GETTING THE EVIDENCE RIGHT: ADAPTING RETRIEVAL AND VISUAL READING FOR MULTIMODAL AGENTIC SEARCH

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal search agents must retrieve evidence that addresses intermediate questions and extract the facts needed to advance their reasoning. Existing approaches typically train the agent while keeping the retriever fixed. Retrieved candidates differ in how fully they support an intermediate query, and even relevant images can be misread. We introduce SPAR (**S**tructuring and **P**rojecting Evidence for **A**cquisition and **R**eading), a framework for adapting retrieval and visual reading. SPAR analyzes what each intermediate query requires and records the facts supported by each candidate. This shared analysis yields graded relevance supervision that prioritizes complete support while retaining useful partial evidence. For visual reading, SPAR reuses the image-grounded facts recorded in the same annotations to teach the agent to extract query-relevant information. We then adapt the agent's search policy on complete trajectories with the adapted retriever held fixed. On MultiModalQA, SPAR achieves 76.03% answer accuracy, exceeding the strongest baseline by 8.43 percentage points. Graded relevance supervision outperforms ungraded supervision both before and after policy adaptation. Under fixed evidence, the SPAR agent extracts visual facts more accurately than the policy-only agent. Without further training, SPAR also transfers to WebQA, MuSiQue, and HotpotQA, reaching 61.89% answer accuracy on HotpotQA. We will release our code upon acceptance.

1 INTRODUCTION

Multimodal search agents answer knowledge-intensive questions by interleaving reasoning with retrieval over text and images (Jiang et al., 2025; Wu et al., 2026; Ning et al., 2026). As reasoning unfolds, they formulate queries to resolve intermediate questions and use the retrieved evidence to decide what to search for next or how to answer. Recent work has used reinforcement learning to improve this process, training agents to formulate queries, invoke search tools, and reason over the returned evidence (Wu et al., 2026; Geng et al., 2026; Narayan et al., 2025). These approaches train the agent while leaving the retrieval backend unchanged. However, the usefulness of each search step also depends on whether the returned evidence addresses the current query's information needs. Adapting the retriever to these intermediate needs is therefore a complementary objective to improving the agent's search policy.

Training the retriever on the agent's own search queries offers a way to improve the evidence available during reasoning (Liu et al., 2026; Chen et al., 2026). These queries reflect the intermediate questions the agent encounters, providing a basis for adapting retrieval to its information needs. The training signal should therefore reflect how each candidate helps resolve the current query. Evidence that fully answers the query provides the most direct support for subsequent reasoning. A candidate that resolves only part of the query can still identify a promising lead or establish a fact needed for the next search. However, binary positive–negative supervision, as used in Agentic-R's contrastive training (Liu et al., 2026), does not explicitly distinguish complete from partial support. This distinction matters for agentic search: the retriever should prioritize complete support while retaining partial evidence that can guide subsequent reasoning.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

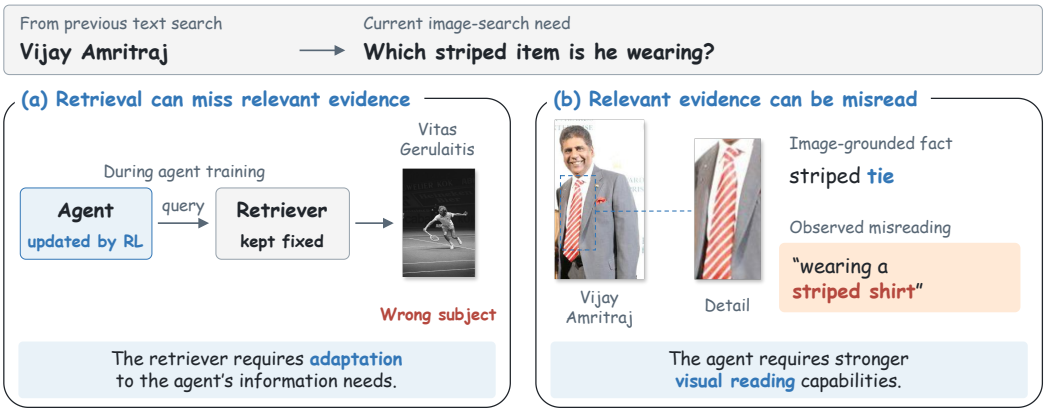


Figure 1: **Retrieval and visual reading bottlenecks in multimodal agentic search.** After identifying Vijay Amritraj through text search, the agent seeks visual evidence to determine which striped item he is wearing. **(a)** The agent is trained with reinforcement learning while the retriever remains fixed; the illustrated search returns an image of another player. **(b)** Even when the relevant image is retrieved, the agent mistakes a striped tie for a striped shirt. These examples motivate adapting retrieval to the agent's information needs and improving visual reading.

Better retrieval alone does not ensure that a multimodal agent can use the evidence it receives. Even when the correct image is retrieved, the agent may misread the visual detail needed to answer the current query. For example, an agent asked to identify a striped item worn by a person may mistake a striped tie for a striped shirt (Figure 1). Unlike retrieved text, visual details and text embedded in images must first be recognized from pixels before they can inform subsequent reasoning. Relevance supervision indicates which images support a query, but does not teach the agent to extract the facts that make them useful. This calls for reading supervision that connects the agent's query to facts grounded in the retrieved image.

We introduce SPAR (**S**tructuring and **P**rojecting Evidence for **A**cquisition and **R**eading), a framework that adapts retrieval and visual reading through shared structured annotations of the agent's search trajectories. For each query, we make its requirements explicit and assess which of them each candidate supports, recording the textual or visual facts underlying these assessments. This analysis establishes both how fully a candidate supports the query and which grounded facts it provides. We derive graded relevance supervision from the support assessments, training the retriever to prioritize complete support while retaining useful partial evidence. For visual reading, we reuse the recorded image-grounded facts as query-conditioned training targets, teaching the agent to extract the information needed for reasoning without a separate fact-annotation process.

Retriever and reading adaptation change both the evidence available to the agent and its ability to extract information from that evidence. We therefore further train the agent on complete search trajectories with the adapted retriever held fixed. This policy adaptation allows the agent to adjust its queries and subsequent reasoning to the evidence returned by the updated retriever, building on the visual reading skills acquired during adaptation.

Experiments on MultiModalQA (Talmor et al., 2021) show that SPAR achieves 76.03% answer accuracy, exceeding the initial agent trained with a fixed retriever by 10.93 percentage points and the strongest baseline by 8.43 points. Ablations show that retriever adaptation improves answering with the agent held fixed. Graded relevance supervision outperforms equally weighted positive supervision derived from the same annotations, both before and after policy adaptation. Reading adaptation increases target-fact accuracy under fixed evidence, and the SPAR agent further improves this accuracy while retaining most responses corrected by reading adaptation. Transfer evaluations on WebQA (Chang et al., 2022), MuSiQue (Trivedi et al., 2022), and HotpotQA (Yang et al., 2018) further assess generalization without additional model training. On HotpotQA, SPAR reaches 61.89% accuracy, exceeding the strongest compared baseline by 3.34 percentage points and supporting transfer from multimodal training to text-only multi-hop question answering.

Our contributions are threefold:

1. We introduce SPAR, a framework that structures the agent's search experience into explicit query requirements, candidate support assessments, and grounded facts, and derives retriever and visual reading supervision from these shared annotations.
2. We develop graded relevance supervision based on explicit analysis of query requirements and candidate evidence. This supervision trains the retriever to prioritize complete support while retaining useful partial evidence for subsequent reasoning.
3. We derive query-conditioned visual reading supervision from the same evidence analysis, using facts grounded in candidate images to train the agent to extract the information needed for reasoning.

2 Related Work

Multimodal agentic search. Multimodal search extends reasoning-guided retrieval (Yao et al., 2023; Trivedi et al., 2023) to text and images (Jiang et al., 2025). In text-based systems, models learn when to retrieve and assess evidence (Asai et al., 2024), or optimize search policies through reinforcement learning (Jin et al., 2025a; Song et al., 2025). Multimodal agents use outcome rewards in MMSearch-R1 (Wu et al., 2026), trajectory supervision followed by reinforcement learning (Geng et al., 2026; Narayan et al., 2025; Yao et al., 2026), or finer-grained supervision: verified reasoning chains in Search-Align (Ning et al., 2026) and process-oriented rewards in ProMMSearchAgent (Yan et al., 2026). For visually rich documents, VRAG-RL and VISOR combine iterative retrieval with visual actions (Wang et al., 2025; Shen et al., 2026). These approaches improve how agents search and use retrieved evidence. SPAR additionally adapts the retriever to the agent's information needs.

Retriever adaptation. Dense retrievers learn from positive–negative supervision (Karpukhin et al., 2020; Qu et al., 2021). Multimodal retrievers support visual question answering (Lin & Byrne, 2022; Lin et al., 2023) and document retrieval (Faysse et al., 2025; Yu et al., 2025). To align retrieval with downstream answering, training signals include reader attention (Izacard & Grave, 2021) and language-model feedback (Izacard et al., 2023; Shi et al., 2024; Lin et al., 2024). LLM-Embedder distills graded utility rewards (Zhang et al., 2024), while SCARLet estimates passage contributions within shared contexts (Xu et al., 2025). For multi-step search, Q-RAG trains the embedder with value-based reinforcement learning (Sorokin et al., 2026). AgentIR (Chen et al., 2026) embeds reasoning traces alongside queries and learns from oracle reranking. Agentic-R (Liu et al., 2026) alternates retriever and agent optimization, selecting passages through local relevance and final-answer correctness. Despite graded relevance scores, its contrastive supervision is binary: one positive passage and sampled negatives. SPAR defines complete and partial support for each executed query through explicit analysis of query requirements and candidate evidence, retaining partial positives while prioritizing complete support. It also reuses grounded visual facts from these annotations to supervise reading.

