DR.CREDIT: RUBRIC-GROUNDED PROCESS CREDIT ASSIGNMENT FOR DEEP RESEARCH AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Rubric-based tasks are increasingly addressed through reinforcement learning (RL), with rubric scores used as training rewards. However, these rewards typically supervise final answers without distinguishing the contributions of intermediate decisions. Many existing credit assignment methods rely on ground-truth answers to define process rewards, limiting their applicability to open-ended tasks without canonical solutions. To address this limitation, the proposed rubric-grounded credit uses task requirements as a shared reference for final answer evaluation and process supervision. The information returned by tools is assessed for the additional support it provides toward satisfying each rubric relative to that rubric's history of accepted support. By referencing these histories, credit distinguishes new support from evidence already present in the trajectory while recognizing partial support for each rubric. Dr.Credit uses rubric-grounded credit to supervise intermediate tool turns in an RL framework for deep research agents. The resulting process advantages are combined with GRPO outcome advantages to guide research decisions while retaining supervision of final-report quality. Evaluations on four in-domain and out-of-domain benchmarks show that Dr.Credit outperforms the evaluated open deep research baselines on every primary metric and submetric. Meanwhile, with an 8B-parameter backbone, the trained agent achieves average performance competitive with the evaluated frontier proprietary models. Further analyses suggest more efficient evidence acquisition and higher-quality reports under limited research-turn budgets, motivating the extension of rubric-grounded process supervision to a broader range of rubric-based tasks.

1 INTRODUCTION

Real-world applications of language models increasingly involve open-ended generation tasks, including novel and screenplay writing (Wu et al., 2025), medical consultation (Arora et al., 2025), and deep research. Response quality in these settings is difficult to assess through exact answer matching, as evaluation requires nuanced judgments across multiple dimensions. Task-specific rubrics make these evaluation requirements explicit by specifying the expected content and qualities of a satisfactory response (Sharma et al., 2026). Rubrics as Rewards (RaR) (Gunjal et al., 2026) extends this evaluation approach to reinforcement learning (RL) by turning task-specific requirements into training rewards. In deep research, DR Tulu (Shao et al., 2026) and DeepRubric (Zhu et al., 2026a) train agents with rubric-based rewards, directly optimizing report quality against these requirements.

Recent work on rubric-based RL has focused on improving rubric quality to provide more reliable supervision for final answers (Shao et al., 2026; Zhu et al., 2026a; Xie et al., 2026a), but leaves credit assignment across intermediate turns unresolved. When these turns share an outcome advantage, a high-scoring answer can reinforce ineffective decisions, while later synthesis errors can cause useful intermediate steps to be penalized (Figure 1(a)). Many existing credit assignment methods use provided ground-truth answers as a reference for assigning intermediate rewards (Wang et al., 2026; Zhu et al., 2026b; Lu et al., 2026) (Figure 1(b)). This dependence limits direct transfer to open-ended tasks where rubrics specify what a satisfactory response should accomplish without prescribing a canonical answer to serve as a reference for turn-level credit assignment.

To address this challenge, the key is to identify a reference for evaluating intermediate turns without relying on a canonical answer. The rubric set used to evaluate the final answer offers a natural refer-

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.

ence, since it defines task-specific requirements that remain applicable across different high-quality responses. Our core idea is to derive process credit from the additional support that newly obtained information provides toward satisfying each rubric. We maintain a separate support history for each rubric to assess these contributions and aggregate them into turn-level credit (Figure 1(c)). The credits are normalized into process advantages and combined with GRPO outcome advantages, extending rubric-based supervision to intermediate decisions while retaining final-answer evaluation.

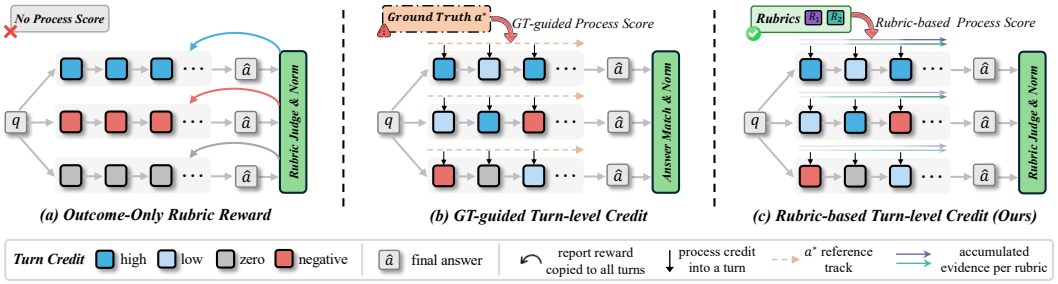


Figure 1: **Credit assignment for rubric-based agent training.** (a) Intermediate turns share an outcome advantage. (b) Ground-truth answers guide turn-level credit. (c) Task rubrics ground turn-level credit in additional evidence support while outcome supervision is retained. In (c), each rubric's track darkens as evidence accumulates across turns. Turn colors denote normalized advantages.

This work applies these rubric-grounded credit assignment and optimization mechanisms to deep research, a representative open-ended task with rubric-based evaluation. The training framework is named Dr.Credit, which uses task-specific rubrics to assign process credit to intermediate tool-use turns. Turns that extract webpage content are credited for the additional support they provide relative to the existing evidence history for each rubric. These credits are then attributed to earlier search turns that returned the corresponding URLs. Search turns can also receive supplementary credit through an independent assessment of rubric support in snippets from results whose webpage content is not subsequently extracted. The resulting credits enter the preceding policy optimization scheme, providing turn-level supervision alongside the reward for the final report.

