Boosting Multimodal Retrieval-Augmented Generation for Knowledge-Based VQA with One-pass Ladder Reranking

Anonymous ACL submission

Abstract

Evidence selection remains a major bottleneck in Multimodal retrieval-augmented generation (RAG) for Knowledge-Based visual question answering (VQA). Current rerankers typically score candidates in isolation or employ single-round selection, failing to model the inherently comparative nature of evidence ranking. As a result, they struggle with hard negatives—candidates that are either visually near-duplicates of the query but textually irrelevant or textually plausible yet visually inconsistent with the image. To address these problems, we propose **M**ultimodal **O**ne-pass **L**adder **T**ournament, called MOLT, which reformulates reranking as a sequential ladder-style tournament. Instead of assigning absolute ranking scores, MOLT progressively filters distractors through explicit multimodal pairwise comparisons in a single decoding pass. To ensure robust learning, we introduce a two-stage training strategy: (1) supervised fine-tuning (SFT) initialized via distillation from a strong teacher model, followed by (2) reinforcement learning using Group Relative Policy Optimization (GRPO) with a composite reward that jointly optimizes output format compliance, step-wise logical consistency, and final selection accuracy. Experiments on two widely-used benchmarks show that MOLT achieves state-of-the-art performance, which outperform compared methods by up to 7.3 percentage points. The code is available at `https://anonymous.4open.science/r/molt`.

1 Introduction

Knowledge-Based Visual Question Answering (KB-VQA) requires external knowledge to answer questions. (Marino et al., 2019; Schwenk et al., 2022; Chen et al., 2023; Mensink et al., 2023) To address this, current systems typically use Multimodal Retrieval-Augmented Generation (MM-RAG). (Caffagni et al., 2024; Yan and Xie, 2024; Yang et al., 2025) Although retrievers narrow the search space, identifying the single correct candidate remains a critical bottleneck.

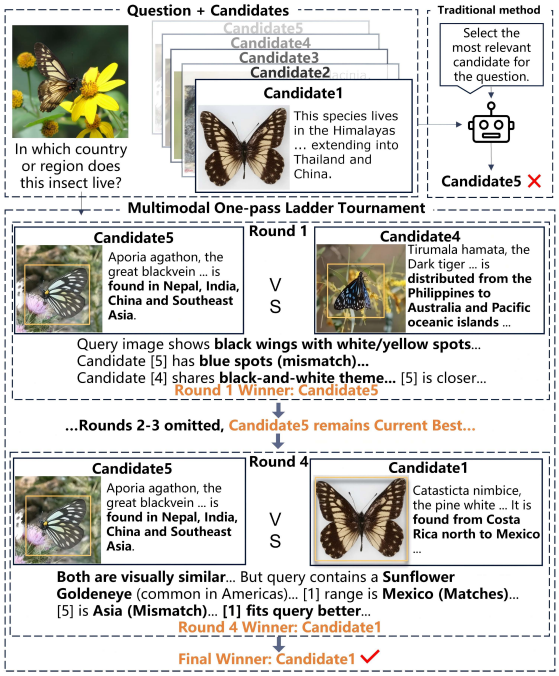

Figure 1: Given a query and multiple retrieved multimodal candidates, a one-shot selector may be misled by superficially relevant content and choose an incorrect candidate. In contrast, our MOLT performs a sequential ladder-style multimodal tournament, progressively filtering distractors through pairwise comparisons and selecting the correct evidence.

Most prior approaches treat candidate selection as a pointwise or one-shot problem. (Yang et al., 2025; Yan and Xie, 2024; Deng et al., 2025; Tian et al., 2025) They either assign an independent relevance score to each retrieved paragraph or ask a VLM to directly pick the best item from a list. We argue that these approaches are suboptimal for resolving hard negatives, as they fail to address two critical types of confusion. First, candidates often contain visually near-duplicate images (e.g., look-alike species), which are difficult to differentiate without direct visual contrast. Second, a candidate

text may appear relevant in isolation by matching question keywords or attributes, but it actually describes a different entity than the target query.

To address these challenges, we propose a novel framework called Multimodal One-pass Ladder Tournament (MOLT), which reformulates evidence selection as a process of sequential relative comparisons. We mitigate visual noise via visual grounding, cropping query-relevant regions for each candidate to focus on the target entity. Crucially, we introduce a new Weak-to-Strong Ladder schedule: the tournament starts with lower-confidence candidates and introduces the highest-scoring retrievals only in the final rounds. As illustrated in Figure 1, by leveraging grounded visual contexts and performing explicit pairwise comparisons, MOLT progressively filters out distractors to identify the optimal evidence. Notably, unlike traditional pairwise approaches that require iterative LLM calls, MOLT executes the entire multi-round comparison chain within a single end-to-end decoding pass, significantly reducing inference overhead.

Implementing such a structured tournament within a generative VLM is non-trivial. Models may drift from the strict comparison protocol or bypass intermediate reasoning steps. We tackle this via a two stage training pipeline: a supervised fine-tuning (SFT) cold start followed by Reinforcement Learning with group relative policy optimization (GRPO). Beyond standard outcome supervision, we further design a composite reward mechanism that enforces step-wise logical consistency. This creates an implicit difficulty-aware curriculum: since recovering a hard sample (buried deep in the list) requires surviving more rounds, it yields significantly higher cumulative rewards. This incentivizes the model to master the resolution of such hard examples.

In summary, our contributions are as follows:

- We propose MOLT, a grounded multimodal reranker that executes sequential ladder tournament comparisons in a single inference pass. By incorporating a Weak-to-Strong schedule, it robustly filters distractors while avoiding the redundancy of iterative interactions.
- We introduce a two-stage training recipe (SFT + GRPO) with a composite reward mechanism. This design not only enforces logical consistency but also establishes an implicit difficulty-aware curriculum that incentivizes the model to master hard samples.
- Extensive experiments on two benchmarks, i.e., E-VQA and InfoSeek, demonstrate that MOLT outperforms recent state-of-the-art baselines, achieving competitive performance by effectively distinguishing hard negatives.

2 Preliminaries

Problem Formulation. The goal of Knowledge-Based VQA (KB-VQA) is to generate a natural language answer A given a multimodal query $Q = (I_q, T_q)$, where I_q is the query image and T_q is the question. Answering these questions requires reasoning over external knowledge from a large-scale multimodal knowledge base $\mathcal{K} = \{D_1, \ldots, D_{N_\mathcal{K}}\}$. Each document $D_i \in \mathcal{K}$ consists of a primary visual image I_i and a textual body T_i. The text T_i is organized as a sequence of discrete paragraphs, i.e., $T_i = \{P_{i,1}, P_{i,2}, \ldots, P_{i,M_i}\}$, where M_i denotes the number of paragraphs in document D_i.
MM-RAG Paradigm. Current state-of-the-art systems typically follow a *Retrieve–Rerank–Generate* pipeline. First, a retriever identifies a subset of potentially relevant documents $\mathcal{D} = \text{Top-}K(\mathcal{K} \mid Q)$. From these documents, the system aims to identify the most supportive evidence paragraph. Let $\mathcal{P}(\mathcal{D})$ denote the set of all paragraphs within the retrieved documents:

$$\mathcal{P}(\mathcal{D}) = \{P_{i,m} \mid D_i \in \mathcal{D},\ m \in [1, M_i]\}. \quad (1)$$

A reranker $s_\theta(Q, P)$ scores these candidates to select the optimal evidence:

$$\mathcal{P}^* = \arg \max_{P \in \mathcal{P}(\mathcal{D})} s_\theta(Q, P). \quad (2)$$

Finally, a generator G produces the answer conditioned on the selected evidence: $A = G(Q, \mathcal{P}^*)$.

3 Method

Figure 2 shows the overall framework, which can enhance the existing standard MM-RAG pipeline. Given a query, we first retrieve candidate documents and ground them by cropping query-relevant image regions. These grounded candidates are then processed by **Multimodal One-pass Ladder Tournament** (MOLT), a reranker that selects the optimal evidence through a sequential process of pairwise comparisons. Finally, a generator produces the answer based on the selected evidence.

