	16.60
	GRPO	58.46	54.94	54.26	53.72	53.03	52.60	54.50	23.75
	Partial GRPO	**58.93**	**56.24**	**55.64**	**55.72**	**55.50**	**55.58**	**56.27**	**15.98**
Qwen2.5-VL	Zero-shot	62.42	55.50	55.52	54.66	52.19	51.79	55.35	42.45
	SURf	61.89	56.32	56.22	57.16	56.06	54.69	57.06	29.00
	SURf+CoT	60.82	59.44	60.19	61.12	59.49	58.06	59.85	5.83
	GRPO	59.86	59.44	60.19	61.11	59.49	58.86	59.83	**0.21**
	Partial GRPO	**66.27**	**64.97**	**64.02**	**64.28**	**63.92**	**63.73**	**64.53**	10.43
InternVL3	Zero-shot	62.31	58.37	57.27	57.09	56.18	54.99	57.70	27.65
	SURf	49.02	42.28	45.64	41.28	42.75	44.48	44.24	28.67
	SURf+CoT	57.49	55.13	55.02	54.85	54.64	54.37	55.25	13.44
	GRPO	63.47	62.02	60.43	60.06	58.67	59.02	60.61	17.13
	Partial GRPO	**64.24**	**63.18**	**62.60**	**61.74**	**61.49**	**60.43**	**62.28**	**11.76**

Table 1: Performance of our approach and baselines on polluted generation

Mean Accuracy (MA). For polluted generation, when the number of harmful retrieval samples added is k, and the overall accuracy is ACC_{+k}, the average accuracy is denoted as:

$$MA_k = \frac{1}{k}\sum_{i=1}^{k} ACC_{+i}. \tag{12}$$

Accuracy Degradation Rate (ADR). For polluted generation, when the number of harmful retrieval samples added is k, and the overall accuracy is ACC_{+k}, the accuracy degradation rate ADR_k is calculated as follows:

$$ADR_k = \frac{1}{k}\sum_{i=1}^{k} \frac{ACC_{+0} - ACC_{+i}}{ACC_{+0}}. \tag{13}$$

6.3 Baselines

We selected Qwen2-VL-7B (Bai et al. 2025), Qwen2.5-VL-7B (Wang et al. 2024), and InternVL3-8B (Chen et al. 2024c) as the base models for training. The following methods were chosen as baselines to compare with our proposed approach:

1) Zero-shot: This approach involves using the base model without any fine-tuning, directly prompting it to generate the final answer.

2) SURf: According to the approach in SURf (Sun et al. 2024), this method does not utilize the CoT information from VaccineRAG, but instead employs only the final answer as supervision for SFT.

3) SURf+CoT: Based on the SURf method, we introduce a CoT process to reason about the retrieved samples.

4) GRPO: Using the traditional GRPO algorithm, multiple reward functions are weighted and summed to evaluate the entire completion segment, and this evaluation is used as the advantage in GRPO.

Type	ACC_{Top-k}			
k	2	5	10	15
Qwen2-VL				
Zero-shot	28.39	30.68	30.26	28.93
SURf	28.39	31.61	31.77	30.70
SURf+CoT	24.94	31.14	34.97	33.57
GRPO	32.58	36.87	40.62	40.09
Partial GRPO	**33.74**	**38.60**	**42.27**	**41.44**
Qwen2.5-VL				
Zero-shot	**37.60**	40.12	40.84	39.98
SURf	27.76	33.45	38.07	40.68
SURf+CoT	37.90	39.58	44.04	42.94
GRPO	31.76	**41.52**	43.98	44.75
Partial GRPO	33.87	40.69	**45.84**	**47.21**
InternVL3				
Zero-shot	26.50	31.82	33.40	37.69
SURf	17.16	19.14	22.56	23.96
SURf+CoT	24.57	30.02	36.83	38.93
GRPO	**28.95**	36.62	41.03	43.17
Partial GRPO	28.76	**36.85**	**42.74**	**46.08**

Table 2: The accuracy of our approach and baselines, varies with the number of retrieved samples

6.4 Main Result

Polluted Generation. To evaluate polluted generation, we systematically injected 0-5 harmful retrieval samples into the set of helpful retrieval samples, measured the accuracy of the multimodal large model's responses, and then calculated the mean accuracy and the accuracy degradation rate.

