GRAPHROVER: STRUCTURED POST-TRAINING FOR SCIENTIFIC AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

To work like scientists, agents must gather useful evidence and provide accurate, complete explanations. Yet imitating successful trajectories can reinforce unnecessary searches, while rewarding final conclusions alone can overlook missing facts and experimental conditions. We introduce GraphRover, a graph-guided post-training framework for scientific agents. First, Graph-Guided Inquiry Distillation constructs an operation–evidence graph to identify how tool interactions inform later decisions and answer formation. It masks imitation loss for confidently redundant or unproductive actions while preserving the full interaction history, allowing agents to learn useful behavior without losing the context behind it. Second, Completeness-Aware Reinforcement Learning uses rubrics grounded in reference answers and retrieved evidence to specify the findings, conditions, and conclusions required by each question. Its reward combines item-level answer quality with the coverage of adequately satisfied requirements, encouraging accurate scientific explanations with fewer omissions. Across five benchmarks spanning scientific reasoning, research, and deep search, GraphRover improves its base model's average score from 38.71 to 55.96. It outperforms both Qwen3.5-35B-A3B and Qwen3.6-35B-A3B on three benchmarks, exceeding the stronger 35B reference on each by 4.50 percentage points on FrontierScience-Olympiad, 9.10 percentage points on FrontierScience-Research, and 7.00 F1 points on DeepSearchQA.

1 INTRODUCTION

To work like scientists, agents must gather useful evidence and provide accurate, complete explanations. Tool-augmented large language models (LLMs), exemplified by ReAct (Yao et al., 2023) and Toolformer (Schick et al., 2023), provide a foundation for this goal by combining language generation with external actions. Building capable scientific agents, however, requires more than access to tools: post-training must teach both how to investigate a question and how to explain the answer.

Recent work has increasingly focused on how agents learn from their interactions. For supervised learning, critical-step selection (Chen et al., 2025b) identifies informative decisions, Hindsight Supervised Learning (Li et al., 2026) masks irrelevant actions, and trajectory supervision (Reddy et al., 2026) examines the value of intermediate interaction histories. For reinforcement learning (RL), Search-R1 (Jin et al., 2025) learns retrieval behavior from outcome feedback, while ARPO (Dong et al., 2026) improves attribution around tool calls and iStar (Liu et al., 2026b) learns implicit step rewards from trajectory preferences. These approaches move beyond simply collecting successful demonstrations toward designing more informative supervision. Our focus is on structuring this supervision around both the investigation process and the requirements of a scientific answer.

This objective presents two related challenges. First, an interaction need not be worth imitating to remain useful as context. Successful trajectories can contain redundant searches or ineffective tool calls, yet their observations may introduce a source, entity, or failure signal needed by later decisions. Uniform imitation can reinforce unnecessary behavior, whereas deleting these interactions can remove information needed to understand subsequent actions. Their training value must therefore be assessed through their dependencies within the investigation, rather than in isolation. Second, a correct conclusion does not ensure a complete scientific explanation. Reporting a measured result without the experimental conditions requested by the question leaves the task incomplete, even when

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.

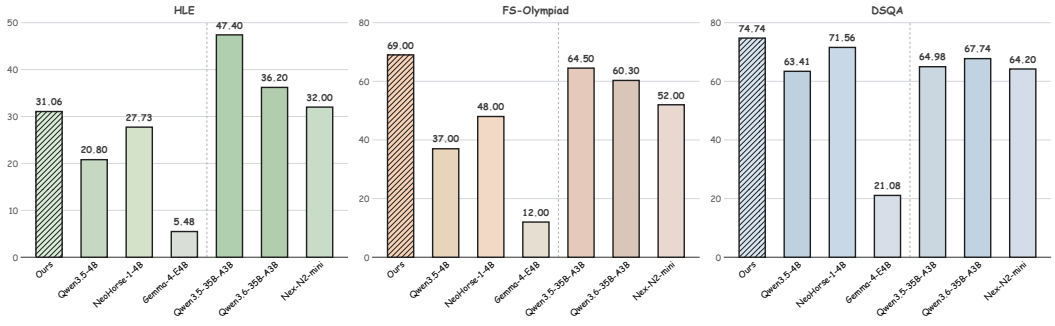


Figure 1: Performance comparison of GraphRover with representative compact and larger open models on HLE, FrontierScience-Olympiad, and DeepSearchQA. The dashed vertical line separates compact models from larger-scale references.

the value is correct. Rewarding only the conclusion can overlook such omissions, while step-level credit assignment alone does not specify which requirements the final answer must satisfy.

To address these challenges, we introduce GraphRover, a structured post-training framework for scientific agents. First, *Graph-Guided Inquiry Distillation* constructs an *operation–evidence graph* over each outcome-correct teacher trajectory to trace how reasoning, tool calls, and observations inform later decisions and answer formation. The graph guides imitation-loss masking for interactions identified as redundant or unproductive with high confidence, while the complete chronological history remains visible as model context. This separates what the agent needs to observe from what it should imitate; the graph is used only during offline training-data processing. Second, *Completeness-Aware Reinforcement Learning* uses rubrics grounded in reference answers and retrieved evidence to specify the findings, experimental conditions, relations, and conclusions required by each question. Its reward combines weighted item-level answer quality with the weighted proportion of requirements that meet a satisfaction threshold, and is optimized using GRPO (Shao et al., 2024). Together, the graph structures which investigative behaviors are learned, while the rubrics structure how the resulting explanations are rewarded.

We instantiate the framework as GraphRover using Qwen3.5-4B (Qwen Team, 2026a). Across five benchmarks spanning scientific reasoning, research, and deep search, GraphRover improves its base model's mean benchmark score from 38.71 to 55.96 (Table 1). It outperforms both Qwen3.5-35B-A3B (Qwen Team, 2026a) and Qwen3.6-35B-A3B (Qwen Team, 2026b) on the Olympiad and Research tracks of FrontierScience (Wang et al., 2026) and on DeepSearchQA (Gupta et al., 2026). Relative to the stronger 35B result on each benchmark, the gains are 4.50 percentage points on FrontierScience-Olympiad, 9.10 percentage points on FrontierScience-Research, and 7.00 F1 points on DeepSearchQA. Figure 1 highlights representative comparisons across model scales. Ablations further show that including requirement coverage improves over optimizing average rubric quality alone on all three ablation benchmarks (Table 2). These results support structured post-training as an effective way to improve scientific agents without increasing model size.

Our main contributions are:

- We propose *Graph-Guided Inquiry Distillation*, which uses operation–evidence dependencies to guide selective imitation while preserving the complete context needed by subsequent decisions.
- We introduce *Completeness-Aware Reinforcement Learning*, which combines rubric-based answer quality with thresholded requirement coverage to encourage accurate scientific explanations with fewer omissions.
- We develop GraphRover and demonstrate substantial improvements across four benchmarks, including performance exceeding two 35B reference models on three scientific reasoning, research, and deep-search benchmarks.

2 Related Work

2.1 Tool-Augmented Reasoning and Structured Trajectory Supervision

Tool-augmented language models extend reasoning through external search, browsing, and computation. ReAct (Yao et al., 2023) interleaves reasoning and actions, while Toolformer (Schick et al., 2023) learns when and how to invoke tools. Recent work further studies how interaction trajectories should be supervised: ATLaS (Chen et al., 2025b) emphasizes critical decisions, Hindsight Supervised Learning (Li et al., 2026) relabels trajectories and masks irrelevant actions, Trajectory Supervision (Reddy et al., 2026) examines the value of intermediate interaction histories, and Anchored Supervised Fine-Tuning (Zhu et al., 2026) combines probability-aware reweighting with reference-policy anchoring.

Our *Graph-Guided Inquiry Distillation* instead uses explicit operation–evidence dependencies to determine imitation targets. By linking reasoning, tool calls, observations, evidence, and downstream decisions, it evaluates each interaction by its role within the investigation. Confidently redundant or unproductive interactions are masked from the imitation loss while the complete chronological history remains visible. Thus, the key distinction lies in dependency-guided supervision rather than loss masking itself; the graph is used only offline and adds no inference-time overhead.