3 Method

3.1 Overview

SPAR aims to adapt evidence retrieval and visual reading to the information needs that arise during an agent's reasoning (Figure 2). We first construct shared structured annotations of query requirements and candidate evidence from the agent's search trajectories (Section 3.2). From these annotations, the acquisition projection derives graded relevance supervision to train the retriever to prioritize complete support while retaining useful partial support (Section 3.3). The reading projection reuses the grounded visual facts to supervise the agent's extraction of query-relevant information from images (Section 3.4). Finally, we adapt the agent's search policy with the adapted retriever held fixed, allowing it to adjust to the updated retrieval and reading capabilities (Section 3.5).

3.2 Trajectory Evidence Structuring

An agent's search trajectories record the queries it issues to resolve intermediate questions during reasoning. These queries provide a basis for constructing supervision around the agent's actual information needs. We collect trajectories from an initial search agent A_{init}, trained with reinforcement

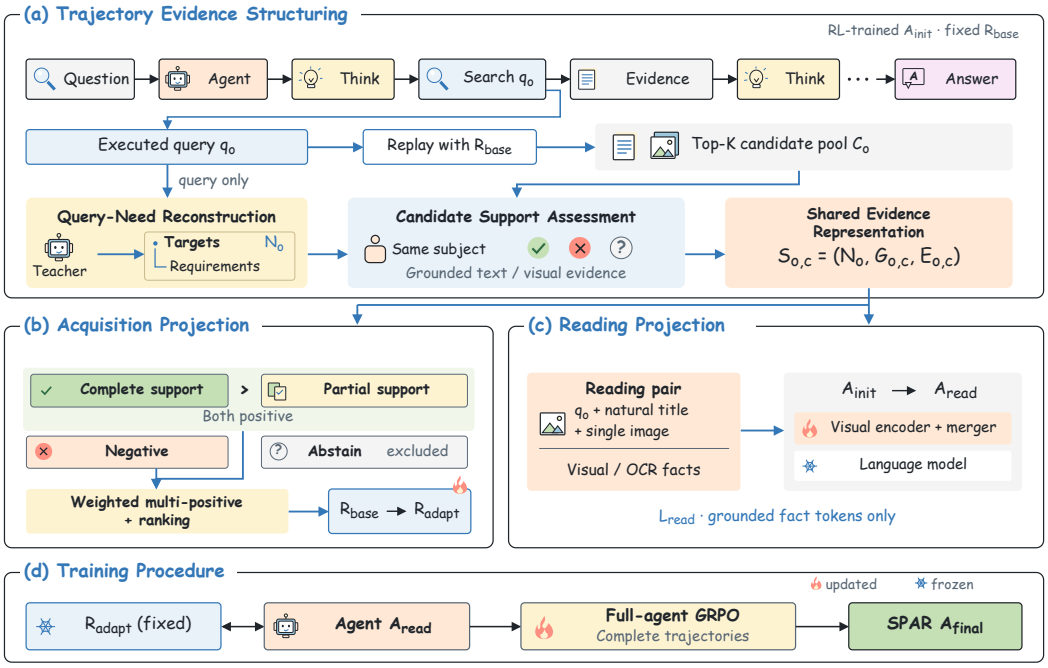


Figure 2: **Overview of SPAR.** Shared structured evidence annotations provide supervision for both retriever adaptation and visual reading. **(a)** Queries from the initial agent's search trajectories are used to construct candidate pools. Query-need reconstruction and candidate support assessment establish query requirements, support judgments, and grounded facts. **(b)** The acquisition projection derives graded relevance supervision, training the retriever to prioritize complete support while retaining partial support as positive evidence. **(c)** The reading projection reuses grounded visual or OCR facts from the same annotations to train the agent's visual encoder and multimodal merger, with the language model frozen. **(d)** Full-agent GRPO then adapts the reading-adapted agent on complete search trajectories with the adapted retriever held fixed.

learning under a fixed pretrained retriever R_{base}, and treat each search step as a distinct occurrence o. For each occurrence, we replay the executed query q_o with the same retriever and retrieve the top-K candidates to form a pool $\mathcal{C}_o$. We then make the query's requirements explicit and assess each candidate against them, recording both its support for those requirements and the underlying textual or visual evidence.

3.2.1 Query-Need Reconstruction

A search query may impose several requirements, while an individual candidate may satisfy only some of them. To assess complete and partial support, we first make these requirements explicit. Given only the executed query q_o, a teacher model identifies the requested targets and specifies the requirements associated with each:

$$N_o = \left[(u_j, [a_{j,1}, \ldots, a_{j,M_j}])\right]_{j=1}^{J_o}, \tag{1}$$

where u_j denotes the j-th target, J_o is the number of targets, and M_j is the number of requirements for target u_j. Each requirement $a_{j,k}$ concerns the target's identity, type, attributes, or relations. These requirements are established independently of the retrieved candidates and serve as the basis for the subsequent evidence assessment. This separation provides a common set of query requirements against which candidates are assessed, making explicit which conditions are supported, contradicted, or unresolved before assigning an overall support label.

3.2.2 CANDIDATE SUPPORT ASSESSMENT

A candidate may describe or depict several entities, so evidence for different query requirements must be associated with the same subject. Given the query, its requirements, and a single candidate, a teacher model identifies the subject corresponding to each requested target and assesses the associated requirements with respect to that subject. Text candidates are presented as a title and passage, and image candidates as a title and image. For each requirement, the teacher determines whether the candidate provides supporting or contradictory evidence, or whether the evidence is missing, uncertain, or conflicting. It also records the textual or visual basis for each assessment. When the subject cannot be identified unambiguously, this uncertainty is retained rather than resolved by assumption.

3.2.3 SHARED EVIDENCE REPRESENTATION

For candidate c at search occurrence o, we organize the resulting annotations into a shared evidence representation:

$$S_{o,c} = \big(N_o, G_{o,c}, E_{o,c}\big), \tag{2}$$

where N_o specifies the query requirements, $G_{o,c}$ records the identified subjects and requirement-level support assessments, and $E_{o,c}$ contains the textual or visual evidence underlying those assessments. The acquisition projection uses these annotations to assign candidate-level relevance labels, while the reading projection reuses the grounded visual or OCR facts as training targets. Both projections therefore draw on the same annotated evidence, without requiring a separate fact-annotation process for visual reading.

3.3 ACQUISITION PROJECTION

Evidence that only partially answers a query can still guide subsequent reasoning. The acquisition projection therefore maps the shared evidence representation $S_{o,c}$ to graded relevance labels, distinguishing complete from partial support for retriever training. A candidate provides COMPLETE SUPPORT if its evidence satisfies all necessary query requirements without a decisive conflict. It provides PARTIAL SUPPORT if it establishes a required or bridging fact that advances the query but is insufficient to resolve it. Topical similarity alone does not qualify as partial support.

Missing information alone does not make a candidate negative, since it may still provide useful partial support. We assign a NEGATIVE label only when the evidence assessment reliably establishes that the candidate offers no meaningful support for the query or decisively contradicts a necessary requirement, such as the requested identity or relation. When ambiguity, conflicting evidence, or limited visibility prevents a reliable judgment, we ABSTAIN and exclude the candidate from retriever supervision for that query.

To learn from both levels of support, we use a weighted multi-positive variant of the contrastive retrieval objective (Karpukhin et al., 2020). Let $\mathcal{D}_o$ and $\mathcal{W}_o$ denote the sets of complete-support and partial-support positives for search occurrence o, respectively:

$$\mathcal{L}_o^{\text{acq}} = -\omega_o \log \frac{\sum_{c \in \mathcal{D}_o \cup \mathcal{W}_o} \alpha_{o,c} \exp(s_\theta(q_o, c))}{\sum_{b \in \mathcal{B}_o} \exp(s_\theta(q_o, b))}. \tag{3}$$

Here, s_θ is the temperature-scaled retrieval score with retriever parameters θ, and $\mathcal{B}_o$ contains the positive and negative candidates used for comparison, excluding query-specific abstentions. The weights $\alpha_{o,c}$ control each positive candidate's contribution, while ω_o controls the contribution of the search occurrence.

Complete-support positives receive unit weight, while partial-support positives receive a smaller weight when both are present. When only partial-support positives are available, we assign them unit weight and downweight the occurrence through ω_o. When both levels of support are present, we additionally apply a ranking loss that encourages the complete-support set to receive higher scores than the partial-support set:

$$\mathcal{L}_o^{\text{rank}} = \text{softplus}(\bar{s}_o(\mathcal{W}_o) - \bar{s}_o(\mathcal{D}_o)), \tag{4}$$

where $\bar{s}_o(\mathcal{P})$ is the log-mean-exp of retrieval scores over the positive set $\mathcal{P}$. The full training objective is detailed in Appendix B.2.

3.4 READING PROJECTION

Even when retrieval succeeds, the agent may fail to extract the visual fact required by the query. The reading projection therefore derives supervision for visual fact extraction from the shared evidence representation $S_{o,c}$. We select images that provide complete or partial support, have an unambiguously identified subject, and contain at least one grounded fact relevant to the query. Eligible facts must be supported by visual content or text within the image; information available only in the title is insufficient.