We employ a two-stage optimization strategy for MOLT: SFT on teacher-generated trajectories to initialize the tournament structure, followed by RL

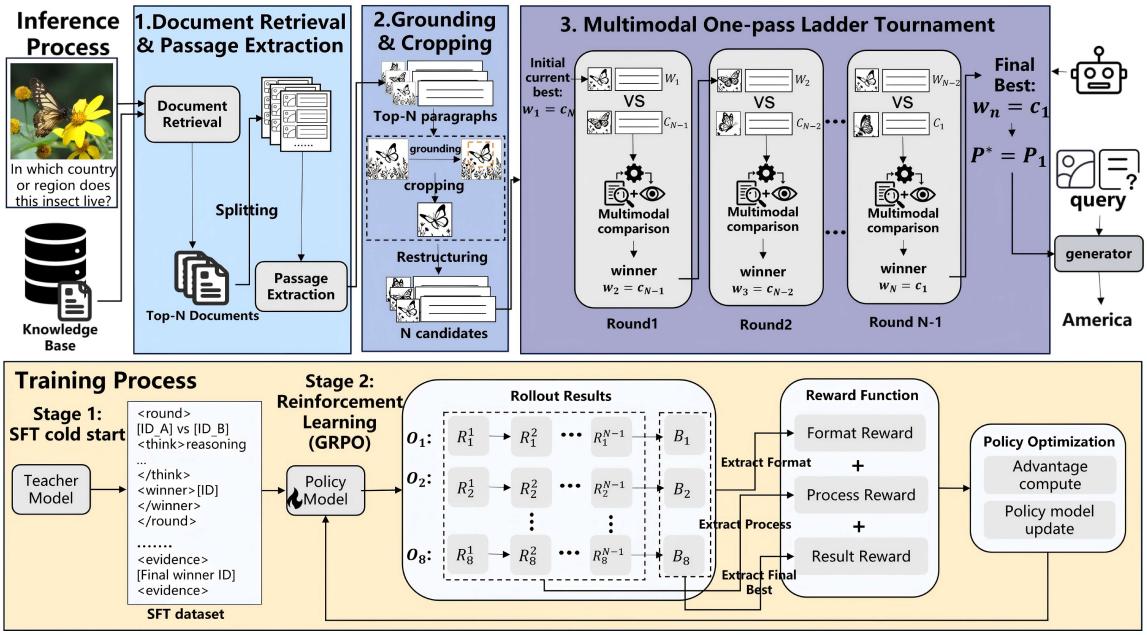

Figure 2: Overview of Our framework. **Top:** The inference pipeline retrieves and grounds Top-N candidates, then employs a single-pass sequential tournament to select the best evidence. **Bottom:** The training pipeline uses SFT initialization followed by GRPO. The rollout outputs are decomposed into reasoning chains $\mathbf{R}$ (representing tournament rounds) and final decisions $\mathbf{B}$ (representing the selected evidence ID). These components are evaluated by a composite reward function—combining format, process, and result signals—to optimize the policy model.

with composite rewards to refine process consistency and selection accuracy.

3.1 Multimodal Retrieval-Augmented Pipeline

Document Retrieval and Passage Extraction. We adopt a coarse-to-fine *document* retrieval strategy (details in Appendix A), followed by query-conditioned passage extraction. Given a multimodal query $Q = (I_q, T_q)$, we first retrieve Top-K candidate documents using CLIP, where each document D_i is represented by its summary text S_i. We then refine the Top-K set with a Q-Former and keep the Top-N documents:

$$\mathcal{D} = \text{Top-}N\big(s_{\text{qf}}(Q, D_i) \;\big|\; D_i \in \text{Top-}K(\mathcal{K} \mid Q)\big), \tag{3}$$

where $s_{\text{qf}}(\cdot)$ denotes the Q-Former score. For each selected document $D_i \in \mathcal{D}$, the document text T_i is segmented into a set of passages $\{P_{i,1}, P_{i,2}, \ldots, P_{i,M_i}\}$. Conditioned on the query text, a passage extractor selects the most relevant passage index as:

$$k_i = \text{VLM}_{\text{sel}}\big(T_q, \{P_{i,m}\}_{m=1}^{M_i}\big). \tag{4}$$

The extracted passage is denoted as:

$$P_i^{\text{para}} = P_{i,k_i}. \tag{5}$$

After extraction, we obtain N document-level evidence candidates $\{(D_i, P_i^{\text{para}})\}_{i=1}^{N}$, which are subsequently grounded and reranked by our tournament reranker.

Visual Grounding and Cropping. For each selected paragraph candidate, we localize a question-relevant region on its associated document image using a pre-trained VLM as a visual grounding module (Appx. B). Given the document image I_i and the question text T_q, the model predicts a single bounding box B_i. We crop I_i according to B_i (using a fixed padding ratio) to obtain the grounded region I_i^{crop}; if localization fails, we fall back to using the full image.

Multimodal One-pass Ladder Tournament Reranking. We construct an ordered multimodal candidate list $\mathcal{C} = [c_1, c_2, \ldots, c_N]$, where candidates are sorted by their initial retrieval scores in *descending* order. Thus, c_1 represents the highest-confidence candidate (strongest), while c_N is the lowest-confidence one (weakest).

To improve robustness, **MOLT** employs a *Weak-to-Strong Ladder* schedule. The intuition is to protect the highest-scoring candidates from being eliminated in early, noisy comparisons. By starting the tournament from the bottom of the list, the high-confidence candidates (e.g., c_1) only enter the tour-

nament in the final rounds as "defenders," reducing the probability of error accumulation.

Formally, we initialize the current winner with the weakest candidate:

$$w_1 = c_N. \tag{6}$$

Then, for step $t = 1, \ldots, N-1$, we compare the current winner against the next stronger candidate:

$$w_{t+1} = \mathrm{Select}(Q,\, w_t,\, c_{N-t})\,, \tag{7}$$

where $\mathrm{Select}(\cdot)$ denotes the multimodal pairwise comparator conditioned on query Q. In each round t, the model reasons whether the current ladder winner w_t is better than the stronger challenger c_{N-t}. After $N-1$ rounds, the final survivor w_N is selected as the optimal evidence c^*. Each comparison jointly considers both modalities.

Crucially, unlike iterative approaches that require separate model calls for each comparison, MOLT executes the entire tournament in a *single autoregressive inference pass*. By encoding all multimodal candidates once and generating the comparison chain sequentially, we avoid redundant visual encoding overhead (detailed analysis in Appendix C).

To ensure faithful execution within this single pass, we constrain MOLT to a strict XML output: each comparison produces one `<round>` block (with `<compare>` and `<winner>`), followed by a final `<evidence>` tag for the selected ID (Table 14).

3.2 Training Optimization

We adopt a two-stage training paradigm consisting of supervised fine-tuning (SFT) for cold-start initialization, followed by reinforcement learning (RL) for policy refinement.

3.2.1 Supervised Fine-Tuning

We employ supervised fine-tuning (SFT) using instruction data generated by a large multimodal teacher model to endow the model with basic visual understanding and structured decision-making. We build the SFT set from a subset of E-VQA: for each sample, we take the top-4 retrieved paragraphs from different documents as negatives and the annotated evidence paragraph as the positive, pairing each paragraph with its document main image to form five multimodal candidates.

To reduce positional bias, we randomly shuffle the five candidates, assign random IDs, and record the positive ID. On this shuffled order, we run a fixed four-round sequential tournament, updating the winner each round. Teacher supervision is generated round-by-round: in each round, the teacher receives the query (image + question) and only the two candidates in that comparison, and outputs XML. When the positive candidate participates, we apply teacher-forced winner constraints to ensure it wins; otherwise, the teacher freely selects the winner. We then concatenate the four round outputs into a multi-round XML transcript (four `<round>` blocks plus the final `<evidence>` tag) as the SFT target.

3.2.2 Reinforcement Learning

After cold-start initialization, we further refine the model using reinforcement learning with the GRPO algorithm (Guo et al., 2025). During sampling, the input consists of the query (image and text), five candidate evidence pairs, and a structured prompt enforcing the Ladder Tournament protocol.

Composite Reward Design. To enable the model to faithfully learn the strict logic of the Ladder Tournament mechanism, we design a composite reward function. The total reward is defined as follow:

$$R_{\text{total}} = \lambda_1 R_{\text{fmt}} + \lambda_2 R_{\text{proc}} + \lambda_3 R_{\text{res}}, \tag{8}$$

where λ_1, λ_2, and λ_3 weight the format, process, and result rewards, respectively.

Format Reward (R_{fmt}). Given the strictly structured output requirements of our framework, we employ an XML parser to validate the generated syntax. A binary reward is assigned based on whether all required tags are present and correctly nested, which is detailed as follows:

$$R_{\text{fmt}} = \begin{cases} 1, & \text{if syntax is valid} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

This reward stabilizes training process by preventing format collapse and discouraging unstructured generations.

Process Reward (R_{proc}). The process reward constitutes the core of our reinforcement learning design and incorporates both logical consistency and dense supervision. For each tournament round t, we perform fine-grained evaluation: *Logical Consistency Check.* We verify whether the winner from round t correctly participates as an input candidate in round $t+1$. If this causal dependency is violated, an early stopping mechanism is triggered, and all

subsequent process rewards are set to zero. This explicitly enforces causal reasoning across comparison steps. *Step-wise Correctness (Dense Supervision).* At each comparison step, if the ground-truth candidate participates in the round and is correctly selected as the winner, an additional bonus reward is granted. Formally, the process reward is computed as:

$$R_{\text{proc}} = \sum_{t=1}^{T} v_t \left(r_{\text{step}} + w_t \cdot r_{\text{bonus}} \right), \tag{10}$$

where $v_t \in \{0, 1\}$ indicates whether the logical chain remains valid at step t, and $w_t \in \{0, 1\}$ indicates whether the ground-truth (oracle) candidate wins round t.