2.2 Agentic Reinforcement Learning and Rubric-Based Rewards

Agentic RL has increasingly focused on policy optimization and fine-grained credit assignment. GRPO (Shao et al., 2024) performs group-relative optimization, while Search-R1 (Jin et al., 2025) applies outcome-based RL to retrieval-augmented reasoning. ARPO (Dong et al., 2026), iStar (Liu et al., 2026b), StepOPSD (Zhang et al., 2026), and InfoTree (Hu et al., 2026) further improve credit assignment or rollout informativeness for long-horizon agent interactions. Another line of work structures the reward itself. Rubrics as Rewards (Gunjal et al., 2025) and Rubric-Grounded RL (Bhattarai et al., 2026) use instance-specific rubric criteria, while Step-wise Rubrics as Rewards (Xie et al., 2026) attributes criteria to reasoning steps and GEAR (Lv et al., 2026) models dependencies among rubric items.

Our *Completeness-Aware Reinforcement Learning* instead combines weighted item-level answer quality with thresholded requirement coverage. This rollout-level reward distinguishes answers with similar average quality but different levels of requirement satisfaction, and is optimized using GRPO. GraphRover thus couples dependency-aware inquiry supervision with explicit optimization of answer quality and coverage.

3 Method

GraphRover structures post-training around two complementary goals: learning useful scientific inquiry and producing accurate, complete explanations. As illustrated in Figure 2, *Graph-Guided Inquiry Distillation* (GGID) uses operation–evidence dependencies to guide selective imitation while preserving the full interaction history. Building on this initialization, *Completeness-Aware Reinforcement Learning* (CARL) optimizes the agent against question-specific scientific requirements such as accuracy and completeness, rewarding both the quality of its answers and the coverage of the required content.

3.1 Problem Formulation

For scientific reasoning, we consider scientific questions that require reasoning and interaction with external tools, such as web search and document readers. Let x denote a question and y its reference answer. A teacher trajectory is represented as

$$\tau = (x, \xi_1, \ldots, \xi_T, s, \hat{y}), \qquad \xi_t = (r_t, a_t, o_t), \tag{1}$$

where r_t is the reasoning associated with interaction t, a_t is a tool invocation, o_t is the returned observation, s is the final synthesis, and $\hat{y}$ is the recorded answer. Our training dataset $\mathcal{D} = \{(x, \tau, y)\}$ contains teacher trajectories that pass answer-level filtering, such that $\hat{y}$ agrees with y.

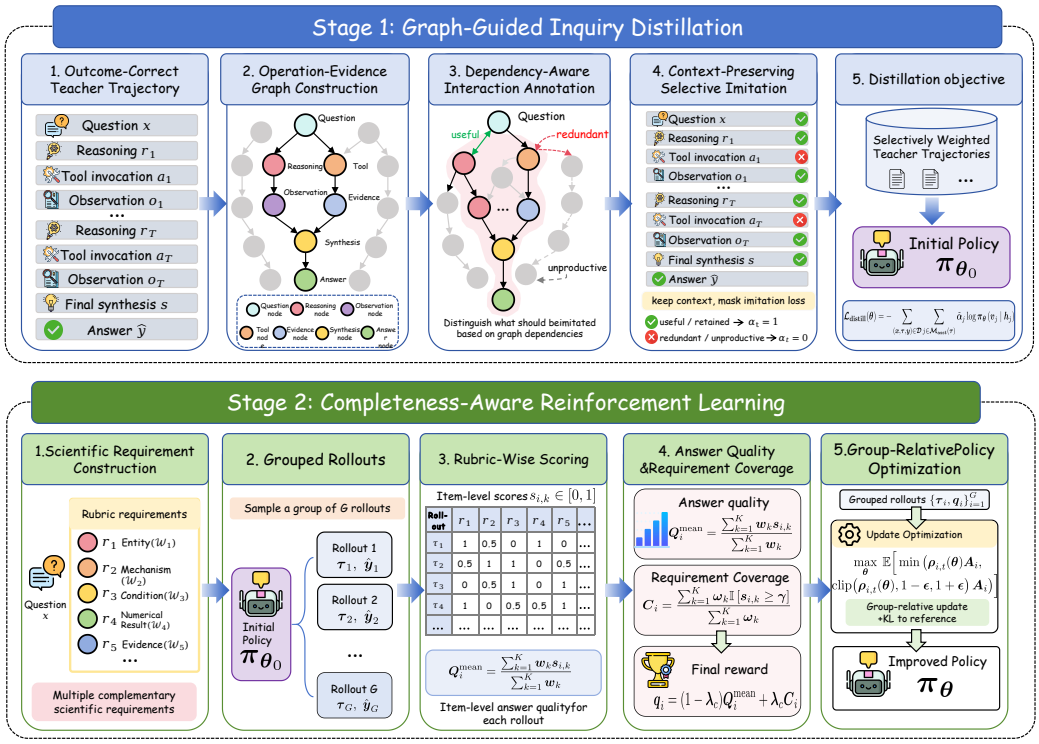


Figure 2: Overview of GraphRover, a structured post-training framework for scientific agents. Stage 1, *Graph-Guided Inquiry Distillation*, constructs an operation–evidence graph over outcome-correct teacher trajectories to guide imitation-loss masking for confidently redundant or unproductive interactions, while preserving the complete chronological history as model context. Stage 2, *Completeness-Aware Reinforcement Learning*, initializes from the distilled policy and evaluates grouped rollouts against scientific rubrics grounded in reference answers and retrieved evidence. Its reward combines item-level answer quality with thresholded requirement coverage for group-relative optimization, encouraging accurate explanations with fewer omissions.

In the context of token-level objectives, define v_j as the j-th token following x in the serialized interaction history, with $h_j = (x, v_{<j})$ representing its full preceding context. We denote by $\mathcal{M}_{\text{asst}}(\tau)$ the positions produced by the assistant, including reasoning, tool invocations, and the final response, whereas tool observations are excluded from the policy loss. Starting from a base policy π_0, Stage 1 learns a selectively supervised policy π_{θ_0}, which initializes Stage 2 online optimization.

3.2 Stage 1: Graph-Guided Inquiry Distillation

A successful investigation may still include unnecessary detours alongside useful decisions. We aim to distinguish their imitation value while preserving the information required to understand subsequent actions. Accordingly, rather than shortening the teacher trajectory, we use the graph to guide supervision.

Operation–evidence graph. For each outcome-correct trajectory τ, we construct a directed operation–evidence graph

$$G_\tau = (V_\tau, E_\tau), \tag{2}$$

representing reasoning operations, tool invocations, observations, acquired evidence, and answer formation, together with their functional dependencies. These dependencies expose how information from one interaction contributes to later decisions, and it also includes interactions that do not directly support the final answer. For example, an observation may introduce an entity needed for a subsequent search even when it does not resolve the current query. In our method, every graph element remains aligned with its original position in the teacher trajectory. The graph schema and construction procedure are provided in the Appendix.

Dependency-aware interaction annotation. We combine deterministic structure extraction with model-assisted semantic judgment. Chronological order, tool-call/observation pairing, normalized tool arguments, and source provenance are recovered deterministically. An offline graph judge $\mathcal{J}_\phi$ then assesses each interaction using both its local content and its dependencies within the completed trajectory:

$$\mathcal{J}_\phi(\xi_t, G_\tau) = (z_t, c_t), \qquad c_t \in [0, 1], \tag{3}$$

where z_t is the estimated functional role and c_t is the judgment confidence. The judge considers how an interaction relates to surrounding observations and later decisions, rather than evaluating its apparent relevance to the final answer alone. It produces supervision annotations without rewriting teacher reasoning, tool arguments, observations, or answers.