To align visual reading with the agent's information needs, we pair each eligible image with the query that retrieved it. These pairs are drawn from the candidate pools $\mathcal{C}_o$ used for evidence structuring across all image-search occurrences. Each training input $z_{o,c}$ contains the executed query q_o, a single image c, and its natural title. The query specifies what information to extract, while the title provides context for identifying the subject. The target $y_{o,c}$ consists of the query-relevant visual or OCR fact descriptions already recorded in $E_{o,c}$.

To focus adaptation on visual fact extraction, we compute the language-modeling loss only over the grounded fact tokens in $y_{o,c}$. Let $\mathcal{M}_{o,c}$ denote their positions. The per-example loss and batch objective are

$$\ell_{o,c} = -\frac{1}{|\mathcal{M}_{o,c}|} \sum_{r \in \mathcal{M}_{o,c}} \log p_\phi(y_{o,c,r} \mid z_{o,c}, y_{o,c,<r}), \qquad \mathcal{L}_{\text{read}} = \frac{\sum_{(o,c)\in\mathcal{B}} \rho_{o,c}\ell_{o,c}}{\sum_{(o,c)\in\mathcal{B}} \rho_{o,c}}, \tag{5}$$

where p_ϕ is the agent's conditional distribution, $\mathcal{B}$ denotes a batch of reading examples, and $\rho_{o,c}$ weights each training example. Further details are provided in Appendix B.3.

3.5 TRAINING PROCEDURE

We begin by training a base multimodal language model with reinforcement learning under the fixed pretrained retriever R_{base}. The resulting initial search agent, A_{init}, generates trajectories on training questions for evidence structuring and supervision construction.

We then adapt R_{base} and A_{init} using the acquisition and reading objectives, producing the adapted retriever R_{adapt} and reading-adapted agent A_{read}, respectively. During reading adaptation, we update only the visual encoder and multimodal merger, keeping the language model frozen to limit interference with the agent's previously learned search and reasoning behavior. The reading-adapted agent initializes policy adaptation with the adapted retriever.

The two adaptation stages change both the evidence the agent receives and how it reads visual content. We therefore further train the agent on complete trajectories to adjust its search and reasoning policy to these changes. We keep R_{adapt} fixed and update all parameters of A_{read} using Group Relative Policy Optimization (GRPO; Shao et al., 2024), yielding the final agent A_{final}. Further training details are given in Appendix B.4.

4 EXPERIMENTS

We first evaluate SPAR's answer accuracy on MMQA and its transfer to WebQA, MuSiQue, and HotpotQA. We then examine the contributions of retriever adaptation, graded relevance supervision, and visual reading adaptation through ablations and targeted analyses.

4.1 EXPERIMENTAL SETUP

Benchmarks. We use MultiModalQA (MMQA; Talmor et al., 2021), a multimodal multi-hop question-answering benchmark, for our main experiments. We assess transfer without further model training to the multimodal WebQA benchmark (Chang et al., 2022) and the text-only multi-hop benchmarks MuSiQue-Ans (MuSiQue) (Trivedi et al., 2022) and HotpotQA (Yang et al., 2018). Agents receive only the question and retrieve evidence from fixed corpora; both text-only benchmarks use the same global December 2018 Wikipedia corpus. Dataset splits and corpus construction are detailed in Appendix A.

Table 1: **Answer accuracy across benchmarks (%, ↑).** WebQA, MuSiQue, and HotpotQA assess transfer without further model training. Text, Image, and Both indicate the evidence modalities required for answering. Accuracy is judged by Qwen2.5-72B-Instruct; judging settings are detailed in Appendix A. **Bold** marks the best result in each column. † denotes methods adapted to our evaluation setting (Appendix A).

Method	MMQA				WebQA			MuSiQue	HotpotQA
	Overall	Text	Image	Both	Overall	Text	Image	Overall	Overall
Direct answering	25.03	26.98	28.03	17.93	41.38	39.39	43.33	8.81	27.25
Multimodal RAG	53.22	61.56	54.99	30.05	61.26	66.48	56.15	12.91	44.40
ReAct†	34.33	47.83	15.09	11.25	26.98	46.97	7.45	12.04	36.45
MMSearch-R1†	41.46	51.83	32.35	20.04	30.83	45.34	16.65	11.71	35.71
Search-Align†	41.99	50.10	41.24	21.09	48.29	53.40	43.29	16.59	41.85
Agentic-R†	67.60	74.15	65.77	51.49	56.75	66.64	47.07	20.77	56.69
Initial agent	65.10	73.02	63.88	44.99	56.58	65.91	47.47	26.73	58.56
SPAR	**76.03**	**78.88**	**77.90**	**67.31**	**68.59**	**73.24**	**64.04**	**28.52**	**61.89**

Evaluation. We report answer accuracy judged by Qwen2.5-72B-Instruct (Yang et al., 2024) against reference answers. For MMQA and WebQA, we also report accuracy by evidence modality. Benchmark-specific inference and judging settings, together with HotpotQA answer EM and token F1, are provided in Appendix A.

Baselines. We compare SPAR with direct answering, a multimodal RAG baseline (Lewis et al., 2020), and ReAct (Yao et al., 2023), all using Qwen2.5-VL-7B-Instruct without task-specific training. We also compare against MMSearch-R1 (Wu et al., 2026), Search-Align (Ning et al., 2026), and a multimodal adaptation of Agentic-R (Liu et al., 2026). Our initial agent with the fixed retriever provides a reference for measuring the gains from SPAR's additional training stages. Baseline implementation and adaptation details are provided in Appendix A.

Implementation. We initialize the agent and retriever with Qwen2.5-VL-7B-Instruct (Bai et al., 2025) and Qwen3-VL-Embedding-8B (Li et al., 2026), respectively. We use Qwen2.5-VL-72B-Instruct to construct shared evidence annotations from the initial agent's search trajectories on the MMQA training split. Training configurations and hyperparameters are provided in Appendix B.

4.2 Main Results

Performance on MMQA. SPAR achieves 76.03% answer accuracy on MMQA, outperforming all compared baselines overall and across evidence modalities (Table 1). It improves over Agentic-R, the strongest baseline, by 8.43 percentage points and over the initial agent by 10.93 points. Relative to the initial agent, the largest gains occur on Image and Both questions—14.02 and 22.32 points, respectively—showing improved answering when visual evidence is required.

Transfer across benchmarks. Without further model training, SPAR achieves 61.89% answer accuracy on HotpotQA, exceeding the initial agent, the strongest compared baseline, by 3.34 percentage points (Table 1). It also achieves the highest answer EM and token F1 (Appendix C.4), showing consistent improvements across semantic judging and lexical matching metrics. These results support transfer from multimodal training to text-only multi-hop answering with global-corpus retrieval. SPAR also achieves 68.59% on WebQA and 28.52% on MuSiQue; benchmark-specific inference and judging settings are detailed in Appendix A.

4.3 Ablation Studies

Contribution of retriever adaptation. With the initial agent held fixed, retriever adaptation improves MMQA answer accuracy from 65.10% to 69.56%, a gain of 4.46 percentage points (Table 2). Accuracy improves across all evidence modalities, with the largest gain of 9.32 points on questions requiring both text and images. These results demonstrate the benefit of retriever adaptation for

Table 2: **Effect of the adaptation stages on MMQA.** All variants start from the same initial agent and retriever. Checkmarks indicate which adaptation stages are included. The first and last rows correspond to the initial system and full SPAR, respectively.

Adaptation stages			**Answer accuracy** (%, ↑)			
Retriever	Reading	Policy	Overall	Text	Image	Both
–	–	–	65.10	73.02	63.88	44.99
✓	–	–	69.56	75.88	67.39	54.31
✓	–	✓	73.66	77.35	74.66	63.27
✓	✓	✓	**76.03**	**78.88**	**77.90**	**67.31**

Table 3: **Effect of acquisition supervision on MMQA.** Answer accuracy is reported for the initial agent and after policy adaptation with each retriever held fixed. All variants omit reading adaptation. The lower block compares supervision variants derived from shared structured annotations. **Bold** marks the best result in each column.

	Answer accuracy (%, ↑)	
Retriever / supervision	Initial agent	Adapted agent
Baseline retrievers		
Initial retriever	65.10	66.90
Agentic-R	66.61	67.60
Supervision variants (ours)		
Complete support only	68.05	71.20
Complete + partial, ungraded	68.33	71.61
Complete + partial, graded	**69.56**	**73.66**

downstream answering and suggest that the retrieved evidence better supports the agent's information needs.

Policy adaptation and reading adaptation. With the adapted retriever held fixed, policy adaptation improves MMQA answer accuracy from 69.56% to 73.66% without reading adaptation (Table 2). The full training path, which includes reading adaptation before policy adaptation, reaches 76.03%, exceeding the path without reading adaptation by 2.37 percentage points. The differences are larger on Image and Both questions, at 3.24 and 4.04 points, respectively. This pattern is consistent with reading adaptation helping the agent use visual evidence more effectively during reasoning. Section 4.5 examines visual fact extraction directly under fixed evidence.