To evaluate the effectiveness of Dr.Credit, we conduct extensive experiments on both in-domain and out-of-domain benchmarks with deep research agents. Results show that Dr.Credit consistently outperforms the evaluated open deep research baselines, suggesting good generalization across benchmarks. Our main contributions are as follows: (1) We propose rubric-grounded credit, a general approach to process reward design for agent RL on open-ended tasks with rubric-based evaluation. It extends task rubrics from final-answer evaluation to turn-level credit assignment by assessing additional support relative to per-rubric histories, without requiring a canonical answer. (2) We apply this approach to deep research and introduce Dr.Credit, a training framework that assigns credit to intermediate research tool turns and integrates the resulting process advantages with GRPO outcome advantages. (3) With an 8B backbone, Dr.Credit achieves average performance competitive with the evaluated frontier proprietary models. Further analyses indicate that Dr.Credit acquires evidence more efficiently and produces higher-quality reports under limited research-turn budgets.

2 Preliminaries

2.1 Problem Formulation

Given an open-ended question q, a policy π_θ generates a rollout $O = (\tau_0, \ldots, \tau_T)$ through successive rounds of reasoning and interaction with an external environment. Each tool-interaction turn τ_t ($t < T$) follows the ReAct paradigm (Yao et al., 2023), comprising `[think]`, `[tool call]`, and `[tool response]`. Conditioned on the question and the preceding interaction history, the policy generates the reasoning and tool call, while the environment supplies the observation returned by its execution. The sampled rollout O_i at the bottom of Figure 2 illustrates how each returned observation is appended to the context, where information gathered so far remains available to subsequent reasoning and tool use. At the final turn τ_T, `[think]` is followed by `[answer]`, yielding a final answer $\hat{a}$ whose quality is then evaluated against the requirements specified by the task's rubric set.

2.2 RUBRIC-BASED REINFORCEMENT LEARNING

Rubrics as Rewards (RaR) (Gunjal et al., 2026) turns rubric-based evaluations of final answers into reward signals for training the policy π_θ through reinforcement learning. For a question q, the task-specific rubric set $\mathcal{R}_q$ contains K rubrics, with a nonnegative weight w_k assigned to each rubric and $\sum_{k=1}^{K} w_k > 0$. An LLM judge evaluates the final answer $\hat{a}$ against each rubric, assigning a satisfaction score $s_k(q, \hat{a}) \in [0, 1]$; the weighted average of these scores provides the rubric reward:

$$r_{\text{rubric}}(q, \hat{a}) = \frac{\sum_{k=1}^{K} w_k s_k(q, \hat{a})}{\sum_{k=1}^{K} w_k}. \tag{1}$$

Group Relative Policy Optimization (GRPO) (Shao et al., 2024) converts rollout-level outcome rewards into advantages by normalizing rewards across rollouts sampled for the same question. Since each advantage is shared by all policy-generated tokens within a rollout, outcome supervision does not explicitly distinguish the contributions of intermediate turns toward satisfying the task's rubrics.

3 METHODOLOGY

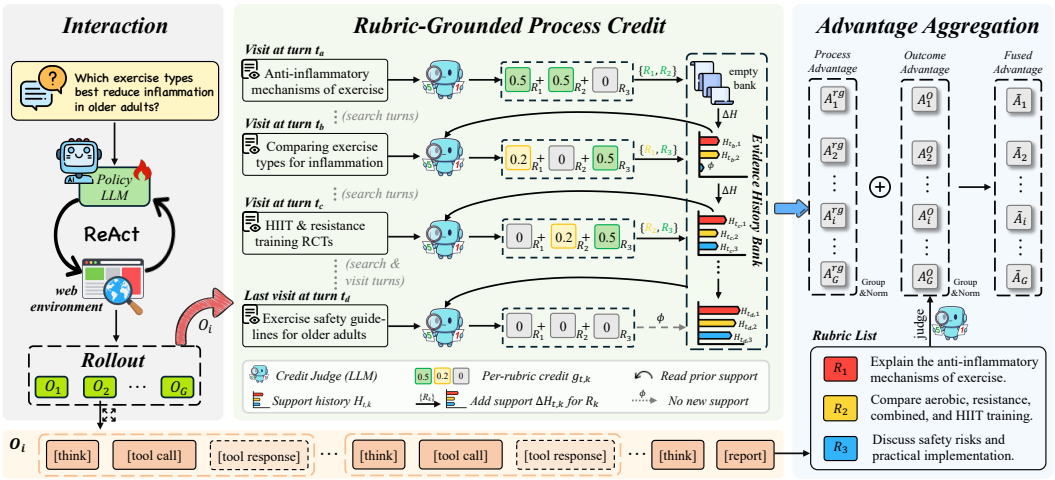


Figure 2: **Overall framework of Dr.Credit**, which incorporates rubric-grounded process credit into reinforcement learning for deep research agents. **Left:** the policy generates a group of rollouts for each query through ReAct interactions with the web environment. **Center:** an LLM judge assesses the additional support each tool turn provides relative to per-rubric evidence histories within its rollout, then updates the histories with accepted support points. **Right:** process credits are normalized across research turns in the rollout group and added to GRPO outcome advantages. **Bottom:** an example rollout O_i. Black dashed boxes mark tool responses masked out of the loss.