The results are shown in Table **??**. Without training, all three models tested experienced varying degrees of accuracy degradation when harmful retrieval samples were added. For example, Qwen2.5-VL achieved the highest ac-

Model	Type	ACC_{+0}	ACC_{+1}	ACC_{+2}	ACC_{+3}	ACC_{+4}	ACC_{+5}	$MA_5 \uparrow$	$ADR_5 \downarrow$
Qwen2-VL	Partial GRPO	**58.93**	**56.24**	**55.64**	**55.72**	**55.50**	**55.58**	**56.27**	**15.98**
	w/o Helpfulness Reward	55.84	53.31	52.22	51.79	52.03	51.04	52.71	18.80
	w/o Conclusion Reward	56.62	55.31	54.01	53.12	53.46	50.88	53.90	16.34
Qwen2.5-VL	Partial GRPO	**66.27**	**64.97**	**64.02**	**64.28**	**63.92**	**63.73**	**64.53**	**10.43**
	w/o Helpfulness Reward	60.44	58.02	57.95	58.20	58.60	57.41	58.44	12.03
	w/o Conclusion Reward	62.28	61.17	60.26	61.59	59.15	57.26	60.29	11.99
InternVL3	Partial GRPO	**64.24**	**63.18**	**62.60**	**61.74**	**61.49**	**60.43**	**62.28**	**11.76**
	w/o Helpfulness Reward	57.82	51.74	53.73	50.35	50.68	50.44	52.46	31.18
	w/o Conclusion Reward	60.49	60.26	58.62	56.95	58.62	57.13	58.68	10.87

Table 3: Performance of Partial GRPO and its variants with certain rewards removed on polluted generation. The variant with the format reward removed is not shown in the table due to unstable training.

curacy of 62.42% without any harmful samples, but its Accuracy Degradation Rate reached 42.45% after five harmful retrieval samples were added, the worst among the three models. On the other hand, although InternVL3 performed slightly worse than Qwen2.5-VL without any harmful samples, it exhibited the best performance in terms of Accuracy Degradation Rate. **This indicates that the performance decline caused by the inclusion of harmful samples is a common issue in the RAG tasks of tested multimodal large models on the dataset we proposed.**

After training with the Partial GRPO method we proposed, the model's Accuracy Degradation Rate showed significant improvement compared to other baselines, without any loss in Mean Accuracy. **This suggests that our method enhances the model's robustness when harmful retrieval samples are added.** Additionally, we observed that while the SURf method also reduces accuracy loss when harmful retrieval samples are included to some extent, it leads to a loss in Mean Accuracy. Adding CoT helped mitigate this loss, as SURf lacks the analytical context of CoT as guidance, making it difficult to learn the relationship between the final answer and the polluted retrieval samples.

Note that ADR is defined as degradation relative to the accuracy without pollution (ACC_{+0}). When ACC_{+0} is already low, there is little room for further degradation, so ADR can look artificially small. This happened on Qwen2.5-VL.

TopK Generation. We employed BGE-VL-base (Zhou et al. 2024) as the retriever, utilizing the questions directly as queries. From the knowledge base constructed by all references in the validation split of WebQA, we selected the Top-K (K=2,5,10,15) samples through maximum inner product search to participate in the generation process.

The experimental results are shown in Table **??**. When using a larger K value, such as $K = 10$ or $K = 15$, our trained model consistently achieved the best performance compared to other baselines, and also attained higher accuracy than with smaller K values. **This indicates that when provided with more retrieval samples, the model is able to effectively utilize the helpful ones while being less affected by harmful samples.** In contrast, for untrained models like Qwen2.5-VL and Qwen2-VL, accuracy declined to some extent as the K value increased. This further confirms that our proposed method enhances the model's robustness against harmful retrieval samples.

6.5 Ablation: Are All Reward Functions Useful?

In our designed methodology, there are three reward functions. To verify their indispensability, we conducted an ablation study on polluted generation by removing each reward function individually, with results shown in Table **??**. We observed that removing the Format Reward during training causes the model's output format to deviate from specifications. This leads to a critical issue: while we cannot accurately assess the other two rewards for the few completion outputs violating format requirements, we could skip backpropagation for these two rewards on such completions to ignore the errors. The Format Reward penalizes non-compliant outputs, ensuring virtually no format violations throughout training. Without it, numerous non-compliant completions emerge, subsequently affecting the other two reward functions and causing model collapse during training. Although removing the other two reward functions still permits stable training, the final evaluation metrics including Mean Accuracy and Accuracy Degradation demonstrate significant deterioration. We observed similar results in the Top-K generation experiments. **This confirms that all three reward functions we designed are indispensable.**

7 Conclusion

In this work, we pose a key bottleneck in RAG systems: over-reliance on retriever precision, which often results in irrelevant or misleading contexts being fed into the generation phase. To evaluate and mitigate this issue, we introduced **VaccineRAG**, a CoT-based Multimodal RAG dataset, which not only serves as a benchmark for evaluating LLMs under varying positive/negative sample ratios, but also provides a scheme to enhance models' abilities in sample discrimination through explicit CoT reasoning. We also proposed **Partial-GRPO**, a new approach for learning multi-step, long-sequence CoT content by modeling LLM outputs as multiple components rather than a monolithic whole, enabling more nuanced preference selection and improves the model's overall capability to handle intricate reasoning tasks.