4.4 Evidence Acquisition Analysis

Comparison under a fixed agent. With the initial agent held fixed, our adapted retriever achieves 69.56% answer accuracy, exceeding the multimodal adaptation of Agentic-R by 2.95 percentage points (Table 3). We next compare three supervision variants derived from the same structured annotations. *Complete support only* uses only complete-support candidates as positives. *Complete + partial, ungraded* includes both complete- and partial-support positives, assigns equal positive and occurrence weights, and omits the ranking loss. *Complete + partial, graded* applies the weighting and ranking objectives in Section 3.3 to prioritize complete support while retaining partial support as positive evidence.

Effect of graded relevance supervision. Using the same structured annotations, complete-only supervision yields 68.05% answer accuracy. Including partial-support positives with equal positive and occurrence weights gives a modest increase to 68.33%. Graded supervision reaches 69.56%, exceeding the ungraded variant by 1.23 percentage points (Table 3). Since the graded and ungraded variants both retain complete and partial support, this comparison supports the effectiveness of graded weighting and ranking in learning from evidence with different levels of support. Appendix C.1 compares additional weighting and ranking variants under the same fixed-agent setting.

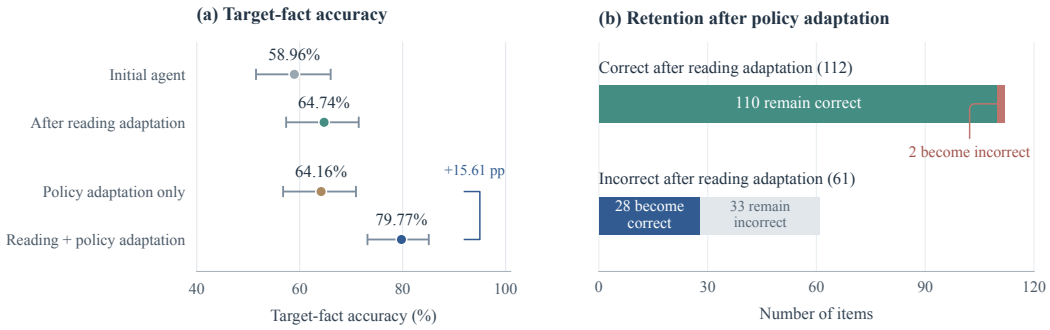


Figure 3: **Visual reading accuracy and retention under fixed evidence.** All four agents receive identical inputs on 173 MMQA examples. **(a)** Target-fact accuracy with 95% Wilson confidence intervals; the bracket compares policy adaptation with and without prior reading adaptation. **(b)** Changes in correctness along the reading-adapted training path, grouped by correctness before policy adaptation. Model configurations, paired comparisons, and scoring are detailed in Appendix C.3.

Comparison after policy adaptation. After policy adaptation with each retriever held fixed, the agent using our graded retriever reaches 73.66% answer accuracy, outperforming the ungraded variant (71.61%) by 2.05 percentage points (Table 3). It also exceeds complete-only supervision (71.20%) and Agentic-R (67.60%). All variants omit reading adaptation. Together with the fixed-agent results, this comparison shows that the advantage of graded supervision persists after the agent adapts its search policy to the retriever.

4.5 Visual Reading Analysis

Reading under fixed evidence. To evaluate visual reading under fixed evidence, we select 173 MMQA examples whose images were not used during reading adaptation. Each example requires a fact that is visible in the image but not directly supplied by the query or title; selection does not use model answers or correctness scores. All four agents receive identical queries, image titles, and images. We manually verify each response against predefined, image-grounded reference facts and report target-fact accuracy, counting abstentions and failures as incorrect. Scoring details are provided in Appendix C.3.

Reading adaptation increases target-fact accuracy from 58.96% to 64.74%, with 17 incorrect responses becoming correct and seven correct responses becoming incorrect. The SPAR agent, trained with reading adaptation followed by policy adaptation, reaches 79.77%, compared with 64.16% for the policy-only agent, a difference of 15.61 percentage points (Figure 3(a)). These results show that the SPAR agent extracts query-relevant visual facts more accurately under fixed evidence.

The SPAR agent retains 110 of the 112 correct responses from the reading-adapted agent, including 16 of the 17 examples corrected by reading adaptation. It also answers 28 previously incorrect examples correctly, with two regressions (Figure 3(b)). Thus, SPAR improves target-fact accuracy while preserving most of the responses corrected by reading adaptation. Appendix C.3 provides detailed paired comparisons, and Appendix D presents complementary case studies of evidence acquisition and visual reading.

5 Conclusion

We introduced SPAR, which derives retriever and visual reading supervision from shared structured annotations of an agent's search trajectories. Graded relevance supervision distinguishes complete from partial support, while query-conditioned reading supervision teaches the agent to extract grounded visual facts. Subsequent policy adaptation integrates the updated retrieval and reading capabilities into the search process. Experiments show improved answering on MMQA and transfer to WebQA, MuSiQue, and HotpotQA without further model training. These findings highlight the value of adapting both evidence acquisition and visual reading to the agent's information needs.

Reproducibility Statement

Section 3 describes SPAR's evidence annotation procedure, supervision construction, and training objectives. Appendix A documents the evaluation datasets and retrieval corpora, baseline adaptations, inference and scoring settings, and evaluated model checkpoints. Appendix B describes the annotation and training procedures and reports training configurations. The construction and scoring of the fixed-evidence visual reading diagnostic are described in Appendix C.3. We will publicly release our code upon acceptance.

AI Use Statement

Generative AI tools assisted with language polishing, LaTeX formatting, organizing author-developed outlines, and identifying related literature. The authors determined the research ideas, experimental designs, manuscript structure, and substantive content. AI tools also assisted with selected coding and experiment-execution tasks under detailed author instructions, including implementation specifications. The authors manually reviewed the AI-assisted code. Teacher models generated structured evidence annotations and grounded reading targets for training (Section 3), and model-based judges were used for evaluation (Appendix A). The authors take responsibility for the final manuscript, methods, results, and associated artifacts.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025. URL https://arxiv.org/abs/2502.13923.

Yingshan Chang, Mridu Narang, Hisami Suzuki, Guihong Cao, Jianfeng Gao, and Yonatan Bisk. WebQA: Multihop and multimodal qa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16495–16504, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Chang_WebQA_Multihop_and_Multimodal_QA_CVPR_2022_paper.html.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Jimmy Lin, Akari Asai, and Victor Zhong. AgentIR: Reasoning-aware retrieval for deep research agents. *arXiv preprint arXiv:2603.04384*, 2026. URL https://arxiv.org/abs/2603.04384.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. ColPali: Efficient document retrieval with vision language models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/99e9e141aafc314f76b0ca3dd66898b3-Abstract-Conference.html.

Xinyu Geng, Peng Xia, Zhen Zhang, Xinyu Wang, Qiuchen Wang, Ruixue Ding, Chenxi Wang, Jialong Wu, Kuan Li, Yida Zhao, Huifeng Yin, Yong Jiang, Pengjun Xie, Fei Huang, Huaxiu Yao, Yi R. Fung, and Jingren Zhou. WebWatcher: Breaking new frontiers of vision-language deep research agent. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=8jsaazdAb3.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Gautier Izacard and Edouard Grave. Distilling knowledge from reader to retriever for question answering. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=NTEz-6wysdb.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251): 1–43, 2023. URL https://jmlr.org/papers/v24/23-0037.html.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanmin Wu, Jiayi Lei, Pengshuo Qiu, Pan Lu, Zehui Chen, Guanglu Song, Peng Gao, Yu Liu, Chunyuan Li, and Hongsheng Li. MMSearch: Unveiling the potential of large models as multi-modal search engines. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/04e6525463afaa460b5a2e6868b18f18-Abstract-Conference.html.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Ö. Arık, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to reason and leverage search engines with reinforcement learning. In *Conference on Language Modeling*, 2025a. URL https://openreview.net/forum?id=Rwhi91ideu.

Jiajie Jin, Yutao Zhu, Zhicheng Dou, Guanting Dong, Xinyu Yang, Chenghao Zhang, Tong Zhao, Zhao Yang, and Ji-Rong Wen. FlashRAG: A modular toolkit for efficient retrieval-augmented generation research. In *Companion Proceedings of the ACM on Web Conference 2025*, pp. 737–740. ACM, 2025b. doi: 10.1145/3701716.3715313. URL https://doi.org/10.1145/3701716.3715313.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Qwen3-VL-Embedding and Qwen3-VL-Reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*, 2026. URL https://arxiv.org/abs/2601.04720.

Weizhe Lin and Bill Byrne. Retrieval augmented visual question answering with outside knowledge. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11238–11254. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.772. URL https://aclanthology.org/2022.emnlp-main.772/.

Weizhe Lin, Jinghong Chen, Jingbiao Mei, Alexandru Coca, and Bill Byrne. Fine-grained late-interaction multi-modal retrieval for retrieval augmented visual question answering. In *Advances in Neural Information Processing Systems*, volume 36, pp. 22820–22840, 2023. URL https://papers.nips.cc/paper_files/paper/2023/hash/47393e8594c82ce8fd83adc672cf9872-Abstract-Conference.html.

Xi Victoria Lin, Xilun Chen, Mingda Chen, Weijia Shi, Maria Lomeli, Richard James, Pedro Rodriguez, Jacob Kahn, Gergely Szilvasy, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. RA-DIT: Retrieval-augmented dual instruction tuning. In *International Conference on Learning Representations*. OpenReview.net, 2024. URL https://openreview.net/forum?id=22OTbutug9.