Context-preserving selective imitation. We suppress supervision only when an interaction is identified as redundant or unproductive with high confidence. Let $\mathcal{Z}_- = \{\text{REDUNDANT}, \text{UNPRODUCTIVE}\}$ denote these negative roles. The interaction-level supervision weight is

$$\alpha_t = \begin{cases} \lambda_{\text{bad}}, & z_t \in \mathcal{Z}_- \text{ and } c_t \geq \tau_{\text{mask}}, \\ 1, & \text{otherwise}, \end{cases} \tag{4}$$

where τ_{mask} is the confidence threshold and $\lambda_{\text{bad}} \in [0, 1]$ controls the supervision retained on flagged interactions. We use $\lambda_{\text{bad}} = 0$ by default, corresponding to loss masking. Low-confidence negative judgments retain full supervision. This conservative rule preserves supervision on exploration, query reformulation, verification, and recovery unless they are confidently judged redundant or unproductive.

Let $\tilde{\alpha}_j$ denote the supervision weight assigned to assistant token v_j, with tokens belonging to interaction ξ_t inheriting α_t. The distillation objective is

$$\mathcal{L}_{\text{distill}}(\theta) = -\sum_{(x,\tau,y)\in\mathcal{D}} \sum_{j\in\mathcal{M}_{\text{asst}}(\tau)} \tilde{\alpha}_j \log \pi_\theta(v_j \mid h_j). \tag{5}$$

When all assistant-token weights are set to one, the training objective becomes standard supervised fine-tuning. The key observation is that masking influences only the direct imitation loss. Because every interaction is preserved in its original chronological order, h_j encompasses the complete teacher history, including the masked actions and their observations. Consequently, the agent is able to learn useful subsequent behavior without being directly optimized to replicate each earlier detour. The resulting policy π_{θ_0} serves to initialize Stage 2. The graph, the graph judge, and the supervision annotations are used exclusively during offline processing and are not required at inference time.

3.3 STAGE 2: COMPLETENESS-AWARE REINFORCEMENT LEARNING

Learning useful investigative behavior alone does not determine what a satisfactory scientific answer should contain. Stage 2 therefore makes the answer requirements explicit and optimizes the policy to fulfill them. To this end, we design scientific rubrics, assess the quality of each required component, and aggregate these scores with requirement coverage to form a rollout-level reward.

Scientific requirement construction. For each training question x, we construct a weighted rubric

$$\mathcal{R}(x) = \{(u_k, \omega_k)\}_{k=1}^{K}, \tag{6}$$

where u_k is an atomic answer requirement and $\omega_k \geq 0$ is its importance weight, with $\sum_{k=1}^{K} \omega_k > 0$. Requirements may specify entities, numerical findings, causal or mechanistic relations, experimental conditions, methodological details, or evidence-supported conclusions.

The construction of training questions ensures that a satisfactory response requires several complementary components instead of one isolated target. The rubrics are based on teacher trajectories employed exclusively for training, reference answers, and retrieved evidence. For example, when a question asks for a measured result and its experimental conditions, both components must be addressed. Reference answers and rubric annotations are used for reward computation after rollout completion; they are never provided to the policy during generation.

Answer quality and requirement coverage. For a rollout τ_i with final answer $\hat{y}_i$, each requirement u_k is evaluated independently to obtain $s_{i,k} \in [0, 1]$. We use deterministic matching whenever reliable and offline semantic evaluation for paraphrased, relational, or explanatory content. The weighted mean score measures overall answer quality:

$$Q_i^{\text{mean}} = \frac{\sum_{k=1}^{K} \omega_k s_{i,k}}{\sum_{k=1}^{K} \omega_k}. \tag{7}$$

This provides graded answer-level feedback rather than reducing the response to a single correctness judgment.

Although omitted requirements reduce the mean score, the mean does not explicitly indicate how many requirements have reached a satisfactory level. We therefore also measure requirement coverage:

$$C_i = \frac{\sum_{k=1}^{K} \omega_k \mathbb{I}[s_{i,k} \geq \gamma]}{\sum_{k=1}^{K} \omega_k}, \tag{8}$$

where γ is the satisfaction threshold. With graded item scores, answers with the same mean quality can satisfy different numbers of requirements above this threshold. Coverage captures this distinction as a weighted fraction; it is not an all-or-nothing guarantee of a complete answer.

The final reward combines quality and coverage:

$$R_i = (1 - \lambda_c) Q_i^{\text{mean}} + \lambda_c C_i, \tag{9}$$

where $\lambda_c \in [0, 1]$ governs how much coverage contributes. The quality term gives credit for partial fulfillment, whereas the coverage term further emphasizes raising requirements to an adequate level. The combination of the two encourages accurate explanations that cover more of the content required by the question. Instead of assigning distinct rewards to individual tool interactions, we aggregate the item scores into a single rollout reward.

Group-relative policy optimization. We optimize this reward using GRPO (Shao et al., 2024). For each question, we sample a group of G rollouts from π_{old}, a snapshot of the current policy. Let μ_G and σ_G denote the mean and standard deviation of their rewards. The advantage for rollout i is

$$A_i = \text{clip}\left(\frac{R_i - \mu_G}{\max(\sigma_G, \delta)}, -A_{\max}, A_{\max}\right), \tag{10}$$

where $\delta > 0$ stabilizes normalization and $A_{\max}$ bounds the advantage magnitude. Groups with negligible reward variance are excluded from policy updates rather than assigned artificial preferences.

For assistant-generated token $v_{i,j}$ with complete interaction context $h_{i,j}$, the importance ratio is

$$\rho_{i,j}(\theta) = \frac{\pi_\theta(v_{i,j} \mid h_{i,j})}{\pi_{\text{old}}(v_{i,j} \mid h_{i,j})}. \tag{11}$$

We minimize

$$\begin{aligned} \mathcal{L}_{\text{RL}}(\theta) = & -\mathbb{E}_{i,j}\Big[\min\Big(\rho_{i,j} A_i,\ \text{clip}(\rho_{i,j}, 1-\epsilon, 1+\epsilon) A_i\Big)\Big] \\ & + \beta_{\text{KL}} D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}}), \end{aligned} \tag{12}$$

where ϵ is the clipping parameter, π_{ref} is a frozen copy of the Stage 1 policy π_{θ_0}, and β_{KL} controls regularization toward this initialization. The expectation is over retained rollouts and their assistant-generated tokens. Each rollout advantage supervises reasoning, tool invocations, and final-answer generation. Tool observations remain visible in the context but receive no policy loss.

4 Experiment

We evaluate GraphRover on scientific reasoning, scientific research, and deep-search benchmarks. We report the experimental setup, main results, and ablations in this section, with further implementation details and analyses provided in Appendix A.

Table 1: Main results on scientific reasoning, research, and deep search. All scores are percentages (DSQA: F1); **Bold** and underlined values indicate the best and second-best results in each column.

Model	Scientific Reasoning		Scientific Research	Deep Search		Avg.
	HLE w/ Tools	FS-Olympiad	FS-Research	DSQA	XBench	
Compact baselines						
Qwen3.5-4B *(Base)*	20.80	37.00	3.33	63.41	69.00	38.71
Qwen3.5-9B	24.08	40.00	16.67	70.52	75.00	45.25
Phi-4-mini	3.21	3.00	0.00	1.87	2.00	2.02
Gemma-4-E2B	4.60	10.00	0.00	9.50	30.00	10.82
Gemma-4-E4B	5.48	12.00	0.00	21.08	40.00	15.71
Llama-3.2-3B	4.40	1.00	0.00	5.62	10.00	4.20
Ministral-3-3B-Reasoning	4.96	11.00	0.00	14.55	3.00	6.70
NeoHorse-1-4B	27.73	48.00	20.00	<u>71.56</u>	76.00	48.66
Larger reference models						
Qwen3.5-35B-A3B	**47.40**	<u>64.50</u>	2.50	64.98	<u>77.00</u>	<u>51.28</u>
Qwen3.6-35B-A3B	<u>36.20</u>	60.30	<u>20.90</u>	67.74	71.00	51.23
Nex-N2-mini	32.00	52.00	5.00	64.20	**82.00**	47.04
GraphRover (Ours)	31.06	**69.00**	**30.00**	**74.74**	75.00	**55.96**

4.1 Experiment Setup

Benchmarks. We evaluate three complementary capabilities. **Scientific Reasoning** is measured with Humanity's Last Exam (HLE) (Phan et al., 2025) and FrontierScience-Olympiad (FS-Olympiad) (Wang et al., 2026). **Scientific Research** is evaluated on FrontierScience-Research (FS-Research) (Wang et al., 2026). **DeepSearch** is evaluated with DeepSearchQA (DSQA) (Gupta et al., 2026) and XBench-DeepSearch (Chen et al., 2025a), which require iterative retrieval, evidence aggregation, and answer synthesis.