Rubrics can support fine-grained process rewards in reinforcement learning, extending their use beyond evaluating final outputs. However, tool turns cannot be judged by rubric satisfaction in the same way as final answers. Their contributions lie in the information they provide toward satisfying each rubric, but assessing it in isolation can repeatedly credit support established earlier. We therefore ground process credit in the evidence accumulated for each rubric, maintaining support histories against which an LLM judge evaluates additional support from newly obtained information.

Following this principle, we introduce Dr.Credit, a rubric-based credit assignment framework for deep research agents. As illustrated in Figure 2, Dr.Credit assigns credit to tool turns based on their contributions toward satisfying the task's rubrics and combines the resulting process advantages with GRPO outcome advantages for policy optimization. We first formalize the underlying rubric-grounded process credit and the maintenance of support histories (Section 3.1), then describe how the resulting credits enter policy optimization (Section 3.2). Section 3.3 presents Dr.Credit as a concrete instantiation of these mechanisms for a basic deep research agent.

3.1 RUBRIC-GROUNDED PROCESS CREDIT

Rubric-Grounded Evidence Support. For a question q with rubric set $\mathcal{R}_q$, let e_t denote the information in the environment's response at tool turn t of rollout O. For each rubric, let $\mathcal{S}_k(q)$ be the space of evidence-grounded support statements for the k-th rubric, including facts, premises, relations, and qualifications that help fulfill its requirement without prescribing a canonical answer or an action sequence. While $s_k(q, \hat{a})$ measures how well the final answer satisfies the rubric, process assessment identifies the support that e_t establishes toward meeting the same requirement.

Per-Rubric Support Histories. To distinguish new support from previously established information, we maintain a history $H_{t,k} \subseteq \mathcal{S}_k(q)$ containing the support points accepted for the k-th rubric before turn t. These histories start empty, $H_{0,k} = \varnothing$, and evolve independently within each rollout. Their collection, $\mathcal{H}_t = (H_{t,1}, \ldots, H_{t,K})$, forms the history bank in Figure 2, preserving prior support as the rubric-specific reference for assessing subsequent information.

History-Aware Credit Assignment. The assessor $\mathcal{J}$ compares the current information e_t with histories $\mathcal{H}_t$, evaluating additional support per rubric and identifying the support increments:

$$(\mathbf{g}_t, \Delta\mathcal{H}_t) = \mathcal{J}(q, \mathcal{R}_q, e_t, \mathcal{H}_t). \tag{2}$$

Here, $\mathbf{g}_t = (g_{t,k})_{k=1}^K$ contains nonnegative per-rubric contributions, and $\Delta\mathcal{H}_t = (\Delta H_{t,k})_{k=1}^K$ contains the corresponding support increments, with $\Delta H_{t,k} \subseteq \mathcal{S}_k(q)$. Each $g_{t,k}$ measures the support that e_t adds relative to $H_{t,k}$, with zero assigned when no additional support is established; the sum of these contributions defines the rubric-grounded credit for the current turn as

$$c_t^{rg} = \sum_{k=1}^{K} g_{t,k}. \tag{3}$$

Once the contributions have been assessed against the existing histories, the accepted support increments update the reference available to subsequent turns through the per-rubric union

$$H_{t+1,k} = H_{t,k} \cup \Delta H_{t,k}. \tag{4}$$

Assessment reads the history before the update, so the current evidence is compared with prior support before its accepted points enter the history; an empty increment leaves that history unchanged.

3.2 POLICY OPTIMIZATION WITH PROCESS CREDIT

Process credits measure additional rubric support from each turn, while the outcome reward evaluates the final answer. To use both signals in policy optimization, the old policy π_{old} samples $G > 1$ rollouts for each question–rubric pair $(q, \mathcal{R}_q)$ in the training set $\mathcal{D}$. Let $c_{i,t}$ be the process credit for non-final turn $t < T_i$ of rollout O_i, where T_i indexes its final turn, and let r_i denote its rollout-level outcome reward. We normalize both signals separately within each rollout group to obtain

$$A_{i,t}^{rg} = \frac{c_{i,t} - \mu^{rg}}{\sigma^{rg}}, \qquad A_i^O = \frac{r_i - \mu^O}{\sigma^O}. \tag{5}$$

The process statistics μ^{rg} and σ^{rg} are the mean and population standard deviation over all non-final policy-turn credits in the rollout group, including zero-credit turns, with equal weight per turn regardless of the length of its rollout. The outcome statistics μ^O and σ^O retain the GRPO mean and sample standard deviation over the G rollout rewards obtained for the same question.

These separately normalized advantages provide the two inputs to the fusion step on the right of Figure 2, where the process term supplements the outcome advantage on each non-final turn:

$$\widetilde{A}_{i,t} = \begin{cases} A_i^O + A_{i,t}^{rg}, & 0 \le t < T_i, \\ A_i^O, & t = T_i. \end{cases} \tag{6}$$