Crucially, combined with our *Weak-to-Strong* schedule, this accumulation mechanism introduces an implicit *difficulty-aware scaling*. Since lower-ranked candidates enter the tournament earlier, a ground-truth candidate buried at the bottom (hard sample) must survive more comparison rounds than a top-ranked one (easy sample), thereby accumulating a significantly higher total reward. This assigns larger gradient advantages to successfully recovering hard negatives, encouraging the model to prioritize resolving retrieval noise. For a formal derivation and visual analysis of this mechanism, please refer to Appendix D.

Result Reward (R_{res}). The result reward evaluates the correctness of the final decision. If the candidate ID emitted in the `<evidence>` tag matches the ground-truth ID, the model receives a reward of 1; otherwise, the reward is 0. The detailed function is defined as:

$$R_{\text{res}} = \begin{cases} 1, & \text{if matches ground truth} \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

4 Experiments

4.1 Datasets and Metrics

Datasets We conduct experiments on two benchmarks: E-VQA and InfoSeek (Mensink et al., 2023; Chen et al., 2023). E-VQA contains over one million samples, where each instance is annotated with a ground-truth (oracle) article and the corresponding evidence paragraph. The questions are categorized into single-hop and multi-hop types. The dataset is split into training, validation, and test sets with approximately 1.0M, 13.6K, and 58K samples, respectively. InfoSeek comprises 1.3 million VQA samples with annotated oracle articles, including a training set (934K) and a validation set (73K). The validation set is further divided into *Unseen-Entity* and *Unseen-Question* subsets to evaluate generalization. Following prior work (Yang et al., 2025; Caffagni et al., 2024; Yan and Xie, 2024), we evaluate our model on the single-hop test split (4.75K) of E-VQA using its provided knowledge base of 2M articles. For InfoSeek, since test annotations are not publicly available, we report results on the full validation set using a knowledge base of 100K Wikipedia entries.

Metrics We employ Recall@K to measure retrieval performance. For downstream VQA evaluation, we follow the official protocols of each dataset (Mensink et al., 2023; Chen et al., 2023), utilizing the BEM score for E-VQA and standard VQA Accuracy and Relaxed Accuracy for InfoSeek. In ablation studies, we additionally employ Evidence Selection Accuracy (Acc) to evaluate the model's paragraph selection capability.

4.2 Implementation Details

In this section, we briefly introduce some details of the various steps in our framework. For granular training configurations and model hyperparameters, we refer readers to Appendix E. Additionally, the full collection of prompt templates utilized across the pipeline is provided in Appendix G.

Model Architecture and Environment. We utilize the Qwen3-VL family as our backbone. We employ the lightweight Qwen3-VL-4B-Instruct (Team, 2025) as the unified backbone for the entire evidence acquisition pipeline, and utilize the larger Qwen3-VL-8B-Instruct for the final answer generator. Experiments are conducted on a single node equipped with four NVIDIA RTX 3090 (24GB) GPUs.

Data Construction Strategy. Since InfoSeek lacks paragraph-level ground-truth annotations, we rely on E-VQA for training the pipeline components. To prevent overfitting to specific high-frequency entities in the E-VQA training set, we implement a Page-Balanced Round-Robin Sampling strategy. We group training instances by their source Wikipedia page ID and sample iteratively across groups. Detailed statistics of the datasets and the specific training splits are provided in Appendix E.

Hyperparameter Configuration. In the MOLT, we set the candidate pool size to $N = 5$. For the composite reward function, we configure the

Method	Generator	RE.FT	Gen.FT	E-VQA	InfoSeek		
				Single-Hop	Unseen-Q	Unseen-E	All
Zero-shot							
LLaVA-1.5-7B	–	–	–	16.3	13.0	10.3	12.2
Qwen3-VL-8B	–	–	–	22.1	19.0	17.5	18.2
Retrieval-Augmented Methods							
Wiki-LLaVA	Vicuna-7B	✗	✓	17.7	30.1	27.8	28.9
RoRA-VLM	Vicuna-7B	✗	✓	20.3	27.3	25.1	26.9
EchoSight	LLaMA3-8B	✓	✗	19.4	–	–	27.7
ReflectiVA	LLaVA-MORE-8B	✓	✓	35.5	40.4	39.8	40.1
OMGM	LLaVA-1.5-7B	✓	✓	50.2	43.5	43.5	43.5
Our Method							
MOLT	Qwen3-VL-8B	✓	✗	55.9	38.3	36.6	37.4
MOLT	LLaVA-1.5-7B	✓	✓	56.7	45.2	44.6	44.9
MOLT	Qwen3-VL-8B	✓	✓	**57.5**	**45.5**	**45.2**	**45.3**

Table 1: Main results on E-VQA (Single-Hop) and InfoSeek. RE.FT and Gen.FT indicate whether the retrieval/reranking module and the answer generator are fine-tuned, respectively.

weights as $\lambda_1 = 0.2$ (format), $\lambda_2 = 0.5$ (process), and $\lambda_3 = 1.0$ (result). Within the process reward, the step validity reward is set to $r_{\text{step}} = 0.1$, and the winner bonus is $r_{\text{bonus}} = 0.2$.

4.3 Main Results

Answer Generation Results. Table 1 reports metrics on E-VQA and InfoSeek. Our proposed reranker consistently improves downstream answer generation by selecting higher-quality and less-confusable evidence. Notably, when using the same LLaVA-1.5-7B generator as the strong baseline OMGM, MOLT achieves 56.7% accuracy on E-VQA, surpassing OMGM (50.2%) by a significant margin of 6.5%. This confirms that the performance gain stems primarily from our robust reranking capability rather than a stronger generator. Furthermore, with the more advanced Qwen3-VL-8B as the generator, MOLT reaches a state-of-the-art accuracy of **57.5%**, outperforming prior systems like ReflectiVA (35.5%) and OMGM. This indicates that, beyond coarse retrieval, resolving visually near-duplicate yet textually plausible candidates in the reranking stage is a key driver of final answer correctness. MOLT also improves accuracy on InfoSeek and generalizes well to both *Unseen-Question* and *Unseen-Entity* subsets. When both reranker and generator fine-tuning are enabled, MOLT attains **45.3%** accuracy on the full validation set, with consistent gains across the generalization splits. This suggests that our sequential multimodal comparisons effectively mitigate hard negatives where distractor paragraphs match superficial cues but describe different entities.

Method	R@1	R@5	R@10	R@20
Wiki-LLaVA	3.3	–	9.9	13.2
ReflectiVA	15.6	36.1	–	49.8
EchoSight	36.8	47.9	48.8	48.8
OMGM	<u>42.8</u>	<u>55.7</u>	<u>58.1</u>	**58.7**
Ours (Doc sel.)	42.4	56.7	58.3	58.6
Ours (+MOLT)	**50.1**	**56.7**	**58.3**	<u>58.6</u>

Table 2: E-VQA retrieval performance (Recall@K). Best in bold, second-best underlined.

Method	R@1	R@5	R@10	R@20
Wiki-LLaVA	36.9	–	66.1	71.9
ReflectiVA	56.1	77.6	–	**86.4**
EchoSight	53.2	77.0	77.4	77.9
OMGM	<u>64.0</u>	<u>80.8</u>	<u>83.6</u>	84.8
Ours (Doc sel.)	60.1	81.0	83.8	84.8
Ours (+MOLT)	**65.6**	**81.0**	**83.8**	<u>84.8</u>

Table 3: InfoSeek retrieval performance (Recall@K). Best in bold, second-best underlined.

Retrieval Results. Table 2 and Table 3 report paragraph-level retrieval performance measured by Recall@K on E-VQA and InfoSeek, respectively. The coarse-to-fine retriever with paragraph pre-selection (Ours (Doc sel.)) establishes a strong baseline on both benchmarks. Notably, applying the proposed reranker (Ours (+MOLT)) significantly boosts Top-1 performance while maintaining the recall ceiling of the candidate pool. Specifically, on E-VQA, MOLT improves R@1 from 42.4% to 50.1% (+7.7%), and on InfoSeek, it increases R@1 from 60.1% to 65.6% (+5.5%). These gains indicate that MOLT effectively discriminates vi-

Passage Ext.	Grounding	MOLT	E-VQA	InfoSeek
✗	✗	✗	22.1	18.2
✓	✗	✗	50.6	40.2
✓	✗	✓	57.2	44.3
✓	✓	✓	**57.5**	**45.3**

Table 4: Step-wise ablation study on E-VQA and InfoSeek using Qwen3-VL-8B. We progressively incorporate Passage Extraction (Passage Ext.), Visual Grounding (Grounding), and the proposed MOLT reranker.

Method / Setting	Prompt Type	Acc (%)
Zero-shot Inference (Base Model)		
Base (Standard)	One-shot 5-way	79.4
Base (Ours)	Tournament	81.0
Trained Models (w/ Tournament Prompt)		
SFT Cold Start	Tournament	85.4
SFT + GRPO (w/o R_{proc})	Tournament	86.7
SFT + GRPO (Full)	Tournament	**88.3**
SFT + SAPO (Full)	Tournament	88.2

Table 5: Ablation study on E-VQA evidence selection. We compare the **Standard Selection** baseline (direct 1-of-5 choice) against our **Tournament** strategy across different optimization stages.