Wenhan Liu, Xinyu Ma, Yutao Zhu, Yuchen Li, Daiting Shi, Dawei Yin, and Zhicheng Dou. Agentic-R: Learning to retrieve for agentic search. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 15987–16005. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.785. URL https://aclanthology.org/2026.findings-acl.785/.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

Kartik Narayan, Yang Xu, Tian Cao, Kavya Nerella, Vishal M. Patel, Navid Shiee, Peter Grasch, Chao Jia, Yinfei Yang, and Zhe Gan. DeepMMSearch-R1: Empowering multimodal LLMs in multimodal web search. *arXiv preprint arXiv:2510.12801*, 2025. URL https://arxiv.org/abs/2510.12801.

Xuying Ning, Dongqi Fu, Tianxin Wei, Mengting Ai, Jiaru Zou, Ting-Wei Li, Hanghang Tong, Yada Zhu, Hendrik Hamann, and Jingrui He. MC-Search: Evaluating and enhancing multimodal agentic search with structured long reasoning chains. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=JEGDp1E4OH.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5835–5847. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.naacl-main.466. URL https://aclanthology.org/2021.naacl-main.466/.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Yucheng Shen, Jiulong Wu, Jizhou Huang, Dawei Yin, Lingyong Yan, and Min Cao. VISOR: Agentic visual retrieval-augmented generation via iterative search and over-horizon reasoning. *arXiv preprint arXiv:2604.09508*, 2026. URL https://arxiv.org/abs/2604.09508.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. REPLUG: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8371–8384. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.463. URL https://aclanthology.org/2024.naacl-long.463/.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-Searcher: Incentivizing the search capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025. URL https://arxiv.org/abs/2503.05592.

Artyom Sorokin, Nazar Buzun, Alexander Anokhin, Egor Vedernikov, Petr Anokhin, Mikhail Burtsev, and Evgeny Burnaev. Q-RAG: Long context multi-step retrieval via value-based embedder training. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=MS9nWFY7LG.

Alon Talmor, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannaneh Hajishirzi, and Jonathan Berant. MultiModalQA: Complex question answering over text, tables and images. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=ee6W5UgQLa.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10014–10037. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.557. URL https://aclanthology.org/2023.acl-long.557/.

Qiuchen Wang, Ruixue Ding, Yu Zeng, Zehui Chen, Lin Chen, Shihang Wang, Pengjun Xie, Fei Huang, and Feng Zhao. VRAG-RL: Empower vision-perception-based RAG for visually rich information understanding via iterative reasoning with reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1915. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/529b876b5d87f997e98ed925a3ce9f01-Abstract-Conference.html.

Jinming Wu, Zihao Deng, Wei Li, Yiding Liu, Bo You, Bo Li, Zejun Ma, and Ziwei Liu. MMSearch-R1: Incentivizing LMMs to search. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2456–2487. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.114. URL https://aclanthology.org/2026.acl-long.114/.

Yilong Xu, Jinhua Gao, Xiaoming Yu, Yuanhai Xue, Baolong Bi, Huawei Shen, and Xueqi Cheng. Training a utility-based retriever through shared context attribution for retrieval-augmented language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 629–648. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.33. URL https://aclanthology.org/2025.emnlp-main.33/.

Wentao Yan, Shengqin Wang, Huichi Zhou, Yihang Chen, Kun Shao, Yuan Xie, and Zhizhong Zhang. ProMMSearchAgent: A generalizable multimodal search agent trained with process-oriented rewards. *arXiv preprint arXiv:2604.20486*, 2026. URL https://arxiv.org/abs/2604.20486.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Huanjin Yao, Qixiang Yin, Min Yang, Ziwang Zhao, Yibo Wang, Haotian Luo, Jingyi Zhang, and Jiaxing Huang. MM-DeepResearch: A simple and effective multimodal agentic search baseline. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=XZAOCytvKQ.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*. OpenReview.net, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Shi Yu, Chaoyue Tang, Bokai Xu, Junbo Cui, Junhao Ran, Yukun Yan, Zhenghao Liu, Shuo Wang, Xu Han, Zhiyuan Liu, and Maosong Sun. VisRAG: Vision-based retrieval-augmented generation on multi-modality documents. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/3640a1997a4c9571cea9db2c82e1fc35-Abstract-Conference.html.

Peitian Zhang, Zheng Liu, Shitao Xiao, Zhicheng Dou, and Jian-Yun Nie. A multi-task embedder for retrieval augmented LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3537–3553. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.194. URL https://aclanthology.org/2024.acl-long.194/.

A Evaluation Protocols

A.1 Benchmarks and Answer Evaluation

Benchmarks and evidence corpora. Table 4 summarizes the evaluation splits and retrieval corpora used in our experiments. MMQA includes 1,501 Text, 371 Image, and 569 Both questions; table evidence is included in Text. WebQA contains 2,455 text and 2,511 image questions. For MuSiQue and HotpotQA, we use the development questions and answer aliases distributed with FlashRAG (Jin et al., 2025b). Retrieval searches the complete Wikipedia corpus, rather than the question-specific candidate paragraphs, and questions are not filtered by reference-evidence coverage. This setting evaluates multi-hop answering when the agent must first locate the relevant evidence.

Table 4: **Evaluation datasets and evidence corpora.** All questions in each listed split are retained. Corpus sizes count indexed text items and images.

Dataset	Split	Questions	Retrieval corpus
MMQA	Development	2,441	228,327 text items; 57,058 images
WebQA	Validation	4,966	787,697 text items; 389,750 images
MuSiQue	Development	2,417	21,015,324 Wikipedia passages
HotpotQA	Development	7,405	21,015,324 Wikipedia passages

Answer evaluation. We measure whether the generated answer correctly resolves the question, allowing semantically equivalent answers. Qwen2.5-72B-Instruct receives the question, reference answer aliases, and extracted final answer, without the method identity or search trajectory. Empty answers and generation failures count as incorrect and remain in the full dataset denominator. Reference answers, support labels, and modality labels are withheld from generation and retrieval.

The judging template is reproduced in Prompt P5 (Appendix F). The judge uses temperature 0.01 and a 64-token output limit. Reference aliases are joined by semicolons; a verdict is positive only if it contains the standalone word `Yes` and no standalone `No`, ignoring case.

Complementary answer metrics. For HotpotQA, all eight methods use the same judging prompt and decoding configuration. We additionally report answer exact match (EM) and token F1, using each method's native final answer. Scoring lowercases answers, removes punctuation and articles, and normalizes whitespace, then takes the maximum score over the reference answers for each question. Incorrect categorical answers receive zero token F1. Results are reported in Appendix C.4.

Transfer without further training. SPAR transfer uses the MMQA-trained model weights without target-benchmark training. WebQA retrieval uses its fixed text and image corpora. MuSiQue and HotpotQA retrieval use the same December 2018 Wikipedia corpus (Karpukhin et al., 2020), which differs from the official HotpotQA FullWiki corpus. Each retriever indexes the same evidence collection. These two benchmarks therefore test the transfer of retrieval and search behavior to text-only multi-hop questions.

A.2 Baseline Adaptations

We adapt the compared methods to the fixed evidence corpora while retaining the principal components of their released inference procedures. Table 5 summarizes the generation and retrieval budgets. These are complete-system comparisons: the methods differ in controller behavior and inference compute as well as in training.

Direct answering, multimodal RAG, and ReAct. These baselines use Qwen2.5-VL-7B-Instruct without task-specific training and use greedy generation. Multimodal RAG retrieves from the initial Qwen3-VL-Embedding-8B text and image-title indexes, with a 512^2 maximum-pixel budget per image. ReAct (Yao et al., 2023) retains its six demonstrations and released controller; Search and Lookup operate on corpus text, and ImageSearch returns real images from the initial retriever. The ReAct result thus evaluates a multimodal fixed-corpus adaptation of the released controller.

Table 5: **Inference configurations on MMQA.** Text/image entries give the number of results per corresponding search. Call limits apply to each generation; sequence thresholds count assistant outputs and inserted tool observations across turns. Controller limits retain their native meanings.

Method	Text / image	Token setting	Controller limit
Direct answering	– / –	512 / call	One generation
Multimodal RAG	3 / 3	512 / call	One generation
ReAct	3 / 3	100 / call	Seven steps; four searches
MMSearch-R1	3 / 3	512 / call	Three generations
Search-Align	1 / 1	128 / call	Nine generations
Initial agent	3 / 3	4,096 / sequence	Five assistant replies
Agentic-R	3 / 3	4,096 / sequence	Five assistant replies
SPAR	3 / 3	4,096 / sequence	Five assistant replies

MMSearch-R1. We evaluate the released MMSearch-R1-7B checkpoint (Wu et al., 2026) without MMQA fine-tuning, using its public inference controller and question-only inputs. Text search uses the generated query. Because the released image-search interface has no query argument, image retrieval uses the original question. We retain the prompts, parsing, history, and greedy generation, and append an instruction describing the corpus interface. Retrieved images use the controller's 672^2 maximum-pixel budget. The final prediction is the last complete answer-tag span.

Search-Align. We reconstruct the Search-Align process-supervised training stage (Ning et al., 2026) from its released questions and intermediate answers. Qwen2.5-VL-72B-Instruct supplies missing reasoning text, yielding 3,263 eligible conversations from 3,333 released questions. We train Qwen2.5-VL-7B-Instruct for two epochs using AdamW (Loshchilov & Hutter, 2019), learning rate 10^{-4}, global batch size 32, cosine decay, and 0.1 warmup. Reconstruction choices include LoRA (Hu et al., 2022) rank 8 with scaling 16, a frozen visual encoder and projector, a 32,768-token cutoff, and a 768^2 image budget. The final second-epoch checkpoint is used without MMQA-based selection.