To apply this turn-level signal to token-level updates, let $\mathcal{P}_i$ contain the policy-generated token positions in O_i, and let $t(i,j)$ identify the turn containing token $x_{i,j}$. All tokens in that turn share $\widetilde{A}_{i,t(i,j)}$, while the policy ratio remains $\rho_{i,j}(\theta) = \frac{\pi_\theta(x_{i,j}|q,x_{i,<j})}{\pi_{\text{old}}(x_{i,j}|q,x_{i,<j})}$, where $x_{i,<j}$ includes earlier

policy tokens and environment observations. Substituting $\widetilde{A}_{i,t(i,j)}$ into the GRPO surrogate gives the following objective, averaged over all policy-generated tokens in the sampled rollout group

$$\mathcal{J}_{\text{Dr.Credit}}(\theta) = \mathbb{E}_{(q,\mathcal{R}_q)\sim\mathcal{D},\,\{O_i\}\sim\pi_{\text{old}}(\cdot|q)}\Bigg[\frac{1}{\sum_{i=1}^{G}|\mathcal{P}_i|}\sum_{i=1}^{G}\sum_{j\in\mathcal{P}_i}\min\Big(\rho_{i,j}(\theta)\widetilde{A}_{i,t(i,j)},\\ \text{clip}\big(\rho_{i,j}(\theta),1-\epsilon,1+\epsilon\big)\widetilde{A}_{i,t(i,j)}\Big) - \beta\mathbb{D}_{\text{KL}}(\pi_\theta\|\pi_{\text{ref}})\Bigg]. \tag{7}$$

Here, ϵ controls probability-ratio clipping, and the KL term denotes token-averaged regularization against reference policy π_{ref}, with coefficient β. The dashed masks in Figure 2 exclude environment observations from the loss while retaining them in the policy's conditioning context.

3.3 Rubric-Grounded Credit for Deep Research

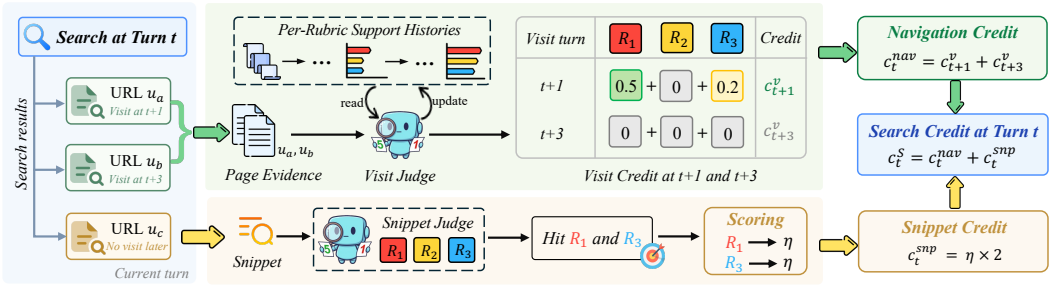


Figure 3: **Composite credit for `Search` turns in Dr.Credit. Upper green branch:** history-aware Visit credits are attributed to the search through URL matching. **Lower yellow branch:** an independent judge matches snippets from results without later visits to rubrics, without reading or updating Visit histories. Navigation and snippet credits are added to obtain Search credit.

Dr.Credit instantiates the preceding credit assignment and policy optimization mechanisms for a basic deep research agent equipped with only two common tools. `Search` discovers candidate sources and returns snippets, while `Visit` extracts the content of selected webpages. `Visit` provides the main body of evidence for the final report and is the primary focus of our credit assessment. `Search` receives credit for discovering sources that yield additional support and for evidence available directly in its snippets, as illustrated in Figure 3. The resulting Visit and Search credits, denoted c^v_t and c^s_t, provide $c_{i,t}$ for policy optimization in Section 3.2.

For a `Visit` turn, the page evidence serves as e_t. An LLM judge implements $\mathcal{J}$ by assessing the additional support this evidence provides relative to each rubric's history $H_{t,k}$ and assigning a contribution score $g_{t,k}$. These per-rubric scores are aggregated into Visit credit c^v_t following the summation in Equation 3. For positive assessments, the judge extracts support points in the current page to form $\Delta H_{t,k}$, which updates the corresponding history after scoring according to Equation 4.

For a `Search` turn, we match normalized result URLs to later visits within the same rollout and aggregate the matched Visit credits into navigation credit c^{nav}_t. The upper branch of Figure 3 illustrates this attribution for visits to u_a and u_b. Results without later visits enter the complementary snippet branch, where an independent LLM judge identifies rubric support directly in their snippets. Each accepted rubric match contributes η to snippet credit c^{snp}_t; in the illustrated example, the snippet from u_c supports R_1 and R_3, yielding two such contributions. The brief, scattered information in snippets motivates coarse rubric-level matching without reading or updating Visit histories. Navigation and snippet credits are combined additively into Search credit c^s_t.

4 Experiments

4.1 Experimental Setup

Datasets & Metrics. DeepRubric Zhu et al. (2026a) provides our training data, pairing research queries with evidence-grounded rubric sets; we randomly hold out 128 examples for in-domain

Table 1: Overall performance on four deep research benchmarks. Dr.Credit outperforms all open deep research baselines on every metric, including all submetrics, and is competitive with proprietary systems. Average is the unweighted mean of the four benchmark-level scores. Bold denotes the best scores across the lower three model groups. * and † indicate results taken from DeepRubric (Zhu et al., 2026a) and ResearchRubrics (Sharma et al., 2026), respectively.