Methods of Comparison. We compare GraphRover with compact open models, including Qwen3.5-4B, Qwen3.5-9B (Qwen Team, 2026a), Phi-4-mini (Abouelenin et al., 2025), Gemma-4-E2B, Gemma-4-E4B (Gemma Team, 2026), Llama-3.2-3B (Grattafiori et al., 2024), Ministral-3-3B-Reasoning (Liu et al., 2026a) and NeoHorse-1-4B (Team et al., 2026). We also include larger references, Qwen3.5-35B-A3B (Qwen Team, 2026a), Qwen3.6-35B-A3B (Qwen Team, 2026b), and Nex-N2-mini (Cai et al., 2025). All models use the same tool interface and evaluation protocol unless otherwise stated.

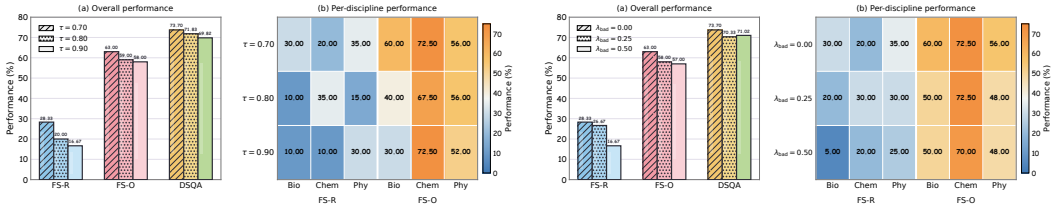
4.2 Main Results

Comparison with the base model. Table 1 shows that GraphRover improves Qwen3.5-4B on all four benchmarks, increasing the average score from 38.71 to 55.96 (+17.25 points; 44.6% relative improvement). The largest absolute gains occur on FS-Olympiad and FS-Research, where scores increase from 37.00 to 69.00 and from 3.33 to 30.00, respectively, corresponding to improvements of 32.00 and 26.67 percentage points. The gains also extend to HLE (+10.26 percentage points), DSQA (+11.33 F1 points), and XBench (+6.00 percentage points). Thus, the improvements span scientific reasoning, research-level question answering, and deep search, rather than being confined to a single benchmark.

Comparison with strong baselines. GraphRover achieves the highest average score among all compared models and ranks first on FS-Olympiad, FS-Research, and DSQA. On these benchmarks, it exceeds the strongest competing result by 4.50 percentage points, 9.10 percentage points, and 3.18 F1 points, respectively. Compared with NeoHorse-1-4B, the strongest compact baseline by average score, GraphRover performs better on four of five benchmarks and improves the average from 48.66 to 55.96 (+7.30 points). This advantage carries over to larger reference models: on FS-Olympiad, FS-Research, and DSQA, GraphRover beats both Qwen3.5-35B-A3B and Qwen3.6-35B-A3B, and its average score exceeds their respective averages of 51.28 and 51.23 by 4.68 and 4.73 points.

Table 2: Ablation of the two-stage post-training framework. FS-R/FS-O use accuracy and DSQA uses F1; discipline-wise accuracy is reported for biology, chemistry, and physics.

Method	Settings			FS-Research				FS-Olympiad				DSQA
	Multi-Req	Rubric Quality	Coverage	overall	Bio	Chem	Phy	overall	Bio	Chem	Phy	
Base	–	–	–	3.33	0.00	10.00	0.00	37.00	20.00	52.50	28.00	63.41
SFT	–	–	–	25.00	20.00	35.00	20.00	66.00	**60.00**	72.50	**62.00**	71.43
No-Graph Masking	–	–	–	13.33	15.00	10.00	15.00	34.00	10.00	37.50	36.00	30.05
Graph Pruning	–	–	–	15.00	5.00	15.00	25.00	33.00	20.00	30.00	38.00	33.80
GGID	–	–	–	28.33	**30.00**	20.00	**35.00**	63.00	**60.00**	72.50	56.00	73.70
GRPO	–	–	–	21.67	10.00	**40.00**	15.00	60.00	**60.00**	67.50	54.00	69.39
Rubric-wise GRPO	–	✓	–	23.33	15.00	25.00	30.00	62.00	50.00	75.00	54.00	73.09
CARL w/o Coverage	✓	✓	–	23.33	15.00	30.00	25.00	59.00	40.00	67.50	56.00	72.43
CARL (Ours)	✓	✓	✓	**30.00**	25.00	35.00	30.00	**69.00**	**60.00**	**80.00**	**62.00**	**74.74**



(a) Masking threshold τ_{mask}. (b) Supervision weight λ_{bad}.

Figure 3: Stage 1 hyperparameter sensitivity. Each subfigure reports overall performance on FS-R, FS-O, and DSQA, and discipline-wise FrontierScience results for biology, chemistry, and physics.

These gains are achieved without increasing the 4B backbone size. However, GraphRover does not lead on every benchmark: Qwen3.5-35B-A3B remains stronger on HLE, while Nex-N2-mini achieves the best XBench result. Overall, the results support structured post-training as an effective approach to improving compact scientific agents, with particularly strong gains on the two FrontierScience tracks. Table 2 further examines the contributions of Graph-Guided Inquiry Distillation and Completeness-Aware Reinforcement Learning.

4.3 Ablation Studies

Table 2 evaluates the two post-training stages and their key design choices. For Stage 1, GGID improves over standard SFT on FS-R and DSQA, while remaining slightly lower on FS-O. More importantly, the controlled Stage 1 variants show that both dependency guidance and context preservation are critical. The mask-budget-matched *No-Graph Masking* variant performs substantially worse than GGID on all three benchmarks, indicating that selective supervision depends on identifying the right interactions rather than merely masking a fixed amount of training signal. Likewise, *Graph Pruning*, which removes flagged interactions instead of retaining them as context, degrades sharply across FS-R, FS-O, and DSQA. Together, these results support the central GGID design of using operation–evidence dependencies to guide loss masking while preserving the complete interaction history.

For Stage 2, all variants are initialized from GGID. Standard GRPO reduces performance, while rubric-wise quality supervision provides only a partial recovery. The full CARL model is the only tested RL variant that improves over GGID on all three benchmarks, reaching 30.00 on FS-R, 69.00 on FS-O, and 74.74 on DSQA. The controlled *w/o Coverage* variant differs from CARL only by removing the coverage term C_i; restoring coverage improves FS-R and FS-O by 6.67 and 10.00 percentage points, respectively, and DSQA by 2.31 F1 points. These gains also hold across all six FrontierScience discipline subsets, showing that requirement coverage provides complementary supervision beyond mean rubric-wise answer quality.

4.4 Additional Analysis

Stage 1 sensitivity. Figure 3a and 3b study the two main GGID hyperparameters. For τ_{mask}, the default value of 0.70 gives the strongest overall performance across FS-R, FS-O, and DSQA, while larger thresholds generally degrade performance. For λ_{bad}, assigning zero supervision to

Table 4: GGID vs. CARL on multi-part FS-Research questions in physics and chemistry.