Evaluation retains the released controller, prompt, history, final-answer parser, and generation function. Corpus retrieval replaces the original search backends, returning one text item or image with the native zero score threshold and missing-result behavior. The exported sampling configuration uses temperature 10^{-6} and repetition penalty 1.05. The common evaluator receives the controller's native final answer.

Agentic-R. We use one iteration of the released local/global utility construction and training-data selection procedure (Liu et al., 2026), retaining its exact-match global utility, selection logic, missing-data handling, and negative sampling. The resulting set contains 21,386 retriever-training examples. Qwen3-VL-Embedding-8B replaces the text-only retriever and consumes real image inputs. Training uses two epochs, learning rate 2×10^{-5}, global query batch size 256, 16 passages per query, contrastive temperature 0.01, and 0.1 warmup. LoRA uses rank 8 and scaling 16 with the visual encoder frozen. Query/passage token limits are 512/1,024, and images use at most 768^2 pixels. We use the final retriever checkpoint.

For the reported comparison, this retriever is paired with our multimodal tool-agent controller. To reproduce Agentic-R's original query formulation, we concatenate the original question and executed query as *Question [SEP] query*. Policy adaptation keeps the retriever fixed and omits reading adaptation. Training and checkpoint selection follow the common protocol in Appendix A.3.

A.3 Model Selection and Inference

All second-stage agent variants originate from the same initial agent. During policy adaptation, we train each variant for one epoch on the MMQA training set, evaluate every 20 optimization steps, and select the checkpoint with the highest MMQA development answer accuracy. SPAR applies reading adaptation before policy adaptation; the variants without reading adaptation proceed directly from the initial agent. Transfer evaluation uses the selected MMQA checkpoint without further training or checkpoint selection.

The initial agent, Agentic-R, and SPAR use greedy decoding, a 4,096-token multi-turn response-length threshold, at most five assistant replies, top-three retrieval per modality, and at most 512^2 pixels per returned image. Table 5 specifies the token accounting and controller limits.

B Training Details

B.1 Structured Evidence Annotation

The annotation process separates the query's requirements from the evidence available in each candidate. Qwen2.5-VL-72B-Instruct performs query-need reconstruction using only the executed query. Candidate support assessment then receives the reconstructed requirements and one candidate: a title and passage for text, or a title and image for visual evidence. Neither stage receives the reference answer or subsequent trajectory outcome. Associating requirement-level evidence with an identified subject prevents facts about different entities from being combined into a single support judgment.

The acquisition projection maps these assessments to COMPLETE SUPPORT, PARTIAL SUPPORT, NEGATIVE, or ABSTAIN, following Section 3.3. Partial support requires a useful fact for the current query; topical similarity alone is insufficient. The reading projection draws on the visual or OCR facts already recorded in the shared annotations, avoiding a separate fact-annotation stage.

A worked example and the key prompts are provided in Appendices E and F, respectively.

B.2 Acquisition Objective and Sampling

Candidate selection. For each search occurrence, the annotation pool contains the top 20 same-modality candidates from the initial retriever. Training retains occurrences with at least one resolved positive and three explicit negatives. Explicit negatives follow the initial retrieval order. Document deduplication and in-batch negatives are enabled; known query-specific positives retain their positive status, and abstentions are masked from that query's comparison set.

Relative weights. The acquisition objective preserves useful partial support while giving complete support greater weight. In Equation 3, complete-support positives receive $\alpha_{o,c} = 1$. Partial-support positives receive $\alpha_{o,c} = \lambda$ when complete support is present and unit weight otherwise. Occurrences with complete support have $\omega_o = 1$; those with only partial support have $\omega_o = \lambda$. We set $\lambda = 0.5$. Weighting partial-only occurrences reduces their gradient contribution while preserving their candidates as positives.

Ordering the support levels. When both positive groups are present, the ranking term in Equation 4 compares their log-mean-exp scores,

$$\bar{s}_o(\mathcal{P}) = \log\left(\frac{1}{|\mathcal{P}|}\sum_{c\in\mathcal{P}}\exp(s_\theta(q_o, c))\right). \tag{6}$$

Let $\mathcal{O}$ be the training occurrences in a batch and $\mathcal{O}_{\text{both}}$ those containing both complete- and partial-support positives. The full objective is

$$\mathcal{L}_{\text{retriever}} = \frac{1}{|\mathcal{O}|}\sum_{o\in\mathcal{O}}\mathcal{L}_o^{\text{acq}} + \frac{\beta}{|\mathcal{O}|}\sum_{o\in\mathcal{O}_{\text{both}}}\text{softplus}(\bar{s}_o(\mathcal{W}_o) - \bar{s}_o(\mathcal{D}_o))\,. \tag{7}$$

We use $\beta = 1$ and normalize both terms by the full number of training occurrences.

Supervision variants. The ungraded variant includes complete- and partial-support positives with unit positive and occurrence weights and omits the ranking term. The complete-only variant removes partial-support positives and excludes occurrences with no complete support. Table 3 compares these supervision variants.

Training scale and optimization. The final SPAR retriever training set contains 30,547 search occurrences: 19,928 text-search and 10,619 image-search occurrences. We adapt Qwen3-VL-Embedding-8B with LoRA of rank 8 and scaling 16 on the language model's attention and feed-forward projections, keeping the base weights frozen. Training uses AdamW, a learning rate of

8×10^{-5} with linear decay and 5% warmup, a global query batch size of 64, and one epoch (478 updates). The contrastive temperature is 0.02. Query and passage lengths are capped at 512 and 1,024 tokens, respectively, and images use at most 768^2 pixels. We use the final retriever checkpoint.

B.3 Reading Supervision

Coverage of the shared annotations. Reading supervision draws from the image candidates annotated for acquisition. We consider all image-search occurrences and candidates at ranks 1–20, applying the eligibility criteria in Section 3.4. Exact duplicate question–query–image–target examples are removed. Table 6 summarizes the final training set.

Table 6: **Reading supervision statistics.** Counts describe the final single-image training set.

Training examples	Questions	Search occurrences	Unique images
26,048	7,162	10,449	10,208

Inputs and targets. Each input consists of the executed query, natural image title, and a single image, formatted with the initial agent's chat template and a short reading instruction. The target contains at most two grounded visual or OCR fact descriptions from the shared annotations. Loss is applied only to tokens fully contained in the fact spans; all other tokens are masked.

Balancing questions and search occurrences. Questions contribute different numbers of search occurrences and eligible images. We assign equal total weight to each question, distributing it uniformly across its represented occurrences and then across their retained candidates. If question d has n_d represented occurrences and occurrence o has k_o retained candidates, the sample weight in Equation 5 is

$$\rho_{o,c} = \frac{1}{n_d k_o}. \tag{8}$$

Image processing. Reading adaptation uses the agent's 512^2 maximum-pixel budget per image.

B.4 Optimization and Policy Adaptation

Reading adaptation. Reading adaptation starts from the initial agent and updates only the visual encoder and multimodal merger, keeping the language model frozen. Table 7 summarizes the optimization settings.

Table 7: **Optimization settings for reading adaptation.**

Setting	Value
Optimizer / learning rate	AdamW / 2×10^{-6}
Adam coefficients / weight decay	$(0.9, 0.95)$ / 0.01
Schedule / warmup fraction	Cosine / 0.03
Global batch size / training duration	64 / one epoch
Gradient clipping	1.0

Policy adaptation. Policy adaptation initializes from the reading-adapted agent and updates all agent parameters with GRPO, keeping the adapted retriever and its indexes fixed. Question batches approximately match the modality proportions of the MMQA development split. Training duration and checkpoint selection follow Appendix A.3.

Agent optimization. Both RL stages use group-normalized GRPO advantages and KL regularization against a frozen reference policy. Table 8 summarizes the principal optimization settings.

Table 8: **Principal agent optimization settings.** Both RL stages update all agent parameters.

Setting	Value
Optimizer / learning rate	AdamW / 10^{-6}
Schedule / warmup fraction	Constant after warmup / 0.05
Questions per batch / rollouts per question	128 / 8
Rollout temperature / top-p	1.0 / 1.0
Policy minibatch / passes per rollout batch	256 trajectories / 1
Reference KL coefficient	10^{-3}

Agent rewards. Both RL stages combine answer quality, reference-evidence retrieval, tool use, and answer formatting (Table 9). Answer quality multiplies normalized token recall against reference answers by normalized BART fluency. Evidence coverage and tool use are averaged over the required modalities; an evidence hit requires retrieving at least one reference item during the rollout. Joint success requires hits in all required modalities and answer quality above 0.5.

Table 9: **Task-reward components and weights.**

Component	Initial agent	Policy adaptation
Answer quality	0.45	0.40
Reference-evidence coverage	0.20	0.20
Evidence hits in all required modalities	0.15	0.20
Required search tools invoked	0.10	0.10
Joint answer–evidence success	0.10	0.05
Answer-format bonus	0.05	0.05

Initial-agent training matches reference evidence against all retrieved IDs. During policy adaptation, hits must come from the corresponding modality-specific tool; image-dependent questions receive answer-quality credit only when all required modalities have evidence hits. Reference answers and evidence IDs are used only for reward computation. Penalties discourage excess search rounds and overly long responses, and rewards are bounded to $[-1, 1]$.