Comparison Schedule	E-VQA
Random Shuffle	85.5
Strong-to-Weak (Start with c_1)	86.3
Weak-to-Strong (Start with c_N)	**88.3**

Table 6: Ablation of candidate input ordering strategies. Strong-to-Weak: The tournament starts with the highest-scoring candidate (c_1), forcing it to survive all $N-1$ rounds. Weak-to-Strong (Ours): The tournament starts with the lowest-scoring candidates, introducing stronger candidates only in later rounds.

sually near-duplicate and textually plausible hard negatives, successfully promoting the oracle evidence to the top rank. Consequently, this accurate evidence selection directly translates to the downstream VQA improvements observed in Table 1.To intuitively demonstrate how MOLT resolves these ambiguities compared to baselines, we provide qualitative visualizations in Appendix H.

4.4 Ablation Studies.

We conduct ablation studies to quantify the contribution of each component in our MM-RAG pipeline and to analyze the effect of different training strategies for the proposed tournament reranker.

Step-wise component ablation. Table 4 reports step-wise results. Starting from the zero-shot baseline (22.1% on E-VQA), enabling Passage Ext. yields a substantial jump to 50.6%, confirming that retrieval is fundamental. Crucially, incorporating MOLT (without grounding) further improves performance to 57.2% on E-VQA and 44.3% on InfoSeek. This indicates that sequential pairwise comparisons effectively resolve visually near-duplicate hard negatives. Finally, integrating Grounding yields the best results (**57.5%**), suggesting that cropping helps the reranker focus on query-relevant regions and reduces background noise.

Training ablation for the reranker. Table 5 isolates the contributions of our tournament structure and training stages. First, in the zero-shot setting, our Ladder Tournament prompt achieves 81.0%, surpassing the Standard Selection baseline (79.4%) by **1.6%**. This confirms that sequential comparisons are more robust than one-shot selection. Next, SFT provides a crucial cold start. It adapts the model to the strict XML format, boosting accuracy to 85.4%. GRPO refinement brings further gains. Notably, adding the process reward (R_{proc}) improves performance from 86.7% to **88.3%**. This validates that enforcing step-wise consistency prevents the model from exploiting spurious correlations. Finally, SAPO (Gao et al., 2025) yields comparable performance (88.2%), suggesting that our composite reward design drives the improvements rather than the specific RL algorithm.

Effect of Comparison Schedule. To verify MOLT's sensitivity to candidate input order, we compare different schedules in Table 6. The Strong-to-Weak strategy, which forces the highest-scoring candidate (c_1) to participate in every round, yields 86.3%. In contrast, our Weak-to-Strong schedule achieves the best accuracy of **88.3%**, significantly outperforming both Strong-to-Weak and Random Shuffle (85.5%). This validates our design intuition: since the ground-truth is typically high-ranked, the Weak-to-Strong schedule effectively treats it as a "seeded player." By allowing the strongest candidate to bypass early comparisons and enter only the final decisive round, we minimize its exposure to error accumulation, preventing accidental elimination in earlier, noisier steps.

4.5 Training Visualization.

Figure 3 illustrates the training dynamics of the GRPO stage. We report moving averages for both

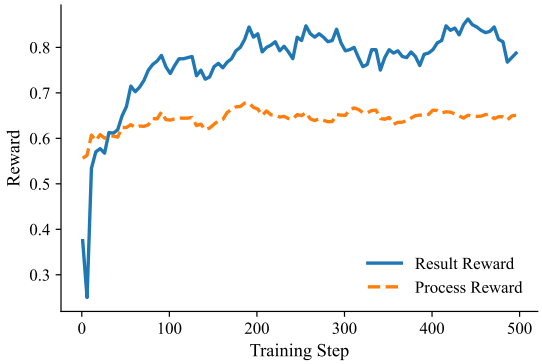

Figure 3: Rewards are shown for the first 500 training steps. Curves are smoothed with a moving average (window size 50) for clarity.

process and result rewards. Initially, the process reward rises rapidly and then stabilizes. This indicates that the model quickly masters the ladder tournament protocol, including candidate propagation and consistency. Notably, this reward flattens in later stages, suggesting the model avoids overfitting to procedural constraints. In contrast, the result reward shows a gradual but consistent upward trend. This demonstrates that after learning the structure, the model focuses on optimizing the final evidence selection."

4.6 Efficiency and Comparative Analysis.

Beyond accuracy, practical deployment requires efficiency. We conduct a comprehensive latency benchmark and compare MOLT against discriminative embedding baselines (e.g., BGE-VL). Our analysis shows that MOLT achieves competitive inference speeds through its one-pass design while significantly outperforming embedding methods in accuracy. Detailed results and discussions are provided in Appendix F.

5 Related Work

5.1 KB-VQA

Knowledge-Based VQA (KB-VQA) extends standard VQA (Antol et al., 2015) by requiring external knowledge unobservable in images. While early benchmarks (e.g., OK-VQA (Marino et al., 2019), A-OKVQA (Schwenk et al., 2022), ViQuAE (Lerner et al., 2022)) established foundations in reasoning and entity grounding, they lack fine-grained retrieval requirements. Conversely, recent datasets like InfoSeek (Chen et al., 2023) and E-VQA (Mensink et al., 2023) demand precise visual information-seeking. Consequently, Multimodal Retrieval-Augmented Generation (MM-RAG) has become the dominant paradigm. Most methods employ a *Retrieve–Rerank–Generate* pipeline, enhancing performance via multimodal knowledge bases (Deng et al., 2025), vision-guided or hierarchical retrieval (Yan and Xie, 2024; Lin et al., 2024; Caffagni et al., 2024; Yang et al., 2025; Qi et al., 2024), and noise mitigation through reconciliation or self-reflection (Tian et al., 2025; Hong et al., 2025; Compagnoni et al., 2025; Cocchi et al., 2025; Ling et al., 2025; Zhang et al., 2024).

5.2 LLM for Reranker

Reranking has evolved from discriminative cross-encoders to generative LLM-based formulations (Zhou et al., 2025b). Pairwise approaches prove effective by modeling relative relevance, with aggregators like PRP achieving strong performance (Qin et al., 2024). Efficiency is improved through batching and sorting-based schedules (Liusie et al., 2024; Chen et al., 2025), while structured prompts and in-context learning enhance robustness (Sinhababu et al., 2024; Hu et al., 2024). Furthermore, reinforcement learning allows direct optimization of ranking behavior via relevance rewards, a strategy extending to multimodal rankers (Zhuang et al., 2025; Liu et al., 2025; Xu et al., 2025).

6 Conclusion

In this work, we identified that evidence selection in KB-VQA is fundamentally a comparative task, yet prior methods treat it as isolated scoring. To bridge this gap, we proposed MOLT, a reranker that reformulates selection as a sequential multimodal ladder tournament. By integrating visual grounding with a Weak-to-Strong comparison schedule, MOLT effectively filters out hard negatives—candidates that are textually plausible but visually mismatched, or visually near-duplicate but distinct entities. Crucially, we demonstrated that this multi-round reasoning can be executed in a single inference pass, avoiding the latency of iterative pairwise approaches. Furthermore, our analysis of the SFT-then-GRPO training recipe reveals that our composite reward mechanism naturally creates a difficulty-aware curriculum, incentivizing the model to master the resolution of retrieval noise. Achieving state-of-the-art results on E-VQA and InfoSeek, MOLT proves that structured, process-supervised reasoning is a viable path for robust Multimodal RAG.

7 Limitations

While MOLT achieves state-of-the-art performance on KB-VQA benchmarks, we acknowledge limitations inherent to its design. First, as a reranking module within the *Retrieve–Rerank–Generate* pipeline, its performance is theoretically upper-bounded by the recall of the upstream retriever; if the correct evidence is not captured within the initial candidate pool (Top-N), MOLT cannot recover the ground truth regardless of its reasoning capabilities. Second, although our single-pass implementation is significantly more efficient than iterative pairwise approaches, generating reasoning chains for the ladder tournament still incurs higher token decoding costs compared to simple one-shot selection. This presents a trade-off between maximizing evidence precision and minimizing absolute latency, which may be a constraint for strictly real-time applications.

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: visual question answering. In *ICCV*, pages 2425–2433.

Davide Caffagni, Federico Cocchi, Nicholas Moratelli, Sara Sarto, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. 2024. Wiki-llava: Hierarchical retrieval-augmented generation for multimodal llms. In *CVPR Workshops*, pages 1818–1826.

Yang Chen, Hexiang Hu, Yi Luan, Haitian Sun, Soravit Changpinyo, Alan Ritter, and Ming-Wei Chang. 2023. Can pre-trained vision and language models answer visual information-seeking questions? In *EMNLP*, pages 14948–14968.