Method	Answer summary	Rubric-level assessment
Physics: Supernova detection		
GGID	Correctly states the PMT trigger, efficiencies, inverse-square scaling, and event separation.	Satisfies most requirements, but gives an *inaccurate duration estimate* and *incomplete background characterization*.
CARL	Provides the trigger, efficiencies, inverse-square scaling, and event separation, together with **event scale and duration** and **detector-specific backgrounds**.	Satisfies **all stored rubric requirements**, yielding a complete multi-component response.
Chemistry: Ni–Fe reaction mechanisms		
GGID	Distinguishes Ni(III)-mediated ethanol oxidation from cooperative Ni–O–Fe oxygen evolution; discusses redox shifts and site dilution.	Omits the *edge–bulk distinction*, *complete redox cycle*, *composition-wise competition*, and d^5/d^7 *comparison*.
CARL	Identifies **trace Fe at edge/defect sites** and the **Ni(II)→Ni(III)→Ni(II)** cycle; discusses redox shifts, site blocking, and electronic changes at higher Fe loading.	Satisfies **more rubric requirements**, although *several mechanistic details remain partial*.

confidently redundant or unproductive interactions performs best overall, whereas retaining partial imitation loss reduces performance. We therefore use $\tau_{\text{mask}} = 0.70$ and $\lambda_{\text{bad}} = 0$ in all experiments.

Answer Length and Tool-Use Analysis We test whether CARL's gains can be explained simply by longer responses or more extensive tool use. As shown in Table 3, CARL produces slightly shorter responses and uses fewer tool calls on FS-Research than GGID and the controlled *w/o Coverage* variant. Despite this, CARL achieves stronger FS-Research accuracy and mean rubric quality, suggesting that the gains are not trivially attributable to greater generation length or search effort. We treat this as descriptive evidence rather than a causal measure of tool-use efficiency.

Table 3: FS-R response length and tool use.

Method	Chars	Calls
GGID	11.6K	24.0
CARL w/o Coverage	11.1K	25.0
CARL	11.0K	22.7

Qualitative Example Table 4 compares GGID and CARL on multi-part FS-Research questions in physics and chemistry. In the physics case, GGID captures the main detection requirements but gives an inaccurate event duration and incompletely characterizes backgrounds. CARL addresses these shortcomings and satisfies all stored rubric requirements. In the chemistry case, GGID identifies the main reaction mechanisms but omits finer-grained distinctions. CARL provides a more complete explanation by explicitly identifying edge/defect Fe sites and the full Ni redox cycle, alongside redox shifts, site blocking, and electronic changes at higher Fe loading, although several mechanistic details remain partial. These examples illustrate improvements in both factual accuracy and requirement coverage, consistent with CARL's joint optimization objective, while showing that broader coverage does not guarantee fully resolved scientific explanations.

5 Conclusion

We introduced GraphRover, a compact scientific model trained with two-stage structured post-training. Graph-Guided Inquiry Distillation (GGID) guides selective imitation through operation–evidence dependencies while preserving full interaction context, while Completeness-Aware Reinforcement Learning (CARL) jointly optimizes answer quality and requirement coverage. Across five benchmarks, GraphRover improves over its 4B base model on every task, raising the average score from 38.71 to 55.96 (+17.25 points; 44.6% relative improvement). It achieves the highest overall average and ranks first on FS-Olympiad, FS-Research, and DSQA, outperforming both 35B references on these three benchmarks. Ablations further support the benefit of explicit requirement coverage beyond average rubric-wise quality. Overall, GraphRover shows that structuring both imitation and reward can strengthen tool-using scientific agents without increasing model size.

AI USE STATEMENT

In this work, we used generative AI tools only for text polishing, including grammar correction and improvements to clarity, fluency, and presentation. We did not use generative AI tools for research ideation, method design, experimental implementation, data analysis, or scientific interpretation. All AI-assisted edits were reviewed by the authors for correctness and faithfulness to the original technical content. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*, 2025. URL https://arxiv.org/abs/2503.01743.

Manish Bhattarai, Ismael Boureima, Nishath Rajiv Ranasinghe, Scott Pakin, and Dan O'Malley. Rubric-Grounded RL: Structured judge rewards for generalizable reasoning. *arXiv preprint arXiv:2605.08061*, 2026. doi: 10.48550/arXiv.2605.08061.

Yuxuan Cai, Lu Chen, Qiaoling Chen, Yuyang Ding, Liwen Fan, Wenjie Fu, Yufei Gao, Honglin Guo, Pinxue Guo, Zhenhua Han, et al. Nex-N1: Agentic models trained via a unified ecosystem for large-scale environment construction. *arXiv preprint arXiv:2512.04987*, 2025. URL https://arxiv.org/abs/2512.04987.

Kaiyuan Chen, Yixin Ren, Yang Liu, Xiaobo Hu, Haotong Tian, Tianbao Xie, Fangfu Liu, Haoye Zhang, Hongzhang Liu, Yuan Gong, et al. xbench: Tracking agents productivity scaling with profession-aligned real-world evaluations. *arXiv preprint arXiv:2506.13651*, 2025a. URL https://arxiv.org/abs/2506.13651.

Zhixun Chen, Ming Li, Yuxuan Huang, Yali Du, Meng Fang, and Tianyi Zhou. ATLAS: Agent tuning via learning critical steps. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 25334–25349. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.findings-acl.1299. URL https://aclanthology.org/2025.findings-acl.1299/.

Guanting Dong, Hangyu Mao, Kai Ma, Licheng Bao, Yifei Chen, Zhongyuan Wang, Zhongxia Chen, Jiazhen Du, Huiyang Wang, Fuzheng Zhang, Guorui Zhou, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. Agentic reinforced policy optimization. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10009306.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as Rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint arXiv:2507.17746*, 2025. doi: 10.48550/arXiv.2507.17746.

Nikita Gupta, Riju Chatterjee, Lukas Haas, Connie Tao, Andrew Wang, Chang Liu, Hidekazu Oiwa, Elena Gribovskaya, Jan Ackermann, John Blitzer, et al. DeepSearchQA: Bridging the comprehensiveness gap for deep research agents. *arXiv preprint arXiv:2601.20975*, 2026. URL https://arxiv.org/abs/2601.20975.

Yuelin Hu, Zhenbo Yu, Zhengxue Cheng, Wei Liu, and Li Song. Maximizing rollout informativeness under a fixed budget: A submodular view of tree search for tool-use agentic reinforcement learning, 2026. URL https://arxiv.org/abs/2605.05262.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan O. Arik, 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*, 2025. URL https://openreview.net/forum?id=Rwhi91ideu.

Zichao Li, Gang Wu, Zichao Wang, Ruiyi Zhang, Wanrong Zhu, Ryan A. Rossi, Vlad I. Morariu, and Jihyung Kil. Spinning straw into gold: Relabeling LLM agent trajectories in hindsight for successful demonstrations. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=QNfmqMSR7r.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, Alexandre Sablayrolles, Amélie Héliou, Amos You, Andy Ehrenberg, Andy Lo, Anton Eliseev, Antonia Calvi, Avinash Sooriyarachchi, Baptiste Bout, Baptiste Rozière, Baudouin De Monicault, Clémence Lanfranchi, Corentin Barreau, Cyprien Courtot, Daniele Grattarola, Darius Dabert, Diego de las Casas, Elliot Chane-Sane, Faruk Ahmed, Gabrielle Berrada, Gaëtan Ecrepont, Gauthier Guinet, Georgii Novikov, Guillaume Kunsch, Guillaume Lample, Guillaume Martin, Gunshi Gupta, Jan Ludziejewski, Jason Rute, Joachim Studnia, Jonas Amar, Joséphine Delas, Josselin Somerville Roberts, Karmesh Yadav, Khyathi Chandu, Kush Jain, Laurence Aitchison, Laurent Fainsin, Léonard Blier, Lingxiao Zhao, Louis Martin, Lucile Saulnier, Luyu Gao, Maarten Buyl, Margaret Jennings, Marie Pellat, Mark Prins, Mathieu Poirée, Mathilde Guillaumin, Matthieu Dinot, Matthieu Futeral, Maxime Darrin, Maximilian Augustin, Mia Chiquier, Michel Schimpf, Nathan Grinsztajn, Neha Gupta, Nikhil Raghuraman, Olivier Bousquet, Olivier Duchenne, Patricia Wang, Patrick von Platen, Paul Jacob, Paul Wambergue, Paula Kurylowicz, Pavankumar Reddy Muddireddy, Philomène Chagniot, Pierre Stock, Pravesh Agrawal, Quentin Torroba, Romain Sauvestre, Roman Soletskyi, Rupert Menneer, Sagar Vaze, Samuel Barry, Sanchit Gandhi, Siddhant Waghjale, Siddharth Gandhi, Soham Ghosh, Srijan Mishra, Sumukh Aithal, Szymon Antoniak, Teven Le Scao, Théo Cachet, Theo Simon Sorg, Thibaut Lavril, Thiziri Nait Saada, Thomas Chabal, Thomas Foubert, Thomas Robert, Thomas Wang, Tim Lawson, Tom Bewley, Tom Edwards, Umar Jamil, Umberto Tomasini, Valeriia Nemychnikova, Van Phung, Vincent Maladière, Virgile Richard, Wassim Bouaziz, Wen-Ding Li, William Marshall, Xinghui Li, Xinyu Yang, Yassine El Ouahidi, Yihan Wang, Yunhao Tang, and Zaccharie Ramzi. Ministral 3, 2026a. URL https://arxiv.org/abs/2601.08584.