Methods	DeepRubric (val)			ResearchQA	DeepResearchBench					ResearchRubrics	Average
	Overall	Factual	Logical		Overall	Comp.	Depth	Instr.	Read.		
Frontier Proprietary Models											
Kimi K3 + Our Tools	71.5	55.5	90.2	76.7	48.4	47.2	47.5	50.2	49.5	58.0	63.6
GPT-5.6-luna + Our Tools	73.2	55.6	93.9	73.2	48.8	47.3	48.4	50.4	49.8	56.6	63.0
DeepSeek-V4-Pro + Our Tools	68.4	52.7	87.0	78.0	44.5	43.5	43.7	46.1	45.6	50.8	60.4
Opus 4.8 + Our Tools	71.0	55.2	89.7	72.9	45.5	43.6	44.8	47.7	47.6	51.5	60.2
GLM-5.3 + Our Tools	68.9	54.2	85.9	72.1	45.6	44.4	45.5	46.1	46.0	52.5	59.8
Perplexity Deep Research	–	–	–	75.3*	42.3*	40.7*	39.3*	46.4*	44.3*	48.7†	–
Gemini Deep Research	–	–	–	68.5*	48.8*	48.5*	48.5*	49.2*	49.4*	61.5†	–
OpenAI Deep Research	–	–	–	79.2*	46.9*	46.8*	45.2*	49.2*	47.1*	59.7†	–
Naive RAG											
Qwen3-8B + RAG	31.6	20.9	44.2	46.0	22.3	19.0	14.3	32.8	26.6	26.2	31.5
Qwen3.5-9B + RAG	42.5	25.4	63.4	47.2	26.1	22.5	18.6	36.1	31.3	32.9	37.2
Qwen3.6-35B-A3B + RAG	57.7	38.1	81.0	59.1	36.7	33.7	32.3	44.3	39.3	43.6	49.3
Open Deep Research Models											
ASearcher-Web-7B	12.3	7.8	16.9	19.4*	7.8*	5.1*	1.7*	15.2*	11.8*	7.6	11.8
Search-R1-7B	10.3	7.2	14.3	27.9*	9.5*	5.2*	2.1*	18.6*	16.8*	5.2	13.2
WebExplorer-8B	45.0	37.0	53.3	64.8*	36.7*	33.7*	28.5*	45.7*	42.2*	29.5	44.0
Tongyi DeepResearch-30B-A3B	57.3	44.3	72.4	66.7*	40.6*	39.1*	34.3*	46.8*	45.4*	37.1	50.4
Qwen3-8B-Based Deep Research Agents											
Qwen3-8B + Our Tools	40.5	30.0	52.8	48.1	28.8	26.6	23.3	35.4	33.0	33.0	37.6
DeepRubric-8B + Our Tools	62.5	43.6	85.6	73.5	41.6	39.7	39.1	45.4	44.3	43.1	55.2
DR Tulu-8B + Our Tools	60.4	41.7	83.3	72.9	42.9	41.2	42.4	45.6	43.2	45.8	55.5
Qwen3-8B-SFT	59.3	45.2	75.3	69.2	40.3	39.3	34.6	45.4	41.8	40.9	52.4
Qwen3-8B-GRPO	67.4	52.5	84.0	75.4	43.2	42.1	40.9	46.8	44.0	46.8	58.2
Dr.Credit (Ours)	**70.6**	**54.0**	**90.0**	**79.8**	**46.1**	**45.1**	**44.7**	**48.7**	**46.7**	**50.3**	**61.7**

validation. Out-of-domain evaluation covers three research benchmarks: ResearchQA Yifei et al. (2026) for scholarly question answering, DeepResearchBench Du et al. (2026) for comprehensive report generation, and ResearchRubrics for open-ended tasks with expert-written rubrics. We report weighted rubric satisfaction on the DeepRubric validation set, rubric coverage on ResearchQA, weighted rubric compliance on ResearchRubrics, and report-quality scores on DeepResearchBench, all on a 100-point scale. Data construction and evaluation settings are detailed in Appendix A.

Baselines. Our baselines cover four groups: (1) frontier models equipped with our tools, including Kimi K3 Team et al. (2026), DeepSeek-V4-Pro Xu et al. (2026), and GLM-5.3 Zeng et al. (2026), together with commercial deep research services; (2) Qwen-based naive RAG Yang et al. (2025); (3) open search agents, including ASearcher Gao et al. (2026), Search-R1 Jin et al. (2025), WebExplorer Liu et al. (2025), and Tongyi DeepResearch Team et al. (2025); and (4) open deep research agents built on Qwen3-8B, including DeepRubric-8B Zhu et al. (2026a), DR Tulu-8B Shao et al. (2026), and our SFT and GRPO baselines, with the base Qwen3-8B agent included as a reference. The GRPO baseline uses outcome rewards without process credit and shares the same SFT initialization, training data, outcome reward, training budget, and evaluation setup as Dr.Credit. Baseline evaluation settings and result provenance are detailed in Appendix A.3.

Implementation Details. For credit assignment (Section 3.3), `Visit` assessments of no additional support, partial new support, and high-value new evidence correspond to $g_{t,k} = 0$, 0.2, and 0.5, respectively, while each accepted rubric match in a `Search` snippet contributes $\eta = 0.1$ to c_t^{snp}. To keep turn-level credits on a shared scale before process-advantage normalization, the sums defining Visit credit c_t^v, navigation credit c_t^{nav}, and Search credit c_t^s are capped at 1. An offline analysis in Appendix B.3 reports similar normalized process advantages across alternative parameter settings.