Yiqun Chen, Qi Liu, Yi Zhang, Weiwei Sun, Xinyu Ma, Wei Yang, Daiting Shi, Jiaxin Mao, and Dawei Yin. 2025. Tourrank: Utilizing large language models for documents ranking with a tournament-inspired strategy. In *WWW*, pages 1638–1652.

Federico Cocchi, Nicholas Moratelli, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. 2025. Augmenting multimodal llms with self-reflective tokens for knowledge-based visual question answering. In *CVPR*, pages 9199–9209.

Alberto Compagnoni, Marco Morini, Sara Sarto, Federico Cocchi, Davide Caffagni, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. 2025. ReAG: Reasoning-augmented generation for knowledge-based visual question answering. *arXiv Preprint*.

Lianghao Deng, Yuchong Sun, Shizhe Chen, Ning Yang, Yunfeng Wang, and Ruihua Song. 2025. Muka: Multimodal knowledge augmented visual information-seeking. In *COLING*, pages 9675–9686.

Chang Gao, Chujie Zheng, Xiong-Hui Chen, Kai Dang, Shixuan Liu, Bowen Yu, An Yang, Shuai Bai, Jingren Zhou, and Junyang Lin. 2025. Soft adaptive policy optimization. *arXiv preprint*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint*.

Yuyang Hong, Jiaqi Gu, Qi Yang, Lubin Fan, Yue Wu, Ying Wang, Kun Ding, Shiming Xiang, and Jieping Ye. 2025. Knowledge-based visual question answer with multimodal processing, retrieval and filtering. *arXiv Preprint*.

Chi Hu, Yuan Ge, Xiangnan Ma, Hang Cao, Qiang Li, Yonghua Yang, Tong Xiao, and Jingbo Zhu. 2024. Rankprompt: Step-by-step comparisons make language models better reasoners. In *LREC/COLING*, pages 13524–13536.

Paul Lerner, Olivier Ferret, Camille Guinaudeau, Hervé Le Borgne, Romaric Besançon, José G. Moreno, and Jesús Lovón-Melgarejo. 2022. Viquae, a dataset for knowledge-based visual question answering about named entities. In *SIGIR*, pages 3108–3120.

Weizhe Lin, Jingbiao Mei, Jinghong Chen, and Bill Byrne. 2024. Preflmr: Scaling up fine-grained late-interaction multi-modal retrievers. In *ACL*, volume 1, pages 5294–5316.

Zihan Ling, Zhiyao Guo, Yixuan Huang, Yi An, Shuai Xiao, Jinsong Lan, Xiaoyong Zhu, and Bo Zheng. 2025. MMKB-RAG: A multi-modal knowledge-based retrieval-augmented generation framework. *arXiv Preprint*.

Wenhan Liu, Xinyu Ma, Weiwei Sun, Yutao Zhu, Yuchen Li, Dawei Yin, and Zhicheng Dou. 2025. Reasonrank: Empowering passage ranking with strong reasoning ability. *arXiv Preprint*.

Adian Liusie, Vatsal Raina, Yassir Fathullah, and Mark J. F. Gales. 2024. Efficient LLM comparative assessment: A product of experts framework for pairwise comparisons. In *EMNLP*, pages 6835–6855.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. OK-VQA: A visual question answering benchmark requiring external knowledge. In *CVPR*, pages 3195–3204.

Thomas Mensink, Jasper R. R. Uijlings, Lluís Castrejón, Arushi Goel, Felipe Cadar, Howard Zhou, Fei Sha, André Araújo, and Vittorio Ferrari. 2023. Encyclopedic VQA: visual questions about detailed properties of fine-grained categories. In *ICCV*, pages 3090–3101.

Jingyuan Qi, Zhiyang Xu, Rulin Shao, Yang Chen, Di Jin, Yu Cheng, Qifan Wang, and Lifu Huang. 2024. Rora-vlm: Robust retrieval-augmented vision language models. *arXiv Preprint*.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2024. Large language models are effective text rankers with pairwise ranking prompting. In *NAACL-HLT (Findings)*, pages 1504–1518.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-OKVQA: A benchmark for visual question answering using world knowledge. In *ECCV*, volume 13668 of *Lecture Notes in Computer Science*, pages 146–162.

Nilanjan Sinhababu, Andrew Parry, Debasis Ganguly, Debasis Samanta, and Pabitra Mitra. 2024. Few-shot prompting for pairwise ranking: An effective non-parametric retrieval model. In *EMNLP (Findings)*, pages 12363–12377.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Yang Tian, Fan Liu, Jingyuan Zhang, Victoria W., Yupeng Hu, and Liqiang Nie. 2025. Core-mmrag: Cross-source knowledge reconciliation for multimodal RAG. In *ACL*, volume 1, pages 32967–32982.

Mingjun Xu, Jinhan Dong, Jue Hou, Zehui Wang, Sihang Li, Zhifeng Gao, Renxin Zhong, and Hengxing Cai. 2025. MM-R5: multimodal reasoning-enhanced reranker via reinforcement learning for document retrieval. *arXiv Preprint*.

Yibin Yan and Weidi Xie. 2024. Echosight: Advancing visual-language models with wiki knowledge. In *EMNLP (Findings)*, pages 1538–1551.

Wei Yang, Jingjing Fu, Rui Wang, Jinyu Wang, Lei Song, and Jiang Bian. 2025. OMGM: orchestrate multiple granularities and modalities for efficient multimodal retrieval. In *ACL*, volume 1, pages 24545–24563.

Tao Zhang, Ziqi Zhang, Zongyang Ma, Yuxin Chen, Zhongang Qi, Chunfeng Yuan, Bing Li, Junfu Pu, Yuxuan Zhao, Zehua Xie, Jin Ma, Ying Shan, and Weiming Hu. 2024. mr^{2}ag: Multimodal retrieval-reflection-augmented generation for knowledge-based VQA. *arXiv Preprint*.

Junjie Zhou, Yongping Xiong, Zheng Liu, Ze Liu, Shitao Xiao, Yueze Wang, Bo Zhao, Chen Jason Zhang, and Defu Lian. 2025a. Megapairs: Massive data synthesis for universal multimodal retrieval. In *ACL*, volume 1, pages 19076–19095.

Yinxin Zhou, Qin Luo, Bin Feng, and Bang Wang. 2025b. Large language models for reranking: A survey. *Authorea Preprints*.

Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2025. Rank-r1: Enhancing reasoning in llm-based document rerankers via reinforcement learning. *arXiv Preprint*.

A Retrieval Stage Details

We adopt a coarse-to-fine document retrieval pipeline followed by query-conditioned passage extraction, a strategy also used in OMGM (Yang et al., 2025).

A.1 Stage-1: Coarse-Grained Cross-Modal Document Retrieval (CLIP)

Given a multimodal query $Q = (I_q, T_q)$, we first retrieve a small set of potentially relevant documents from a large-scale knowledge base $\mathcal{K} = \{D_1, \ldots, D_{N_\mathcal{K}}\}$. Each document D_i is associated with a short summary text S_i and a main image I_i. In the coarse retrieval stage, we use an efficient CLIP-style cross-modal encoder to map the query image and document summaries into a shared embedding space.

Index Construction. We encode each summary S_i into a dense vector $\mathbf{s}_i$ and build a FAISS index over $\{\mathbf{s}_i\}$ for maximum inner-product search.

Query Encoding and Retrieval. We encode the query image I_q into a dense vector $\mathbf{v}_q$. The Top-K candidate documents are retrieved as:

$$\mathcal{D}_K = \text{Top-}K\big(\langle \mathbf{v}_q, \mathbf{s}_i \rangle\big), \tag{12}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. We denote the coarse retrieval score as $s_{\text{clip}}(Q, D_i) = \langle \mathbf{v}_q, \mathbf{s}_i \rangle$.

A.2 Stage-2: Multimodal Fusion Document Reranking (Q-Former)

The Top-K set $\mathcal{D}_K$ produced by Stage-1 prioritizes recall but may contain visually near-duplicate or textually confusable documents. We therefore apply a Q-Former-based multimodal fusion reranker to refine the candidate list.

For each document $D_i \in \mathcal{D}_K$, we compute a multimodal relevance score $s_{\text{qf}}(Q, D_i)$ using the query (I_q, T_q) and the document main image and summary (I_i, S_i). We then keep the Top-N documents:

$$\mathcal{D} = \text{Top-}N\big(s_{\text{qf}}(Q, D_i) \;\big|\; D_i \in \mathcal{D}_K\big). \tag{13}$$

In our implementation, the Q-Former reranker adopts late interaction to aggregate token-level interactions into a single similarity score.

A.3 Query-Conditioned Passage Extraction

After selecting the Top-N documents $\mathcal{D}$, we extract one query-relevant passage from each document to form the evidence candidates used by downstream grounding and reranking.