Xiaoqian Liu, Ke Wang, Yuchuan Wu, Fei Huang, Yongbin Li, Jianbin Jiao, and Junge Zhang. Agentic reinforcement learning with implicit step rewards. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=ooROvpmxMV.

Can Lv, Mingju Chen, Heng Chang, and Shiji Zhou. Mitigating false credit propagation: Probabilistic graphical reward aggregation for rubric-based reinforcement learning. *arXiv preprint arXiv:2606.03361*, 2026. doi: 10.48550/arXiv.2606.03361.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025. URL https://arxiv.org/abs/2501.14249.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026b. URL https://qwen.ai/blog?id=qwen3.6-35b-a3b.

Vishnu Vardhan Reddy, Sagnik Chatterjee, and Soumik Bhatta. Trajectory supervision for continual tool-use learning in LLMs, 2026. URL https://arxiv.org/abs/2605.09734.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2302.04761.

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.

NeoHorse Team, Guoliang Cao, Guohao Dai, Tianyu Guo, Kai Han, Hailin Hu, Zihan Jiang, Xiang Kuang, Boxun Li, Yulong Li, et al. Neohorse-1: Towards recursive self-improvement via agentic post-training with routing harness. *arXiv preprint arXiv:2609.08183*, 2026.

Miles Wang, Robi Lin, Kat Hu, Joy Jiao, Neil Chowdhury, Ethan Chang, and Tejal Patwardhan. FrontierScience: Evaluating AI's ability to perform expert-level scientific tasks. *arXiv preprint arXiv:2601.21165*, 2026. URL https://arxiv.org/abs/2601.21165.

Weichu Xie, Haozhe Zhao, Wenpu Liu, Yongfu Zhu, Liang Chen, Minghao Ye, Zirong Chen, Yuqi Xu, Shuai Dong, Ziyue Wang, Xinbo Xu, Kean Shi, Ruoyu Wu, Xiaoying Zhang, Wenqi Shao, Baobao Chang, Nan Duan, and Jiaqi Wang. Step-wise rubric rewards for LLM reasoning. *arXiv preprint arXiv:2605.17291*, 2026. doi: 10.48550/arXiv.2605.17291.

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*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Yanfei Zhang, Xu Lin, and Chenglin Wu. StepOPSD: Step-aware online preference distillation for agent reinforcement learning, 2026. URL https://arxiv.org/abs/2605.27140.

He Zhu, Junyou Su, Peng Lai, Ren Ma, Wenjia Zhang, Linyi Yang, and Guanhua Chen. Anchored supervised fine-tuning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=PORko7QT64.

A APPENDIX

A.1 BENCHMARK DETAILS

Humanity's Last Exam (HLE). Humanity's Last Exam is a multimodal benchmark consisting of 2,500 questions across mathematics, humanities, and natural sciences (Phan et al., 2025). Developed by subject experts, it provides a frontier-level test of academic competence where current LLMs still perform far below human experts.

FrontierScience-Olympiad (FS-O). FrontierScience-Olympiad evaluates expert-level scientific problem solving across physics, chemistry, and biology (Wang et al., 2026). The benchmark emphasizes structured, multi-step reasoning on challenging Olympiad-style questions with well-defined solutions.

FrontierScience-Research (FS-R). FrontierScience-Research evaluates open-ended, research-level scientific reasoning across physics, chemistry, and biology (Wang et al., 2026). Compared with short-answer scientific benchmarks, it requires models to integrate multiple facts, experimental conditions, mechanisms, and evidence-supported conclusions into a complete response.

DeepSearchQA (DSQA). DeepSearchQA evaluates long-horizon information seeking and evidence synthesis over external sources (Gupta et al., 2026). Its questions require iterative retrieval, evidence aggregation, and final answer synthesis rather than direct recall from parametric knowledge.

XBench-DeepSearch. XBench-DeepSearch evaluates agentic search, planning, reasoning, and information synthesis in realistic long-horizon tasks (Chen et al., 2025a). It measures whether an agent can effectively retrieve and integrate evidence across multiple interaction steps.

A.2 IMPLEMENTATION DETAILS

A.2.1 GRAPH-GUIDED INQUIRY DISTILLATION (GGID)

Data and training. We initialize GGID from Qwen3.5-4B and train on 1,098 outcome-correct teacher trajectories containing 24,881 interactions. The original interaction content and chronological order are fully preserved. The resulting dataset contains approximately 16.7M input tokens and 3.8M supervised tokens under the Qwen3.5 chat template, with a maximum sequence length of 49,152 tokens.

We use the selective supervision rule defined with $\tau_{\text{mask}} = 0.70$ and $\lambda_{\text{bad}} = 0$. Thus, high-confidence redundant or unproductive assistant interactions remain in the model context but contribute no imitation loss. Tool and environment observations are always treated as context and receive zero policy loss. In total, 1,278 assistant interactions are masked in the final training set.

We perform full-parameter training for one epoch with fused AdamW, a learning rate of 1×10^{-6}, cosine decay, and a warmup ratio of 0.03. The per-device batch size is 1 with gradient accumulation of 4. Training uses bfloat16 precision and gradient checkpointing. Graph annotations are produced offline by combining deterministic structural rules with semantic interaction labeling. Further details of the graph construction and annotation procedure are given in Appendix A.4.

A.2.2 COMPLETENESS-AWARE REINFORCEMENT LEARNING (CARL)

Training setup. CARL is initialized from the final GGID policy, which is also used as the frozen reference policy. We train on 256 unique scientific prompts with question-specific rubric requirements and sample $G = 4$ rollouts per prompt. Optimization uses fused AdamW with learning rate 2×10^{-8}. Rollouts are sampled with temperature 0.75 and top-p 0.95. The clipped policy objective uses $\epsilon = 0.10$, and a reference-policy KL penalty is applied with $\beta_{\text{KL}} = 0.02$. Groups with negligible reward variance are excluded from policy updates.

Rubric reward. We use a requirement-satisfaction threshold of $\gamma = 0.70$. A requirement contributes to coverage only when its item score reaches this threshold. The coverage weight is set

to $\lambda_c = 0.35$, keeping rubric-wise answer quality as the primary reward while using requirement coverage as a complementary signal. The resulting reward follows Eq. 9. Rubric items are evaluated using deterministic matching when applicable and an offline semantic evaluator for requirements that require semantic comparison. Rubrics, reference answers, item-level scores, and rewards are never exposed to the actor during rollout generation. The contribution of requirement coverage is analyzed in Appendix A.6.

Rollout control. Each rollout is limited to 2,048 policy-generated tokens and at most 12 tool calls, with 384 tokens reserved for final-answer synthesis. When the remaining generation budget reaches the reserved threshold, further tool use is disabled and the model produces its final response from the evidence collected so far. Tool observations remain visible in context but receive zero policy loss.

Controlled ablation. The *CARL w/o Coverage* variant uses the same Stage 1 initialization, training data, optimization settings, rollout configuration, and rubric-wise scorer as full CARL. Its only algorithmic change is to remove the coverage term C_i from the reward, yielding $R_i = Q_i^{\text{mean}}$. This provides a controlled comparison for isolating the contribution of requirement coverage.