Acknowledgments

This research is supported in part by National Key R&D Program of China (2022ZD0115502), National Natural Science Foundation of China (No. 62461160308, No. 62576024, U23B2010), "the Fundamental Research Funds for the Central Universities" (No. 501RCQD2025), "Pioneer" and "Leading Goose" R&D Program of Zhejiang (No. 2024C01161), Beijing Natural Science Foundation (QY25227), Ningbo Science and Technology Innovation 2025 Major Project (2025Z034), NSFCRGC Project (N CUHK498/24).

References

Asai, A.; Wu, Z.; Wang, Y.; Sil, A.; and Hajishirzi, H. 2024. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. In *The Twelfth International Conference on Learning Representations*.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; Zhong, H.; Zhu, Y.; Yang, M.; Li, Z.; Wan, J.; Wang, P.; Ding, W.; Fu, Z.; Xu, Y.; Ye, J.; Zhang, X.; Xie, T.; Cheng, Z.; Zhang, H.; Yang, Z.; Xu, H.; and Lin, J. 2025. Qwen2.5-VL Technical Report. arXiv:2502.13923.

Caffagni, D.; Cocchi, F.; Moratelli, N.; Sarto, S.; Cornia, M.; Baraldi, L.; and Cucchiara, R. 2024. Wiki-LLaVA: Hierarchical Retrieval-Augmented Generation for Multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 1818–1826.

Chang, Y.; Narang, M.; Suzuki, H.; Cao, G.; Gao, J.; and Bisk, Y. 2022. Webqa: Multihop and multimodal qa. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16495–16504.

Chen, J.; Xiao, S.; Zhang, P.; Luo, K.; Lian, D.; and Liu, Z. 2024a. M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 2318–2335. Bangkok, Thailand: Association for Computational Linguistics.

Chen, T.; Wang, H.; Chen, S.; Yu, W.; Ma, K.; Zhao, X.; Zhang, H.; and Yu, D. 2024b. Dense X Retrieval: What Retrieval Granularity Should We Use? In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 15159–15177. Miami, Florida, USA: Association for Computational Linguistics.

Chen, Z.; Wang, Z.; Wang, Z.; Liu, H.; Yin, Z.; Liu, S.; Sheng, L.; Ouyang, W.; Qiao, Y.; and Shao, J. 2023. Octavius: Mitigating task interference in mllms via lora-moe. *arXiv preprint arXiv:2311.02684*.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024c. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24185–24198.

Chen, Z.; Xu, C.; Qi, Y.; and Guo, J. 2024d. MLLM Is a Strong Reranker: Advancing Multimodal Retrieval-augmented Generation via Knowledge-enhanced Reranking and Noise-injected Training. *CoRR*, abs/2407.21439.

Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Wang, H.; and Wang, H. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2: 1.

Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Wang, M.; and Wang, H. 2024. Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv:2312.10997.

He, X.; Tian, Y.; Sun, Y.; Chawla, N.; Laurent, T.; LeCun, Y.; Bresson, X.; and Hooi, B. 2024. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems*, 37: 132876–132907.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International conference on machine learning*, 2790–2799. PMLR.

Hu, E. J.; yelong shen; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Hu, W.; Gu, J.-C.; Dou, Z.-Y.; Fayyaz, M.; Lu, P.; Chang, K.-W.; and Peng, N. 2025. MRAG-Bench: Vision-Centric Evaluation for Retrieval-Augmented Multimodal Models. In *The Thirteenth International Conference on Learning Representations*.

Jiang, Z.; Ma, X.; and Chen, W. 2024. Longrag: Enhancing retrieval-augmented generation with long-context llms. *arXiv preprint arXiv:2406.15319*.

Karimi Mahabadi, R.; Henderson, J.; and Ruder, S. 2021. Compacter: Efficient low-rank hypercomplex adapter layers. *Advances in Neural Information Processing Systems*, 34: 1022–1035.

Lan, T.; Cai, D.; Wang, Y.; Huang, H.; and Mao, X.-L. 2023. Copy is All You Need. In *The Eleventh International Conference on Learning Representations*.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; Riedel, S.; and Kiela, D. 2020a. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 9459–9474. Curran Associates, Inc.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474.

Li, X. L.; and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Lin, S.-C.; Lee, C.; Shoeybi, M.; Lin, J.; Catanzaro, B.; and Ping, W. 2025. MM-EMBED: UNIVERSAL MULTIMODAL RETRIEVAL WITH MULTIMODAL LLMS. In *The Thirteenth International Conference on Learning Representations*.