For each document $D_i \in \mathcal{D}$, the document text T_i is segmented into a set of passages $\{P_{i,1}, P_{i,2}, \ldots, P_{i,M_i}\}$. Conditioned on the query text T_q, a passage extractor selects the most relevant passage index:

$$k_i = \mathrm{VLM}_{\mathrm{sel}}\big(T_q, \{P_{i,m}\}_{m=1}^{M_i}\big), \tag{14}$$

and the extracted passage is denoted as:

$$P_i^{\mathrm{para}} = P_{i,k_i}. \tag{15}$$

This yields N document-level evidence candidates $\{(D_i, P_i^{\mathrm{para}})\}_{i=1}^{N}$, where each candidate consists of the document main image I_i and the extracted passage P_i^{para}.

A.4 Plug-and-Play Interface with MOLT

The proposed reranking module (MOLT) is retriever-agnostic and operates on the Top-N extracted candidates produced above. Concretely, MOLT takes as input the query $Q = (I_q, T_q)$ and the candidate set $\{(P_i^{\mathrm{para}}, I_i)\}_{i=1}^{N}$ (after optional grounding/cropping in Sec. 3.1), and performs sequential tournament comparisons to select the final evidence passage for answer generation.

B Grounding Training Details

RL Formulation for Grounding. We formulate the visual grounding task as a Reinforcement Learning problem. Instead of relying on ground-truth bounding box annotations—which are absent in the E-VQA dataset—we use the image and question pairs as prompts to optimize the model policy. Given a query $Q = (I_q, T_q)$, the model policy π_θ generates a bounding box action $B = [x_1, y_1, x_2, y_2]$. The objective is to maximize the expected reward that measures the semantic alignment between the cropped region and the question.

Reward Function. We design a reference-free reward function using EVA-CLIP-8B. For a generated bounding box B, we first validate its format. Invalid outputs (e.g., syntax errors or coordinates violating $x_1 < x_2$) receive a penalty reward R_{penalty}. For valid boxes, we crop the image region I_B^{crop} (with a padding ratio ρ) and compute the alignment score:

$$R(B) = \max\left(0, \cos\left(\mathbf{v}_{\mathrm{img}}(I_B^{\mathrm{crop}}), \mathbf{v}_{\mathrm{text}}(T_q)\right)\right) \tag{16}$$

This reward signal incentivizes the model to autonomously discover and localize the most semantically relevant image regions.

Training Setup. We use Qwen3-VL-4B-Instruct as the policy initialization. Training is conducted using Reinforcement Learning for 1 epoch over 10K prompts sampled from the E-VQA training split. The maximum generation length is set to 128 tokens to accommodate the coordinate output. The prompt template guiding the model to generate the specific coordinate format is provided in Table 11.

C Efficiency via Single-Pass Inference

Motivation. Standard pairwise reranking is typically implemented as an *iterative* pipeline: given N candidates, the system performs $N-1$ discrete pairwise comparisons. In this setup, the *current winner* must be carried over to the next round, meaning the winner candidate (including its image) is re-input to the model for each subsequent comparison. Consequently, the computationally expensive visual encoder is forced to recompute image features for the surviving candidate multiple times, leading to significant redundant computation.

Single-pass tournament inference. MOLT consolidates the entire ladder-style tournament into a *single* model inference pass. All candidates (cropped image regions and paragraphs) are provided in one input sequence. During the initial *pre-fill* phase, the model encodes each candidate image *exactly once* and stores the resulting hidden states in the KV cache. Subsequent multi-round comparisons are executed *internally* by the autoregressive decoder, attending to the cached representations without re-running the visual encoder. This design completely eliminates redundant visual encoding and avoids fragmented cache rebuilds.

Computational implications. Let C_{vis} and C_{lm} denote the computational cost of one visual encoding forward pass and one language-model decoding step, respectively. In an iterative pairwise pipeline with N candidates, the visual cost scales up to $O(N^2)$ in the worst case (where the same winner is re-encoded $N-1$ times). In contrast, MOLT incurs

Aspect	Iterative Pairwise	MOLT
Inference calls	$N-1$	1
Visual encoding	Repeated	Once per candidate
KV cache	Rebuilt	Reused
Comparison	External loop	Internal attention

Table 7: Efficiency comparison between standard iterative pairwise reranking and **MOLT**. **MOLT** performs the full tournament in a single inference pass.

a fixed visual cost of $O(N)$ (each candidate encoded once) regardless of the tournament outcome. In practice, this effectively shifts the overhead from expensive visual re-computation to lightweight token decoding, which is highly efficient on modern accelerators.

D Analysis of Difficulty-Aware Reward Scaling

In this section, we provide a detailed analysis of how the Weak-to-Strong tournament schedule, combined with our cumulative process reward design, constructs an intrinsic difficulty-aware learning objective. We visualize this mechanism in Figure 4 and provide a formal derivation below.

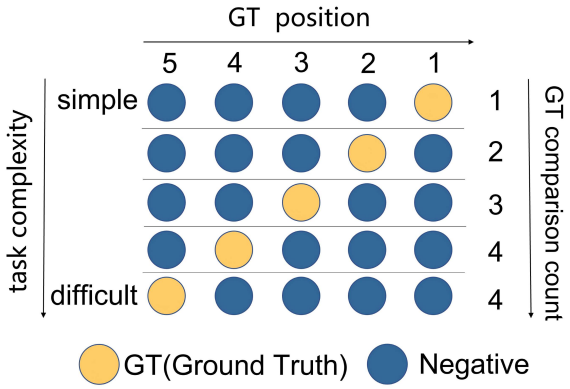

Figure 4: **Visualizing Task Complexity and Reward Scaling.** The x-axis represents the initial retrieval rank of the Ground Truth (GT) candidate, where 5 represents the lowest retrieval score and 1 represents the highest. The y-axis (right) shows the number of comparisons the GT must win to be selected. A GT retrieved at rank 5 (Difficult) must survive 4 consecutive rounds (bottom row), whereas a GT at rank 1 (Simple) enters only in the final round (top row). Since rewards are accumulated per winning step, harder tasks yield significantly higher total rewards.

Mechanism Explanation. As illustrated in Figure 4, the input schedule dictates the "survival depth" required for the ground-truth candidate. This creates two distinct scenarios:

- **Simple Cases (GT Rank 1):** When the retriever is confident (GT is ranked 1st, i.e., highest score), the GT candidate is introduced only in the final round. The model performs only **1 comparison** involving the GT. Consequently, the task complexity is minimal, and the cumulative process reward is low ($1 \times r_{\text{bonus}}$).

- **Difficult Cases (GT Rank 5):** When the retriever performs poorly (GT is ranked 5th, i.e., lowest score), the GT candidate serves as the initial seed and must participate starting from the very first step. To become the final evidence, it must defeat 4 negative candidates in a row. This corresponds to 4 comparisons, representing high task complexity. Consequently, the cumulative process reward is maximized ($4 \times r_{\text{bonus}}$).

Mathematical Derivation. To formalize this intuition, let $\mathcal{C} = [c_1, c_2, \ldots, c_N]$ be the list of candidates sorted by retrieval confidence in descending order. Let $r^* \in [1, N]$ denote the rank of the ground-truth evidence. In our tournament schedule, the candidate with the lowest rank (c_N) initializes the winner bucket. Any subsequent candidate with rank $r < N$ enters the tournament as a challenger at round step $t = N - r$.

For the ground-truth candidate c_{r^*} to be selected, it must satisfy two conditions: (1) win the round where it enters (either as the initial seed or a challenger), and (2) continue to win all subsequent rounds against higher-ranked candidates. The number of successful comparison rounds required for the ground truth c_{r^*} is defined as:

$$K_{\text{wins}} = \begin{cases} N-1 & \text{if } r^* = N \text{ (Lowest Score)} \\ r^* & \text{if } 1 \le r^* < N \end{cases} \tag{17}$$

(Note: For $r^* = 1$, $K_{wins} = 1$, which fits the second case $r^* < N$ if we treat the upper bound inclusive of logic, but distinct separation highlights the extremes.)

Since the process reward accumulates a bonus r_{bonus} for each successful round involving the ground truth, the maximum obtainable reward is proportional to difficulty. Assuming perfect format compliance and logical validity ($v_t = 1$), the total

Datasets	Sel. Train	MOLT Train	Gen. Train	Test	KB
E-VQA	20K	12K	20K	4,750	2M
InfoSeek	–	–	20K	71,335	100K

Table 8: Statistics of the InfoSeek and E-VQA datasets used in our experiments. **Sel. Train**, **MOLT Train**, and **Gen. Train** correspond to the training samples used in passage selection, multimodal ladder tournament reranking, and answer generation, respectively.

reward approximates:

$$R_{\text{total}}(c_{r^*}) \approx K_{\text{wins}} \cdot r_{\text{bonus}} + R_{\text{res}} + \text{const.} \quad (18)$$

Thus, a "Hard" sample (e.g., $r^* = N$) offers a potential reward significantly higher than an "Easy" sample ($r^* = 1$), scaling linearly with the number of required wins.