C Supplementary Analyses

C.1 Acquisition Objective Variants

We compare acquisition objective variants with different weighting and ranking choices. All four variants use the same training occurrences and complete- and partial-support positives, with the same number of optimization steps. Each retriever is evaluated with the initial agent fixed, without reading or policy adaptation. Ungraded supervision assigns unit positive and occurrence weights; graded supervision uses the weights in Appendix B.2. We compare our set-level ranking term with a CoSENT-style pairwise alternative, which applies softplus to the log-mean-exp of partial-minus-complete score differences over all complete–partial pairs within each occurrence.

Table 10: **Acquisition objective variants.** MMQA answer accuracy (%, ↑) with the initial agent fixed. All variants retain the same complete- and partial-support positives.

Weighting	Ranking	Accuracy
Ungraded	None	68.33
Graded	None	68.91
Graded	Pairwise (CoSENT-style)	68.46
Graded	Set-level (ours)	**69.56**

Without a ranking term, ungraded and graded weighting yield 68.33% and 68.91% accuracy, respectively. Graded weighting with pairwise ranking yields 68.46%. Combining graded weighting with set-level ranking achieves the highest observed accuracy among the evaluated variants, at 69.56% (Table 10).

C.2 Sensitivity to the Explicit Negative Cap

We examine how the number of explicitly labeled negatives affects retriever adaptation. We vary the cap over $\{1, 3, 6, 12, 15\}$ while keeping the query cohort, positive sets, optimizer steps, and initial agent fixed. Explicit negatives follow retrieval order, and in-batch negatives remain enabled. Among the evaluated settings, a cap of three achieves the highest MMQA answer accuracy (69.56%; Figure 4), supporting the setting used in our main experiments.

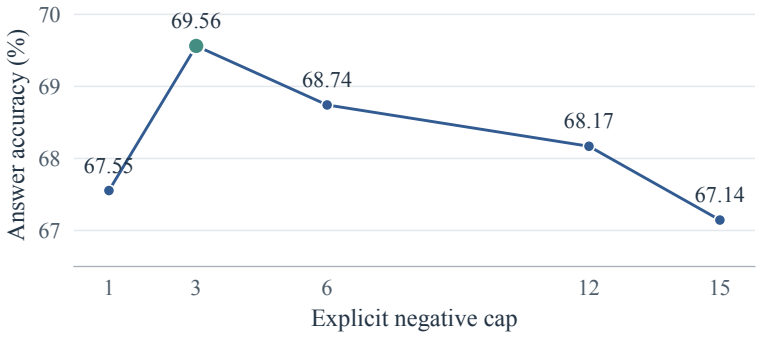


Figure 4: **Effect of the explicit negative cap on MMQA.** Each retriever is evaluated with the same initial agent.

C.3 Visual Reading under Fixed Evidence

Sample construction. We evaluate visual fact extraction on 173 MMQA development examples whose images were not used during reading adaptation. Each example requires a fact that is visible in the image and not supplied by the query or title.

Inputs and scoring. The initial, reading-adapted, policy-only, and SPAR agents receive identical query, title, and image inputs, using a 512^2 maximum-pixel budget, greedy decoding, and a 128-token output limit without retrieval. Reference facts are specified before inference. Reference facts and correctness judgments for all four agents have been manually verified. An answer receives credit when it states the required visual fact correctly; additional non-target claims are not scored. Abstentions and failures count as incorrect. The policy-only and SPAR agents follow the common checkpoint-selection protocol in Appendix A.3.

Paired comparison. SPAR achieves the highest target-fact accuracy among the four agents on the 173 examples (Figure 3). Relative to the policy-only agent, it corrects 30 responses and regresses three, a gain of 15.61 percentage points.

C.4 HotpotQA Results under Global Retrieval

Table 11 complements the judged-accuracy results with answer EM and token F1 under the same global-retrieval protocol (Appendix A). SPAR achieves the highest score on both metrics, exceeding the initial agent by 2.15 EM points and 3.13 F1 points. This agrees with the judged-accuracy comparison in Table 1.

Table 11: **HotpotQA answer EM and token F1 (%, ↑).** Evaluation uses all 7,405 development questions under global Wikipedia retrieval. † denotes adapted methods (Appendix A.2).

Method	EM	F1
Direct answering	6.78	14.74
Multimodal RAG	16.76	27.00
ReAct†	23.51	31.53
MMSearch-R1†	24.85	31.99
Search-Align†	8.06	21.68
Agentic-R†	40.65	51.98
Initial agent	42.31	53.63
SPAR	**44.46**	**56.76**

D CASE STUDIES

Acquiring evidence that advances reasoning. Partial support provides intermediate facts that guide the agent's next search (Figure 5). In the Cannes example, the festival list identifies candidate films and directors but leaves the poster condition unresolved. With this list ranked first by the graded retriever, the agent searches for the poster of *The BFG* and verifies the visual condition before answering. In the text-only example, a passage on *The Fairy-Queen* identifies its libretto as an adaptation of *A Midsummer Night's Dream* but leaves the query's 2005 production condition unresolved. A subsequent query using *The Fairy-Queen* retrieves evidence of Armonico Consort's 2005 production.



Figure 5: **Partial support that guides further search.** Both cases compare binary and graded relevance supervision with the same initial agent, without reading or policy adaptation. **Top:** for the same initial query, the Cannes list moves from rank three to rank one, providing the film title used in the subsequent image search. **Bottom:** selected steps from two trajectories with different executed queries. The graded trajectory uses *The Fairy-Queen* to verify the remaining production condition; the binary trajectory revisits the source play without producing a final answer. Questions and passages are condensed for readability; the ellipsis in the lower case abbreviates the executed query.

Reading the required visual fact. Reading adaptation corrects errors in interpreting the same retrieved images (Figure 6). In the hat example, the initial agent answers *front*, whereas the reading-adapted agent identifies the brim behind the head and answers *back*. In the *Reservoir Dogs* example, the initial agent counts four people, while the reading-adapted agent correctly counts five. Both facts depend on visual content and cannot be determined from the image titles alone.

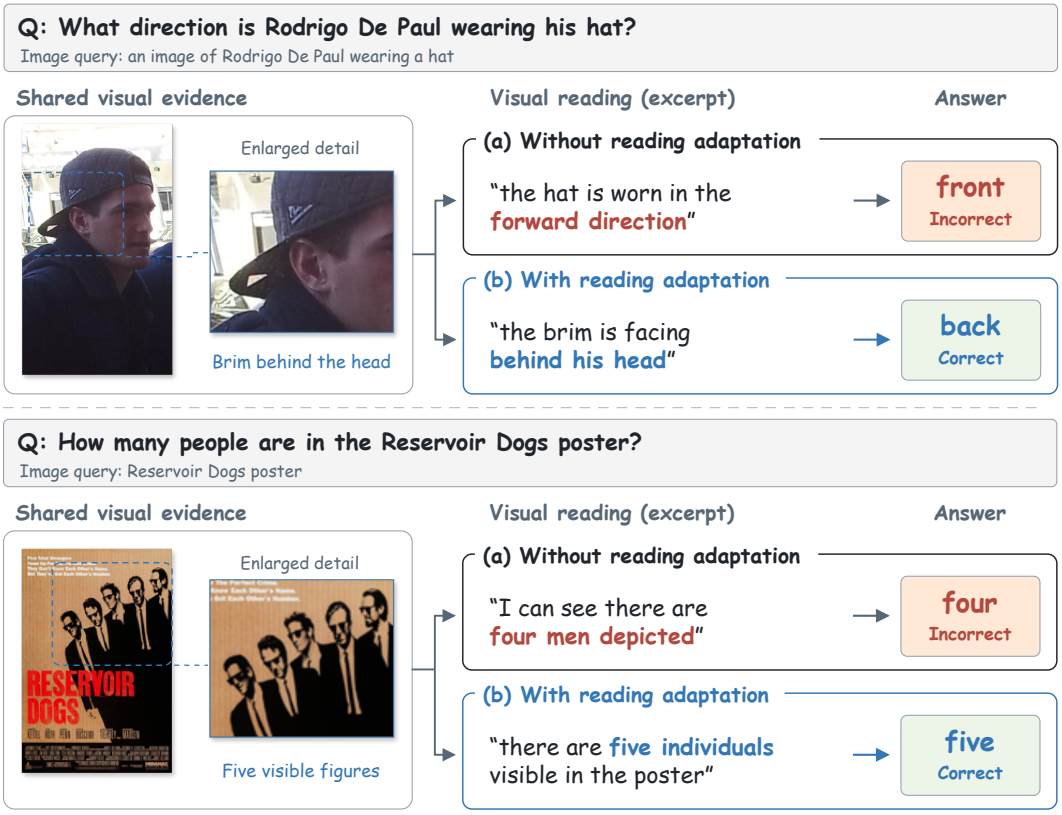


Figure 6: **Correcting visual errors under shared evidence.** Each case compares the initial and reading-adapted agents with the same retriever, query, and three returned images, including their titles and order, without policy adaptation. The rank-one image is shown; enlarged details are for illustration only. Response excerpts show the correction of a hat-orientation error (top) and a counting error (bottom).