With these credit settings, RL starts from a Qwen3-8B checkpoint obtained after two epochs of SFT on 1,505 filtered research trajectories. Dr.Credit is then trained on 7,149 query–rubric pairs for 200 steps, with 32 queries per batch and $G = 8$ rollouts per query. For both process and outcome supervision, Qwen3.6-35B-A3B performs history-aware `Visit` assessment and independent `Search` snippet matching, as well as evaluating rubric satisfaction and citation quality in final reports. The corresponding judge prompts are provided in Appendix D, while Appendix A documents data construction, training and interaction settings, and the definitions of each outcome reward component.

4.2 Overall Performance

Table 1 shows that Dr.Credit is competitive with frontier models, surpassing DeepSeek-V4-Pro, Opus 4.8, and GLM-5.3 in average score while approaching Kimi K3 and GPT-5.6-luna. With an 8B backbone, Dr.Credit achieves an average score of 61.7 and outperforms all models in the Naive RAG and Open Deep Research Models groups across every metric and submetric, including larger models such as Qwen3.6-35B-A3B and Tongyi DeepResearch-30B-A3B.

Among deep research agents built on the same Qwen3-8B base model, Dr.Credit achieves the highest scores on all four benchmarks and their submetrics. The results for DeepRubric-8B and DR Tulu-8B in Table 1 are from our evaluations using the same web search and web visit tools as Dr.Credit (see Appendix A.3). Under matched training and evaluation settings, Dr.Credit improves over the GRPO baseline on all four benchmarks, with a 3.5-point average gain from adding process credit. Beyond the in-domain DeepRubric validation set, Dr.Credit achieves clear gains on all three external benchmarks, suggesting that our credit assignment framework improves out-of-domain generalization.

4.3 Ablation Study

To understand which design choices contribute to Dr.Credit's performance, we conduct three groups of ablation experiments. Specifically, we evaluate the process credit design, the benefits of combining outcome and process supervision, and the contribution of per-rubric evidence history.

Table 2: **Ablation of process credit designs.** Visit denotes history-aware credit for Visit turns; gray shading marks the final Dr.Credit configuration. DRub, RQA, DRB, and RR denote DeepRubric (val), ResearchQA, DeepResearchBench, and ResearchRubrics, respectively. Avg is the arithmetic mean of the four benchmark scores; **bold** marks the best score per benchmark and the highest Avg.

Visit	Navigation	Snippets	Discount	DRub	RQA	DRB	RR	Avg
✓	✗	✗	✗	68.8	76.8	44.7	48.6	59.7
✓	✓	✗	✗	**70.8**	78.4	45.0	47.2	60.4
✓	✓	✓	✗	70.6	**79.8**	**46.1**	**50.3**	**61.7**
✓	✓	✗	✓	68.2	77.3	44.5	45.9	59.0
✓	✓	✓	✓	68.0	78.0	45.3	48.1	59.9

Process Credit Design. Table 2 reports the performance of agents trained with different process credit designs. Relative to Visit-only credit, adding navigation attribution from subsequent Visits improves scores on DeepRubric, ResearchQA, and DRB, while reducing the score on RR. Adding independent snippet scoring for unvisited search results further improves scores on ResearchQA, DRB, and RR, yielding the highest four-benchmark average of 61.7, with only a 0.2-point decrease on DeepRubric. These comparisons support assigning credit to both source discovery and evidence available directly in search snippets. Discounted propagation with $\gamma = 0.95$ lowers scores on all four benchmarks, both with and without snippet scoring, relative to the corresponding variants without propagation. Together, these results support the final process credit design adopted in Dr.Credit.

Outcome and Process Supervision. As shown in Table 3(a), removing the process advantage A^{rg} reduces Dr.Credit to GRPO and lowers the three-benchmark average by 3.5 points. This gap demonstrates the value of rubric-grounded process credit as additional supervision for research turns. However, relying on process supervision alone for these turns also lowers the average by 2.3 points. These results suggest complementary roles for the two signals: process credit guides evidence acquisition at individual turns, while outcome supervision helps align these decisions with the overall quality of final reports.

Table 3: Ablations of Dr.Credit: (a) removing either outcome or process advantage from research-turn supervision; (b) removing evidence history when computing process credit. Each ablation is relative to full Dr.Credit. Values are score changes, with ΔAvg_3 averaged across the three benchmarks.

Removed	ΔDRub	ΔRQA	ΔDRB	ΔAvg_3
(a) Outcome and Process Supervision				
Process A^{rg}	-3.2	-4.4	-2.9	-3.5
Outcome A^{O}	-1.2	-3.6	-2.1	-2.3
(b) Evidence History				
History	-3.0	-3.9	-2.1	-3.0

Effect of Evidence History. Table 3(b) shows that computing process credit without evidence history lowers scores on all three benchmarks, with an average decrease of 3.0 points. A page can support a rubric without extending the evidence already available, so scoring it in isolation risks rewarding repeated retrieval. The paired scoring analysis in Appendix B.2 compares Visit credits and rubric-level judgments with and without prior support on fixed trajectories. These diagnostics complement the training ablation, suggesting that per-rubric histories help supervise the additional evidence acquired at each turn while accounting for support already available within the trajectory.