Curriculum Effect in Gradient Optimization. This linear relationship between retrieval difficulty (r^*) and reward magnitude creates an implicit curriculum for the reinforcement learning stage. In the GRPO framework, the policy gradient is scaled by the advantage term $A = R - \bar{R}$, where $\bar{R}$ is the group baseline.

1. For **simple cases** ($r^* = 1$), most sampled outputs successfully select the GT. The variance in rewards is low, leading to small advantage values. The model maintains performance but gradient updates are limited.

2. For **difficult cases** ($r^* = 5$), the GT is buried deep in the list. Early in training, the model likely fails to sustain the GT through all 4 rounds, resulting in a low baseline $\bar{R}$. When the model successfully recovers such a hard instance, it receives a large cumulative reward R, resulting in a large positive advantage ($A \gg 0$).

This mechanism ensures that optimization is dominated by *hard negatives*, incentivizing the system to prioritize resolving retrieval noise rather than overfitting to easy samples.

E More Implementation Details

Generator Training and Inference. We adopt Qwen3-VL-8B-Instruct as the generator model. We construct the generator training data by sampling 20K instances from E-VQA and 20K instances from InfoSeek, resulting in a total of 40K samples. For each instance, we apply the document retrieval and paragraph extraction pipeline to obtain the associated evidence passage, and then build a VQA dataset augmented with retrieved evidence for generator training. The generator is trained for 3 epochs on this combined dataset. Since E-VQA and InfoSeek impose different requirements on answer formats and evaluation protocols, we design dataset-specific prompting strategies for the two datasets (as detailed in Table 15 and Table 16). We implement the fine-tuning process using the ms-swift framework. To ensure training efficiency, we utilize LoRA (Low-Rank Adaptation) with a rank of $r = 8$ and $\alpha = 32$ targeting all linear modules, while keeping the Vision Transformer (ViT) and the aligner frozen. The training is conducted on 4 NVIDIA GPUs with DeepSpeed ZeRO-2 optimization to manage memory usage. We set the learning rate to 1e-4 with a warmup ratio of 0.05 and a cosine decay scheduler. The effective global batch size is set to 16 (batch size 1 per device $\times$ 4 devices $\times$ 4 gradient accumulation steps), and the maximum sequence length is restricted to 2,548 tokens. Specific environment variables for Qwen3-VL image processing are set to default values (e.g., `IMAGE_MAX_TOKEN_NUM=648`).

Passage Selection Training. We utilize the Qwen3-VL-4B-Instruct model as the backbone for the query-conditioned passage extractor. The model is fine-tuned on the 20K passage selection samples derived from E-VQA using the `ms-swift` framework. We apply LoRA fine-tuning ($r = 8, \alpha = 32$) targeting all linear modules for 3 epochs, while keeping the vision encoder and aligner frozen. To accommodate the input of document-level contexts containing multiple paragraphs, we significantly expand the maximum sequence length to 8,192 tokens and increase the image token budget to 1,024. Due to the increased memory requirements from the long context, we employ DeepSpeed ZeRO-3 optimization distributed across 4 NVIDIA 3090 GPUs. The learning rate is configured to `1e-4` with a cosine scheduler and a warmup ratio of 0.05. The effective global batch size is 16 (batch size 1 per device $\times$ 4 devices $\times$ 4 gradient accumulation steps).

Reranker Training Details. To initialize the model with basic visual understanding and structured decision-making, we conduct supervised fine-tuning (SFT) using instruction data generated by a large multimodal teacher model. We use `Qwen3-VL-235B-Instruct` (Team, 2025) as the

teacher model, which is also adopted in the official Qwen RL training pipeline, ensuring consistent output distributions.

F Further Analysis

In this section, we provide a deeper analysis of the proposed MOLT framework, focusing on its comparative advantage against discriminative baselines and its real-world inference efficiency.

F.1 Comparison with Discriminative Reranker

To further validate the superiority of our generative tournament approach, we compare MOLT against BGE-VL-v1.5-mmeb (Zhou et al., 2025a), a state-of-the-art multimodal embedding model optimized for retrieval tasks. For the baseline, we compute the cosine similarity between the multimodal query embedding and the candidate embeddings to rank the paragraphs.

Reranking Method	Selection Acc	VQA Score
BGE-VL-v1.5-mmeb†	80.3	50.4
MOLT (Ours)	**88.3**	**57.5**

Table 9: Comparison of evidence selection performance on E-VQA (1-of-5 reranking setting). †: We use the cosine similarity of the query and candidate embeddings for ranking.

As shown in Table 9, MOLT significantly outperforms the embedding-based method, achieving an improvement of **8.0%** in selection accuracy and **7.1%** in downstream VQA performance. This highlights that while embedding models are efficient for coarse retrieval, resolving hard negatives requires the fine-grained reasoning inherent to our tournament mechanism.

F.2 Efficiency Analysis

To demonstrate the real-world feasibility of MOLT, we conducted a comprehensive latency benchmark on a single NVIDIA RTX 3090 GPU. We measured the end-to-end latency for processing a single query paired with 5 multimodal candidates (each comprising a dynamic image region and text). We compared three distinct experimental setups:

1. **BGE-VL-v1.5-mmeb**: A state-of-the-art embedding baseline using the heavy LLaVA-1.6 backbone (∼7.6B params). Since reranking involves query-specific image cropping that cannot be pre-indexed, we measured the cost of on-the-fly encoding for the query and all 5 candidates.

2. **Iterative Pairwise**: The standard generative approach that invokes the VLM $N-1$ times sequentially to complete the tournament.

3. **MOLT (Ours)**: Our proposed framework using the parameter-efficient Qwen3-VL backbone (∼4B params), executing the tournament in a single pass with vLLM optimization.

Method	Backbone	Latency
BGE-VL-v1.5-mmeb	LLaVA-1.6 (7.6B)	4,633 ms
Iterative Pairwise	Qwen3-VL (4B)	10,304 ms
MOLT (Ours)	**Qwen3-VL (4B)**	**4,231 ms**

Table 10: Inference latency comparison on a single RTX 3090. BGE-VL incurs high latency due to the heavy computational cost of on-the-fly visual encoding for multiple high-resolution candidate images using a 7.6B backbone. MOLT achieves the lowest latency by leveraging a lighter 4B backbone and a one-pass tournament mechanism, effectively eliminating the overhead of repeated model invocations.

The results in Table 10 reveal two critical insights regarding efficient multimodal reranking.

First, online visual encoding is a significant bottleneck for embedding models. While embedding models are typically efficient for retrieval over precomputed indexes, the reranking stage necessitates real-time processing of dynamic candidate images. Encoding the visual features of six high-resolution images (1 query + 5 candidates) in a single forward pass with a 7.6B parameter backbone incurs a substantial computational burden (∼4.6s), negating the typical speed advantage of discriminative models.

Second, the One-pass design effectively eliminates generative overhead. The standard Iterative Pairwise approach is prohibitively slow (∼10.3s) due to the overhead of repeated model invocations and redundant memory access for the same visual contexts. MOLT bridges this gap by consolidating the visual processing and reasoning into a single optimized pass. Consequently, MOLT achieves the lowest latency (∼4.2s) among all methods, demonstrating that with an optimized one-pass architecture and a parameter-efficient backbone, deep generative reasoning can be deployed with latency comparable to, or even lower than, heavy visual encoders.

G Prompts Used in MM-RAG Pipelines

In this section, we present the specific prompt templates utilized across the entire pipeline. We categorize them into four logical stages: retrieval and grounding, reranking, answer generation, and baseline comparison.

Retrieval and Grounding Prompts. The pipeline begins with extracting evidence and grounding visual contexts. Table 13 details the Paragraph Selection Prompt, which instructs the model to identify the most relevant section ID from a Wikipedia article. Subsequently, Table 11 presents the Visual Grounding Prompt, designed to localize and output the pixel coordinates of the query-relevant image region.

MOLT Reranking Prompts. We employ two distinct prompts for the tournament reranking stage. The Teacher Prompt (Table 12) is used for data construction, instructing the teacher model to perform isolated pairwise comparisons. The Student Prompt (Table 14) is utilized during inference and RL training. It strictly enforces the sequential Ladder Tournament protocol and XML formatting (e.g., `<round>` and `<evidence>` tags), which is essential for parsing reasoning chains and computing rewards.

Answer Generation Prompts. For the final generation stage, we tailor the prompts to the specific annotation styles of the benchmarks. Table 15 and Table 16 show the templates for E-VQA and InfoSeek, respectively. These prompts ensure the generator produces concise answers or specific entity formats aligned with the evaluation metrics.

One-shot Baseline Prompt. To benchmark our method against standard approaches, we use a One-shot 5-way Selection Prompt (Table 17). This prompt instructs the model to directly select the single best candidate from the list without intermediate reasoning steps.