Liu, X.; Ji, K.; Fu, Y.; Tam, W.; Du, Z.; Yang, Z.; and Tang, J. 2022. P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 61–68. Dublin, Ireland: Association for Computational Linguistics.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 2507–2521. Curran Associates, Inc.

Lu, P.; Qiu, L.; Yu, W.; Welleck, S.; and Chang, K.-W. 2023. A Survey of Deep Learning for Mathematical Reasoning. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 14605–14631. Toronto, Canada: Association for Computational Linguistics.

Qiao, S.; Ou, Y.; Zhang, N.; Chen, X.; Yao, Y.; Deng, S.; Tan, C.; Huang, F.; and Chen, H. 2023. Reasoning with Language Model Prompting: A Survey. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5368–5393. Toronto, Canada: Association for Computational Linguistics.

Ram, O.; Levine, Y.; Dalmedigos, I.; Muhlgay, D.; Shashua, A.; Leyton-Brown, K.; and Shoham, Y. 2023. In-Context Retrieval-Augmented Language Models. *Transactions of the Association for Computational Linguistics*, 11: 1316–1331.

Sarto, S.; Cornia, M.; Baraldi, L.; Nicolosi, A.; and Cucchiara, R. 2024. Towards Retrieval-Augmented Architectures for Image Captioning. *ACM Trans. Multimedia Comput. Commun. Appl.*, 20(8).

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y.; Wu, Y.; et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Shohan, F. T.; Nayeem, M. T.; Islam, S.; Akash, A. U.; and Joty, S. 2024. XL-HeadTags: Leveraging Multimodal Retrieval Augmentation for the Multilingual Generation of News Headlines and Tags. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 12991–13024. Bangkok, Thailand: Association for Computational Linguistics.

Sun, J.; Zhang, J.; Zhou, Y.; Su, Z.; Qu, X.; and Cheng, Y. 2024. SURf: Teaching Large Vision-Language Models to Selectively Utilize Retrieved Information. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 7611–7629. Miami, Florida, USA: Association for Computational Linguistics.

Wang, P.; Bai, S.; Tan, S.; Wang, S.; Fan, Z.; Bai, J.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Fan, Y.; Dang, K.; Du, M.; Ren, X.; Men, R.; Liu, D.; Zhou, C.; Zhou, J.; and Lin, J. 2024. Qwen2-VL: Enhancing Vision-Language Model's Perception of the World at Any Resolution. arXiv:2409.12191.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; ichter, b.; Xia, F.; Chi, E.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 24824–24837. Curran Associates, Inc.

Xia, P.; Zhu, K.; Li, H.; Wang, T.; Shi, W.; Wang, S.; Zhang, L.; Zou, J. Y.; and Yao, H. 2025. MMed-RAG: Versatile Multimodal RAG System for Medical Vision Language Models. In Yue, Y.; Garg, A.; Peng, N.; Sha, F.; and Yu, R., eds., *International Conference on Representation Learning*, volume 2025, 66188–66217.

Xia, P.; Zhu, K.; Li, H.; Zhu, H.; Li, Y.; Li, G.; Zhang, L.; and Yao, H. 2024. RULE: Reliable Multimodal RAG for Factuality in Medical Vision Language Models. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 1081–1093. Miami, Florida, USA: Association for Computational Linguistics.

Xu, G.; Jin, P.; Li, H.; Song, Y.; Sun, L.; and Yuan, L. 2024. LLaVA-CoT: Let Vision Language Models Reason Step-by-Step. arXiv:2411.10440.

Yan, Y.; and Xie, W. 2024. EchoSight: Advancing Visual-Language Models with Wiki Knowledge. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Findings of the Association for Computational Linguistics: EMNLP 2024*, 1538–1551. Miami, Florida, USA: Association for Computational Linguistics.

Yu, Z.; He, L.; Wu, Z.; Dai, X.; and Chen, J. 2023. Towards Better Chain-of-Thought Prompting Strategies: A Survey. arXiv:2310.04959.

Zhang, Z.; Zhang, A.; Li, M.; Zhao, H.; Karypis, G.; and Smola, A. 2023. Multimodal Chain-of-Thought Reasoning in Language Models. *arXiv preprint arXiv:2302.00923*.

Zhao, X.; Zhang, Y.; Zhang, W.; and Wu, X.-M. 2024. UniFashion: A Unified Vision-Language Model for Multimodal Fashion Retrieval and Generation. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 1490–1507. Miami, Florida, USA: Association for Computational Linguistics.

Zhou, J.; Liu, Z.; Liu, Z.; Xiao, S.; Wang, Y.; Zhao, B.; Zhang, C. J.; Lian, D.; and Xiong, Y. 2024. MegaPairs: Massive Data Synthesis For Universal Multimodal Retrieval. *arXiv preprint arXiv:2412.14475*.