4.4 In-Depth Analysis

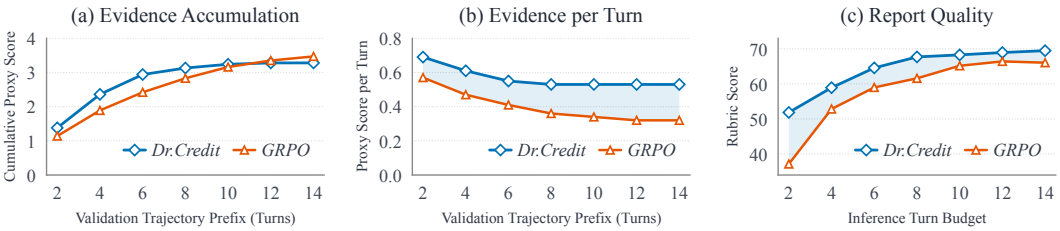


Figure 4: Evidence acquisition and report quality on DeepRubric. (a) Mean cumulative evidence proxy score. (b) Mean proxy score per actual research turn. (c) Report rubric scores under inference-time turn budgets. Blue shading in (b) and (c) indicates differences between method means.

To investigate what underlies Dr.Credit's strong performance, we analyze how it acquires evidence along research trajectories and assess report quality under constrained turn budgets. Figure 4 summarizes these analyses on the DeepRubric validation set. Evidence acquisition is tracked using a credit-based proxy that accumulates uncapped credit from retrieved pages and search snippets, excluding navigation credit. For each trajectory prefix, the proxy includes only contributions from content returned up to that point. Scoring and aggregation details are provided in Appendix B.1.

Under this proxy, Dr.Credit accumulates evidence more rapidly in early research turns, reaching a cumulative score of 2.94 within 6 turns compared with GRPO's 2.83 within 8 (Figure 4(a)). Dr.Credit leads at every measured prefix through ten turns, while GRPO slightly surpasses it at twelve and fourteen turns. Together with the decline in estimated research-turn counts during training (Appendix C.1, Figure 10(b)), these results suggest that Dr.Credit maintains effective evidence acquisition as its trajectories become shorter. Figure 4(b) reports mean per-turn proxy scores by dividing each trajectory's cumulative score by the number of research turns actually completed within the prefix and then averaging these ratios across trajectories (see Appendix B.1 for detailed calculations). Dr.Credit scores higher at every measured prefix, reaching 0.53 versus GRPO's 0.36 at eight turns (+47.2%), indicating more efficient evidence acquisition per research turn under this proxy.

Separate evaluations under inference-time turn budgets assess final-report quality (Figure 4(c)), following the report-generation protocol in Appendix B.1. Dr.Credit outperforms GRPO at all seven tested budgets. With an eight-turn budget, it achieves a rubric score of 67.69, exceeding GRPO's best observed score of 66.41 at twelve turns. Dr.Credit thus exceeds GRPO's best observed quality with a one-third smaller allowed turn budget. These results suggest that Dr.Credit follows more efficient research trajectories to produce higher-quality reports under constrained turn budgets.

4.5 Computational Efficiency

Although Dr.Credit achieves strong performance, its process-credit scoring may introduce substantial computational overhead during training. Figure 5 therefore compares the learning progress and cumulative training time of Dr.Credit and GRPO over 200 steps, including process-credit scoring.

Learning progress. As illustrated in Figure 5(a), our rubric-grounded credit assignment enables faster empirical convergence and higher final validation performance than GRPO baseline. Dr.Credit surpasses GRPO's final validation score by step 120 and finishes at step 200 with a 3.2-point advantage. The supplementary training curves in Appendix C.1 show consistent improvements.

Training time. Figure 5(b) shows an early cumulative-time disadvantage for Dr.Credit. This is expected given the additional workload of process-credit scoring, as reported in Table 7. From

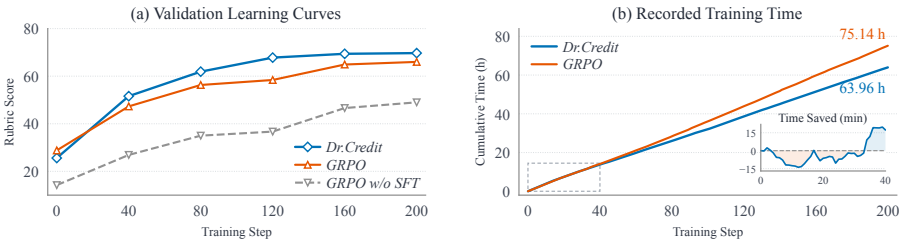


Figure 5: Learning progress and recorded training time. (a) Validation rubric scores on DeepRubric; GRPO without SFT is included as an initialization reference. (b) Cumulative recorded training-step time over 200 steps, including rollout collection, reward scoring, and policy updates. The inset shows GRPO time minus Dr.Credit time over the first 40 steps, in minutes.

step 34 onward, however, Dr.Credit maintains a lower cumulative training time than GRPO, saving 11.18 hours (14.9%) over 200 steps with scoring included. Dr.Credit learns more efficient research behavior that reduces interaction and evaluation work, while asynchronous execution in the training framework allows scoring to overlap with rollout generation. These factors help explain the time savings under the measured setup, with a detailed analysis provided in Appendix C.2.

5 Related Work

Rubric-Based Rewards for Deep Research. Instance-specific rubrics and checklists provide evaluation criteria for open-ended tasks, with recent work improving their coverage and discriminability Sharma et al. (2026); Viswanathan et al. (2025); Shen et al. (2026). In deep research, DR Tulu Shao et al. (2026) evolves rubrics using retrieved evidence and on-policy rollouts, while DeepRubric Zhu et al. (2026a) and DR-Rubric Mei et al. (2026) ground rubrics in evidence from structured expansion or agentic search. Learning Query-Specific Rubrics Lv et al. (2026) aligns rubric generation with human preferences for research reports, and QUEST Xie et al. (2026a) uses rubric trees to define rewards across research tasks. RubricEM Li et al. (2026) extends rubric supervision to intermediate research stages, assigning a shared advantage to all policy tokens within each stage.