H Qualitative Analysis

Figure 5 visualizes the evidence selection performance. Strong embedding-based retrievers like BGE-VL-v1.5-mmeb often struggle with hard negatives. For instance, in the fungi example (top-left), the baseline incorrectly selects a visually similar "look-alike" species (*Exidia nigricans*) instead of the ground truth (*Bulgaria inquinans*). Similarly, in the spider example (top-middle), the baseline is distracted by a textually plausible but generic description, whereas MOLT correctly aligns the specific visual feature ("orb-shaped web") with the precise habitat evidence. These cases demonstrate that our sequential tournament mechanism provides the necessary fine-grained discrimination to filter out subtle distractors that confuse one-shot retrievers.

System:
You are an AI assistant specialized in visual grounding.
Your task is to locate and return the bounding box coordinates of the most relevant region in the image based on the user's question.
TASK:
Locate and return the bounding box coordinates of the region in the image that is most directly related to the question.

REQUIREMENTS:
1) Return ONLY ONE bounding box — the most relevant region.
2) Use PIXEL COORDINATES (not normalized values).
3) Format: `[x1, y1, x2, y2]`.
4) (x_1, y_1) is the top-left corner; (x_2, y_2) is the bottom-right corner.
5) The user will provide the image size. Ensure your coordinates are within those bounds.
6) If no relevant region exists OR the entire picture is related to the question, return: `[]`.

OUTPUT:
Return ONLY the bounding box (or `[]`), no other text.

User:
Identify the single most relevant region in this image for the question:
Image: <image>
Question: `{question}`
Image size is `{img_width}x{img_height}` pixels.

Table 11: Prompt template used for training and inference of the visual grounding module.

System:
You are a helpful assistant assisting in selecting the best matching document for a visual query.
I will present a Query Image/Question and two Candidate documents (identified by their IDs).
Your task is to determine which candidate implies the correct answer or matches the query best.

OUTPUT FORMAT:
For each round, output XML:
<think>
Detailed reasoning based ONLY on visual and textual evidence. Do not mention you know the answer beforehand.
</think>
<winner>[ID_OF_WINNER]</winner>

LENGTH LIMIT:
For each round, your output must be **no more than 200 English words**. Be concise but clear.

Table 12: Teacher prompt used for constructing **MOLT** training data.

System:
Read the Wikipedia article and identify the single most relevant section ID that answers the user's query.
Output ONLY the integer ID of the section (e.g., 0).
Do not output any other text or explanation.

User:
Query: {question}
Image: <img></img>
Wiki Article: {formatted_wiki_content}
Best Section ID:

Table 13: Prompt template used for paragraph selection.

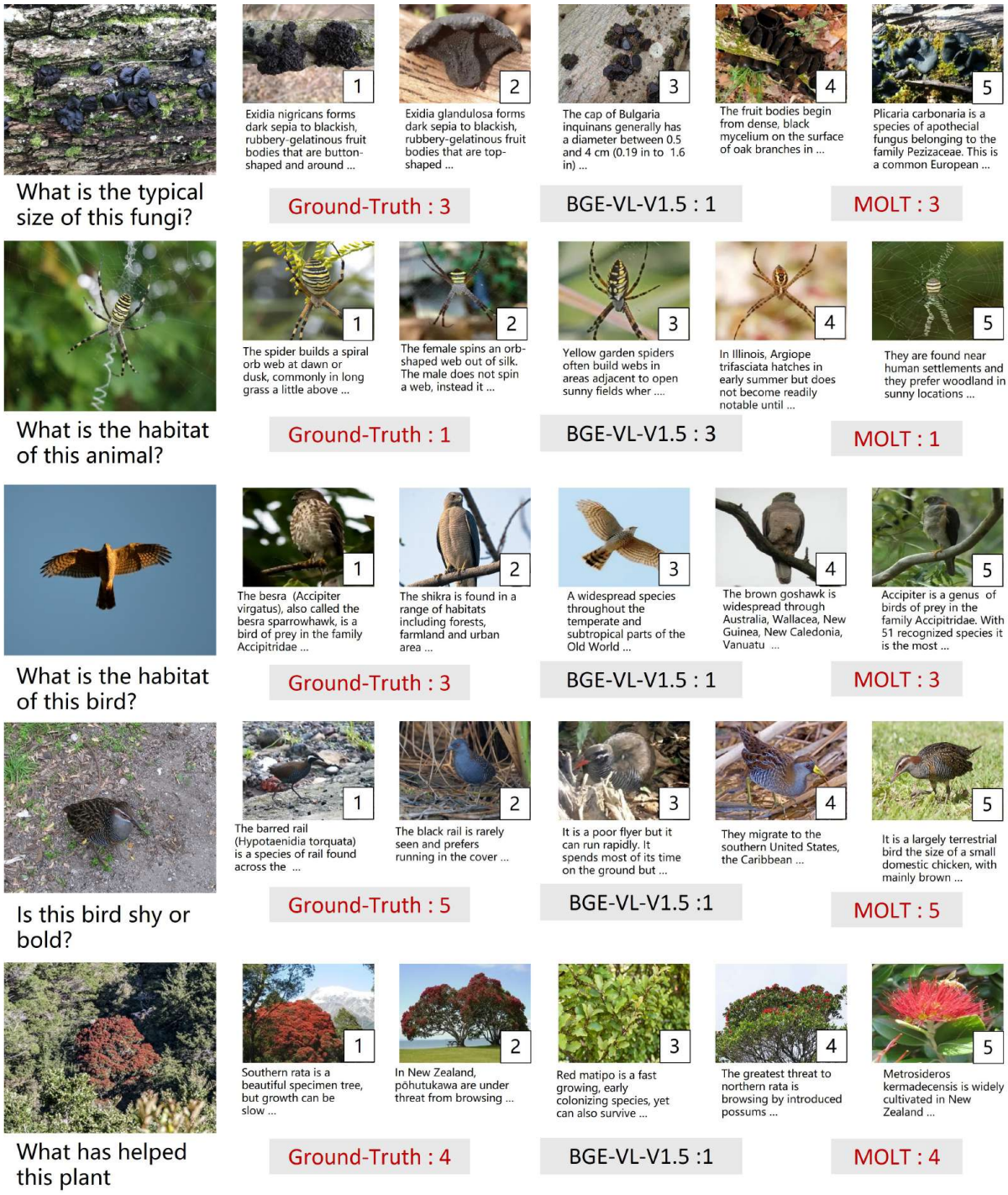

Figure 5: **Qualitative comparison on E-VQA.** We compare MOLT against the BGE-VL-v1.5-mmeb baseline. The baseline often succumbs to hard negatives candidates that are visually similar (e.g., look-alike fungi) or semantically related (e.g., generic spider descriptions) but factually wrong. MOLT effectively resolves these ambiguities through fine-grained pairwise comparisons.

System:
You are an expert multimodal reranker.
Your goal:
Given a user query (text + image) and a list of candidates (each with an image region and a paragraph),
select the SINGLE most relevant candidate ID.

Ladder Tournament strategy:
1. Initialize Current Best = Candidate [N].
2. For i = N-1..1, compare Current Best vs. Candidate [i] ONLY and update Current Best to the winner.
3. After all comparisons, Current Best is the final evidence.

OUTPUT FORMAT:
For EACH step, output one <round>...</round>block:
<round>
  <compare>[best_id] vs [next_id]</compare>
  <think>Short reasoning why the winner is better.</think>
  <winner>[winner_id]</winner>
</round>
After all rounds, output one final line:
<evidence>[Best_ID]</evidence>

CRITICAL:
Output ONLY <round>blocks and the final <evidence>tag.
Do NOT output any other text.

Table 14: Student prompt used for **MOLT** inference.

System:
Given the provided image, the associated question, and relevant information from Wikipedia,
respond to the question concisely and directly without any additional explanation.
If the knowledge does not contain the information needed to answer the question,
you should use your own knowledge to answer it.

User:
Image: <image>
Knowledge: {retrieved_section_text}
Question: {question}
Answer:

Table 15: Prompt template used for answer generation on E-VQA.

System:
You are a visual question answering assistant with encyclopedic knowledge.
Look at the image and give a direct, concise answer without explanation.
If the knowledge does not contain the information needed to answer the question,
you should use your own knowledge to answer it.
If you need to answer questions about numbers or time,
please output the corresponding numerical format directly.

User:
Image: <image>
Knowledge: {retrieved_section_text}
Question: {question}
Answer:

Table 16: Prompt template used for answer generation on InfoSeek.

System:
You are a multimodal reranker. Given a query (image + question) and five candidates (each with an image region and a paragraph), select the single candidate that is most relevant to the question and best matches the query image. Output only the candidate ID (1–5), nothing else.

User:
Query Image: <image>
Question: `{question}`

Candidates:
1) Image: <image_1>
   Paragraph: `{para_1}`
2) Image: <image_2>
   Paragraph: `{para_2}`
3) Image: <image_3>
   Paragraph: `{para_3}`
4) Image: <image_4>
   Paragraph: `{para_4}`
5) Image: <image_5>
   Paragraph: `{para_5}`

Output:
Return **ONLY** one number in `{1,2,3,4,5}`.

Table 17: One-shot 5-way evidence selection prompt used as a baseline reranker.