A.3 Key Hyperparameters

Table 5 summarizes the main hyperparameters used in the two post-training stages.

A.4 Graph Annotation and Selective Supervision Details

Transcript-preserving graph view. For each outcome-correct teacher trajectory, we construct an operation–evidence graph $G = (V, E)$ as an annotation view over the original interaction history. The graph does not delete, reorder, or rewrite any model-visible message. Its nodes represent the question, assistant reasoning, tool calls, returned observations, extracted evidence, and the final answer, while each node remains aligned with its original message position.

Table 5: Key hyperparameters for GGID and CARL.

Setting	GGID	CARL
Initialization	Qwen3.5-4B	GGID
Training examples	1,098	256
Learning rate	1×10^{-6}	2×10^{-8}
Group size G	–	4
τ_{mask}	0.70	–
λ_{bad}	0	–
γ	–	0.70
λ_c	–	0.35
KL coefficient	–	0.02
Max rollout tokens	–	2,048
Max tool calls	–	12

Structural relations such as chronological order, tool-call/observation pairing, and source identity are recovered deterministically. Additional semantic relations capture whether an interaction contributes to evidence acquisition, discovery, recovery, verification, or later decision making. The resulting graph is used only to determine supervision weights and is not provided to the policy at inference time.

Deterministic structure extraction. Tool names and arguments are normalized before structural matching. Equivalent search aliases are mapped to a common search operation, while page reading aliases are mapped to a common visit operation. Tool calls and observations are paired using interaction and call identifiers, and exact source identifiers such as URLs are used to recover deterministic provenance relations. The original messages remain unchanged and accessible through their trajectory indices.

Semantic interaction annotation. For each interaction t, an offline semantic judge predicts a functional role z_t and confidence c_t. Useful roles include evidence acquisition, discovery, productive recovery, verification, and other reasoning that contributes to later decisions. A failed interaction may still be useful when it provides context for a subsequent recovery step.

We distinguish two negative roles for selective supervision. REDUNDANT denotes an interaction that substantially repeats previous search or reading behavior without introducing new evidence, sources, or decision-relevant information. UNPRODUCTIVE denotes an interaction that neither contributes to

Table 6: Hybrid annotation used by GGID.

Annotation	Source	Purpose
Chronological order	Deterministic	Preserve execution order
Call–observation pairing	Deterministic	Link actions and returns
Tool/source normalization	Deterministic	Match equivalent operations
Functional role z_t	Semantic judge	Estimate interaction utility
Confidence c_t	Semantic judge	Control masking threshold
Claim support / dependency	Semantic judge	Identify useful downstream links
Loss weight α_t	Deterministic	Apply selective supervision

the answer nor enables a useful downstream decision or recovery. Only high-confidence instances of these two roles are masked. Ambiguous and low-confidence cases retain their original supervision.

From annotation to supervision. The graph annotation does not modify the trajectory itself. Instead, the functional-role label and confidence are converted into the interaction weight defined in Eq. 4. With our final setting, $\tau_{\text{mask}} = 0.70$ and $\lambda_{\text{bad}} = 0$, a high-confidence REDUNDANT or UNPRODUCTIVE interaction remains in the model context but contributes no imitation loss. All other assistant interactions retain unit supervision, while tool and environment observations receive zero policy loss.

Illustrative example. Consider a teacher trajectory that asks which lexical database is integrated with the Visual Concept Ontology. An initial search discovers the relevant VCO paper, and reading the paper provides direct evidence that the linked lexical database is WordNet. A later search repeats essentially the same query without introducing new evidence and is therefore labeled REDUNDANT. In contrast, a failed exact-phrase query remains useful context because it motivates a reformulated search that independently verifies the same answer.

The corresponding useful dependency pattern can be summarized as

$$Q \rightarrow I_1^{\text{DISCOVERY}} \rightarrow I_2^{\text{EVIDENCE}} \rightarrow c_{\text{WordNet}} \rightarrow \text{FINALANSWER},$$

together with the recovery path

$$I_3^{\text{FAILEDCONTEXT}} \rightarrow I_4^{\text{VERIFICATION}} \rightarrow c_{\text{WordNet}}.$$

The redundant interaction remains at its original position in the trajectory but receives zero assistant-side loss. This example illustrates the central principle of GGID: *the interaction context is preserved, while imitation supervision is selectively masked*.

A.5 SCIENTIFIC REQUIREMENT CONSTRUCTION AND SCORING

Prompt and rubric construction. The Stage 2 training pool is derived exclusively from the 1,098 training trajectories used in Stage 1, together with their tool observations, retrieved evidence, reference answers, and offline annotations. From this pool, we construct 256 scientific questions with multiple complementary requirements, each drawn from a distinct source trajectory.

Each question is associated with three to six independently scorable rubric items and is designed to require multiple complementary answer components rather than a single short target. Rubric items cover several types of scientific requirements, including entities, numerical results, causal or mechanistic relations, experimental conditions, methods, comparisons, and evidence-supported conclusions. Every item is grounded in the corresponding training trajectory, reference answer, or retrieved evidence.

The final training set contains 256 prompts with no source-trajectory reuse. Most prompts contain five or six rubric items, providing a substantially denser supervision signal than the original short-answer objectives. The training pool spans physics, chemistry, biology, and other scientific domains without aggressively oversampling small domain subsets.

Table 7: Stage 1 design ablations. No-Graph Masking removes explicit dependency guidance while preserving the full trajectory context and uses a masking budget matched to GGID. Graph Pruning uses graph-guided interaction selection but physically removes flagged interaction blocks instead of masking their imitation loss.

Method	FS-R	FS-O	DSQA
SFT	25.00	**66.00**	71.43
No-Graph Masking	13.33	34.00	30.05
Graph Pruning	15.00	33.00	33.80
GGID	**28.33**	63.00	**73.70**

Rubric-wise scoring. For rollout τ_i, each rubric item receives a score $s_{i,k} \in [0, 1]$. Deterministic matching is used where appropriate, including entity matching and numerical comparison, while an offline semantic evaluator handles paraphrased, relational, mechanistic, and explanatory requirements. The item-level scores are aggregated into the weighted rubric quality Q_i^{mean} defined in Eq. 7.

Requirement coverage is computed with threshold $\gamma = 0.70$ as defined in Eq. 8. We use $\lambda_c = 0.35$, so that rubric-wise quality remains the primary reward while coverage provides a complementary signal. The final rollout reward follows Eq. 9. The contribution of requirement coverage is further analyzed in Appendix A.6.

End-to-end example. Consider a scientific question with multiple complementary requirements derived from a training trajectory on the Bakry–Émery curvature-dimension inequality. The corresponding rubric contains six requirements covering the inequality itself, its associated operators, the gradient estimate, its geometric interpretation, the roles of K and N, and the resulting unified conclusion.

For one rollout, the evaluator assigns

$$(s_{i,1}, \ldots, s_{i,6}) = (0.8, 0.5, 0.5, 0.7, 0.6, 0.7).$$

With equal item weights,

$$Q_i^{\text{mean}} = 0.6333.$$

Three of the six rubric items satisfy the requirement threshold $\gamma = 0.70$, yielding

$$C_i = 0.5.$$

The resulting CARL reward is

$$R_i = 0.65(0.6333) + 0.35(0.5) = 0.5867.$$

This example illustrates how CARL distinguishes partial correctness from requirement coverage within a single scientific response.

A.6 Additional Analysis

We provide additional analyses of GGID and CARL using stored predictions and evaluation outputs. No additional training is performed for the results in this section. Unless otherwise stated, GGID denotes the selected Stage 1 checkpoint, while CARL denotes the selected Stage 2 checkpoint.

A.6.1 Ablating Dependency Guidance and Context Preservation

We further analyze the two design choices underlying GGID: dependency-guided interaction selection and preservation of the complete trajectory context. Table 7 compares GGID with standard SFT and two controlled Stage 1 variants.