Dataset ambiguity in failure cases. Ambiguity in a dataset question can cause a visually supported answer to be scored as incorrect (Figure 7). The question asks how many people are playing a board game but does not identify the intended scene. The reference answer, *four*, is tied to the dataset's target image, while another retrieved image shows three players at the foreground table. The agent answers *three*, which is supported by that image but is marked incorrect by the evaluator. Under question-only global retrieval, this example exposes missing image-specific context in the dataset question: the reference selects a particular scene that the question does not distinguish from other relevant scenes.



Figure 7: **A failure case with an underspecified dataset question.** Both images match the question's description and occur in the same search result. The answer *three* is supported by the foreground players in *Shadow Hunters*, but is scored as incorrect because the reference answer *four* corresponds to the dataset target, *Board game*. The illustrated trajectory comes from a separate diagnostic checkpoint.

E ANNOTATION EXAMPLE

Figure 8 illustrates how the same structured evidence annotations support acquisition and reading adaptation. The executed query requests a ferris wheel and a clock tower in London. The *London* montage supports both targets, whereas *South Bank* supports the ferris wheel but leaves the clock tower unresolved. The acquisition projection retains both candidates as positives and assigns greater weight to complete support. The reading projection reuses a visual fact recorded for *South Bank* as a training target, without a separate fact-annotation step.

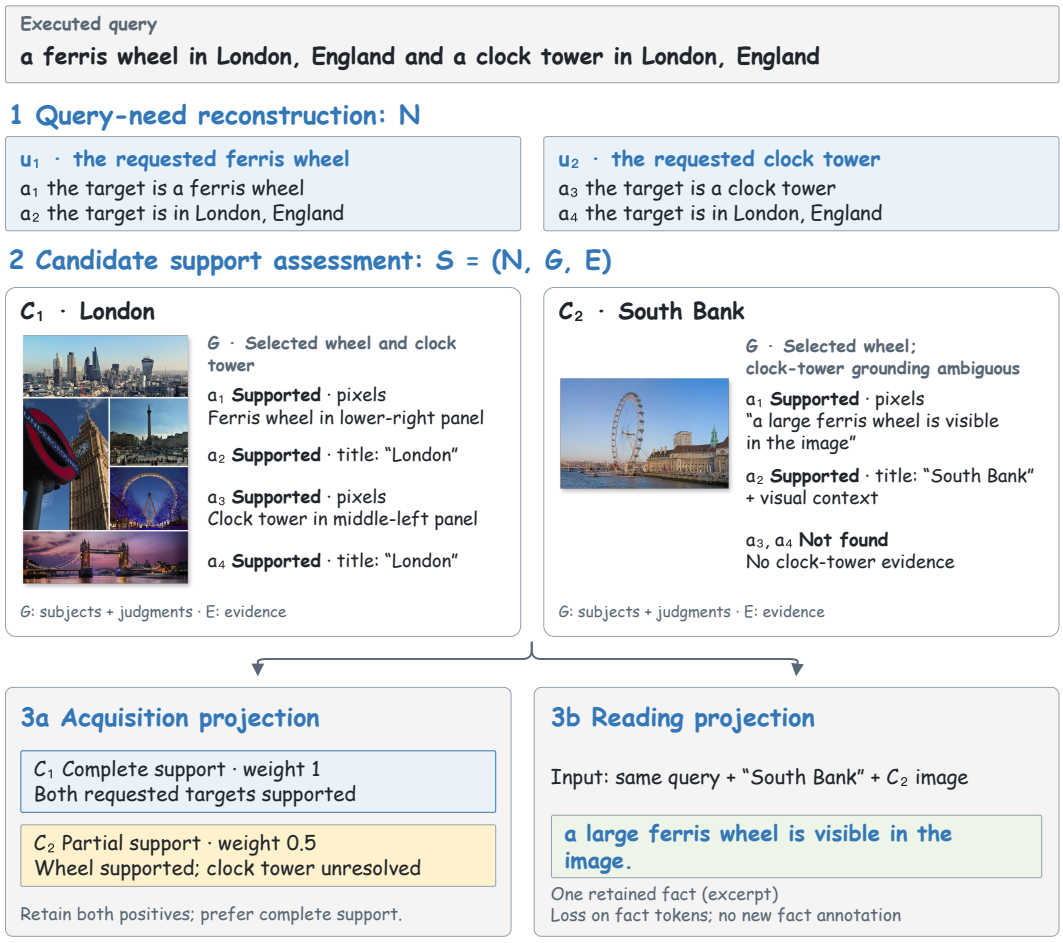


Figure 8: **Shared annotations for acquisition and reading supervision.** The query requirements N are shared across candidates; G records identified subjects and support judgments, and E contains the supporting evidence. The acquisition weights of 1 and 0.5 apply to complete and partial support, respectively, when both levels are present. The reading branch shows one retained visual fact from the *South Bank* annotation. Evidence descriptions are abridged for display; the reading-target excerpt is reproduced verbatim.

F PROMPTS

Prompts P1–P3 show selected instructions for query-need reconstruction, candidate support assessment, and the acquisition projection. P4 provides the full system instruction and input template for reading adaptation. P5 reproduces the complete answer-judging instructions. Quoted instructions are reproduced verbatim; input and output summaries are abbreviated.

P1 Query-need reconstruction

Input: Executed query only.
Output: Ordered target roles and requirements: {`targets: [{target, atoms: [...]}]`}.

Preserve the meaning of the original query exactly. Do not answer the query, retrieve information, inspect candidates, correct the query, or use outside knowledge.

A target group represents one semantic result role, not one concrete answer instance. A target role may match one or multiple results.

Create separate target groups only when the query independently requests semantically distinct result roles.

Under each target, split the query into the smallest independently judgeable factual atoms. However, do not split a relation from a time, purpose, manner, location, comparison, or other modifier that is necessary to preserve the relation's exact meaning.

P2 Candidate support assessment

Input: Executed query, reconstructed requirements, and one candidate: title and passage or title and image.
Output: Subject grounding; per-requirement status and evidence with `source`, `stance`, and `content`.

Evaluate only the current candidate. Do not compare candidates and do not output a final relevance label.

All atom judgments for one target must refer to the same grounding. Do not combine evidence from different foreground or background subjects.

Missing support is not contradiction. A different, related, or non-exclusive fact is not contradiction.

Titles are valid first-class evidence. For image candidates, title and pixels may be used together. Identity, location, or context does not need to be established by pixels alone.

Pixel evidence must describe only visibly observable content. Do not include external biography, dates, league membership, or other non-visible facts in pixel evidence.

Status vocabulary: `supported`, `contradicted`, `not_found`, `uncertain`, `conflicting_evidence`.

Acquisition projection and reading input. In P3, `direct` and `validated_component` denote complete and partial support, respectively; *Stage 1* refers to the candidate support assessment in P2. P4 pairs the reading input with visual or OCR facts already recorded in the shared annotations.

P3 Acquisition projection

Input: Executed query, reconstructed requirements, exact candidate title, and candidate support assessment.
Output: {`label: direct | validated_component | negative | abstain`}.

Stage 1 has already examined the candidate. Do not request, reconstruct, or infer any unseen candidate text or image content.

Do not decide by counting supported, contradicted, not_found, uncertain, or conflicting_evidence atoms.

Choose direct when the available evidence is sufficient to satisfy the complete retrieval intent of the original query.

Choose validated_component when the candidate does not satisfy the complete query but provides a real, useful, query-specific part of it.

Supporting only a generic object type, such as being a person, film, building, document, or place, is insufficient.

Choose negative when the candidate provides no usable query-specific support, or when reliable evidence establishes a decisive mismatch.

Choose abstain only when the available Stage 1 information is insufficient or internally unresolved in a way that prevents a reliable choice among direct, validated_component, and negative.

P4 Reading adaptation input

Input: Executed query, natural title, and one image.
Target: Query-relevant visual or OCR facts reused from the shared annotations.

System instruction (complete). Read the supplied image for the current search query. The title provides identity context. Describe only relevant details or text supported by the image pixels; do not infer unseen facts from the title.

User template.
```
Search query: {executed_query}
Image title: {natural_title}
Image:
{single_image}
```

Target excerpt (*South Bank*, Figure 8). "a large ferris wheel is visible in the image."

Answer-judging prompt. P5 gives the complete system instruction and user template for answer evaluation.

P5 Standard answer judge

Input: Question, reference answer aliases, and extracted final answer.
Output: `Yes` or `No`.

System instruction (complete).

You are an AI assistant tasked with evaluating the correctness of model responses based on the question, and ground truth answer.

Your judgment should follow these principles:

1. Consider the question, and ground truth answer holistically before evaluating the model's response.

2. Your decision should be strictly Yes or No, based on whether the model's response is factually accurate and aligns with the ground truth answer.

3. If the model response is a more specific form of the ground truth answer, it is correct.

4. If the model response includes all key information but adds minor details, it is correct as long as the extra details are factually correct.

5. If the model response contradicts, modifies, or omits critical parts of the answer, it is incorrect.

6. For numerical values, ensure correctness even when presented in different units.

7. For names, check for first and last name correctness. If the middle name is extra but correct, consider it correct.

8. For yes/no questions, the response must exactly match "Yes" or "No" to be correct.

9. If the model response contains refusal statements, and does not directly answer the question, it must be judged incorrect.

10. If there are multiple candidate answers, you can also evaluate the model's response against all of them. If the response aligns with at least one candidate according to the rules above, it should be considered correct.

User template.

Question: {question}

Ground truth answer: {ground_truth}

Model response: {model_response}

Is the model response correct? Reply with strictly 'Yes' or 'No'.