Dependency guidance. No-Graph Masking preserves the complete interaction history but removes explicit operation–evidence dependency modeling from the annotation process. To control for the amount of suppressed supervision, we match its masking budget to GGID. As shown in Table 7, No-Graph Masking reaches 13.33 on FS-R, 34.00 on FS-O, and 30.05 on DSQA, substantially below GGID on all three benchmarks. This result indicates that the benefit of GGID cannot be explained simply by masking a fixed amount of supervision. Rather, explicitly modeling dependencies helps determine which interactions should not be directly imitated.

Table 8: FS-R rubric-level analysis. Mean rubric uses the available official aggregate scores, while recovered coverage statistics use only examples whose item-level awards can be consistently parsed.

Method	Mean Rubric	Recovered Coverage	Satisfied Items	Fully Complete (%)
GGID	4.20	29.34	3.23	0.00
CARL w/o Coverage	4.35	**34.31**	**3.56**	0.00
CARL	**4.46**	33.60	3.49	**2.70**

Context preservation. Graph Pruning retains graph-guided interaction selection but removes flagged interaction blocks from the training trajectory instead of preserving them as context. It obtains 15.00 on FS-R, 33.00 on FS-O, and 33.80 on DSQA, again substantially below GGID. This degradation suggests that interactions judged undesirable for direct imitation may still contain observations, failed attempts, or intermediate information that supports subsequent decisions. GGID therefore separates what the model should *observe* from what it should *imitate*: flagged interactions remain in the chronological context but contribute no direct imitation loss.

Overall comparison. Standard SFT supervises every teacher interaction and obtains 25.00 on FS-R, 66.00 on FS-O, and 71.43 on DSQA. GGID improves FS-R and DSQA to 28.33 and 73.70, respectively, while reaching 63.00 on FS-O. Together with the two controlled variants, these results show that neither undirected selective masking nor physical trajectory pruning reproduces the behavior of GGID. The strongest Stage 1 performance is obtained by combining dependency-guided interaction selection with context-preserving loss masking.

A.6.2 Rubric-Level Answer Analysis

We examine rubric-level behavior on FS-Research. The official evaluator stores an aggregate rubric score together with item-level textual judgments, but does not provide a structured item-score vector. We therefore recover item-level scores only when all item awards can be unambiguously parsed and their sum matches the corresponding official total. This yields 39, 43, and 37 recoverable examples for GGID, CARL w/o Coverage, and CARL, respectively. Because these subsets are incomplete and not identical across models, the recovered item-level statistics are used only for exploratory diagnosis rather than as full-benchmark estimates.

As shown in Table 8, CARL achieves the highest official mean rubric score and is the only variant with a non-zero fully complete rate on the recoverable subset. Compared with GGID, it also increases the recovered fraction of satisfied requirements. CARL w/o Coverage obtains a slightly higher recovered coverage value than CARL; however, because these statistics are computed over different recoverable subsets, we do not draw a causal conclusion from this difference. Instead, the controlled benchmark ablation in Table 2 provides the primary evidence for the contribution of requirement coverage.

CARL achieves the highest official mean rubric score and is the only variant with a non-zero fully complete rate on the recoverable subset. Compared with GGID, it also increases the recovered fraction of satisfied rubric requirements. The *CARL w/o Coverage* variant obtains a slightly higher recovered coverage value on its recoverable subset. Since the item-level statistics are computed over different subsets, we avoid drawing a causal conclusion from this difference. Instead, we use the controlled benchmark ablation in Table 2 as the primary evidence for the contribution of requirement coverage, where the full CARL consistently improves task-level performance over the matched *w/o Coverage* variant.

A.6.3 Discipline-wise Breakdown

We next examine whether the gains of CARL are concentrated in a single scientific domain. Table 9 reports biology, chemistry, and physics accuracy on FS-Research and FS-Olympiad.

Relative to the controlled *CARL w/o Coverage* variant, full CARL improves performance across biology, chemistry, and physics on both FrontierScience tracks. The aggregate gain is therefore not driven solely by a single scientific domain. The individual subsets are nevertheless small, particu-

Table 9: Discipline-wise accuracy (%) on the biology, chemistry, and physics subsets of FrontierScience-Research (FS-R) and FrontierScience-Olympiad (FS-O).

Method	FS-R			FS-O		
	Bio	Chem	Phy	Bio	Chem	Phy
GGID	30.00	20.00	35.00	60.00	72.50	56.00
Rubric-wise GRPO	15.00	25.00	30.00	50.00	75.00	54.00
CARL w/o Coverage	15.00	30.00	25.00	40.00	67.50	56.00
CARL	25.00	**35.00**	30.00	**60.00**	**80.00**	**62.00**

larly the biology subset of FS-Olympiad, so we treat the discipline-level results as diagnostic rather than as statistically stable estimates of domain-wide performance.

For FS-Olympiad, a small number of GGID examples do not have valid judge outputs. The main paper follows a fixed denominator for aggregate reporting, while scored-only and common-subset results are retained for diagnostic comparisons.

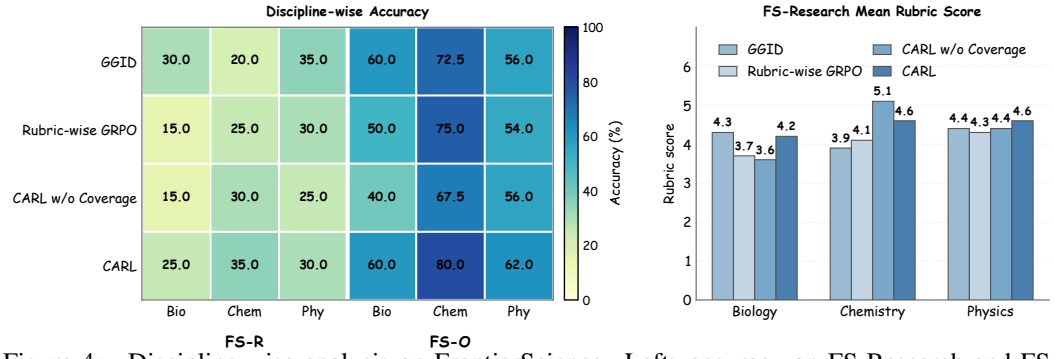


Figure 4: Discipline-wise analysis on FrontierScience. Left: accuracy on FS-Research and FS-Olympiad. Right: mean rubric score on FS-Research.

A.7 Training Dynamics

Figure 5 compares three CARL checkpoints at approximately 25%, 50%, and 100% coverage of the Stage 2 training set. Both FrontierScience performance and the FS-Research mean rubric score peak at the intermediate checkpoint. Performance declines with further optimization, suggesting that the main benefits of CARL emerge early in training. We therefore select the 50% checkpoint using the development suite.

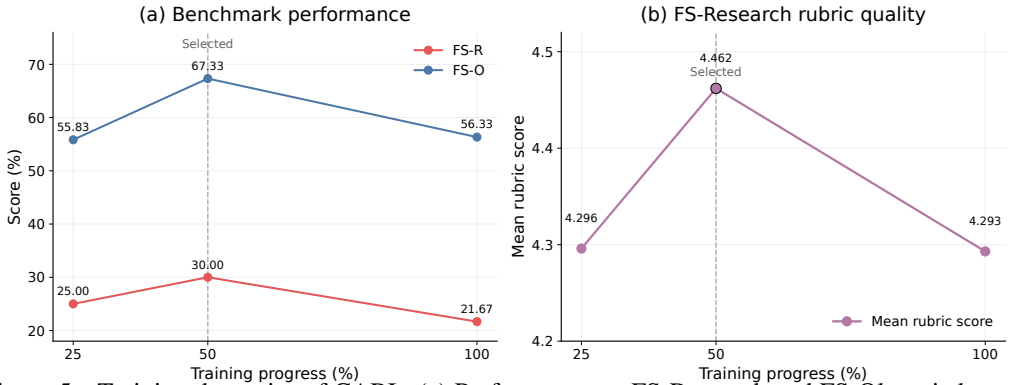


Figure 5: Training dynamics of CARL. (a) Performance on FS-Research and FS-Olympiad across training checkpoints. (b) Mean FS-Research rubric score. The selected model corresponds to the intermediate checkpoint at 50% prompt coverage.