Fine-Grained Credit Assignment. Classical temporal credit assignment uses TD(λ) Sutton (1988) to propagate prediction errors through eligibility traces; generalized advantage estimation Schulman et al. (2016) builds on this principle to balance bias and variance in policy-gradient estimation. RUDDER Arjona-Medina et al. (2019) addresses delayed rewards by redistributing returns according to changes in predicted return. For LLM agents, GiGPO Feng et al. (2025) compares actions from repeated anchor states across trajectories; HCAPO Tan et al. (2026) uses hindsight reasoning to refine step-level value estimates; and HiPER Peng et al. (2026) estimates advantages for both planning and execution. A complementary line measures progress toward a target answer: IGPO Wang et al. (2026) rewards increases in the policy's probability of generating the correct answer; TIPS Xie et al. (2026b) shapes rewards using a teacher model's answer potential; and ΔBelief-RL Auzina et al. (2026) rewards changes in the agent's belief in the target solution.

6 Conclusion

In this work, we propose rubric-grounded credit, which extends task rubrics from final-answer evaluation to process supervision. Intermediate steps receive credit for the additional support they provide relative to each rubric's evidence history, without requiring a canonical answer. Applied to deep research, this approach underlies Dr.Credit, a training framework combining process and GRPO outcome advantages to supervise research decisions and final-report quality. Evaluations across four benchmarks show that Dr.Credit outperforms the evaluated open deep research baselines and generalizes well to out-of-domain tasks. Even with only 8B parameters, the trained Qwen3-8B agent achieves average performance competitive with that of the evaluated frontier proprietary models. Beyond these benchmark results, further analyses suggest that the trained agent acquires useful evidence more efficiently and produces higher-quality reports under the same research-turn budgets. These findings support rubric-based supervision of evidence acquisition and motivate extending rubric-grounded credit to other open-ended tasks trained with rubric-based reinforcement learning.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic SFT trajectories with GLM-5.2. We also used GPT-6 Astra to translate the manuscript, clean and reformat SFT and RL prompts, and assist with prompt design for the agent components. We have not used generative AI tools to develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine research hypotheses, design experiments, conduct qualitative or thematic data analysis, or interpret experimental results, and mathematical proof-related tasks are not applicable to this work. Additionally, we used GPT-6 Astra for English-language polishing, literature search, and coding assistance. We have reviewed all AI-assisted work. We checked the generated SFT data and prompts, reviewed the translated and polished text for technical accuracy, and verified the cited literature against the original sources. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Jose A. Arjona-Medina, Michael Gillhofer, Michael Widrich, Thomas Unterthiner, Johannes Brandstetter, and Sepp Hochreiter. RUDDER: Return decomposition for delayed rewards. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32. Curran Associates, Inc., 2019.

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, et al. HealthBench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*, 2025.

Ilze Amanda Auzina, Joschka Strüber, Sergio Hernández-Gutiérrez, Shashwat Goel, Ameya Prabhu, and Matthias Bethge. Intrinsic credit assignment for long horizon interaction. In *International Conference on Machine Learning (ICML)*, 2026.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Licheng Zhang, Xiaorui Wang, and Zhendong Mao. Deepresearch bench: A comprehensive benchmark for deep research agents. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 42414–42448, 2026.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for LLM agent training. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 46375–46408, 2025.

Jiaxuan Gao, Wei Fu, Minyang Xie, Shusheng Xu, Chuyi He, Zhiyu Mei, Banghua Zhu, and Yi Wu. Unlocking long-horizon agentic search with large-scale end-to-end rl. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 107331–107352, 2026.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 127924–127945, 2026.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Gaotang Li, Bhavana Dalvi Mishra, Zifeng Wang, Jun Yan, Yanfei Chen, Chun-Liang Li, Long T Le, Rujun Han, George Lee, Hanghang Tong, et al. Rubricem: Meta-rl with rubric-guided policy decomposition beyond verifiable rewards. *arXiv preprint arXiv:2605.10899*, 2026.

Junteng Liu, Yunji Li, Chi Zhang, Jingyang Li, Aili Chen, Ke Ji, Weiyu Cheng, Zijia Wu, Chengyu Du, Qidi Xu, et al. Webexplorer: Explore and evolve for training long-horizon web agents. *arXiv preprint arXiv:2509.06501*, 2025.

Yijun Lu, Rui Ye, Jiajun Wang, Yuwen Du, Tian Jin, Songhua Liu, and Siheng Chen. Abseeker: Training long-horizon search agents via answer-backtracked credit assignment. *arXiv preprint arXiv:2608.05102*, 2026.

Changze Lv, Jie Zhou, Wentao Zhao, Jingwen Xu, Shihan Dou, Zisu Huang, Muzhao Tian, Xiaohua Wang, Zhengkang Guo, Yang Liu, et al. Learning query-specific rubrics from human preferences for DeepResearch report generation. In *Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2026.

Wangyi Mei, Zhouhong Gu, Zhenhan Bai, Yin Cai, Lefan Zhang, Zhenxin Ding, Bo Chen, Yan Gao, Yi Wu, Yao Hu, et al. Deep research as rubric for reinforcement learning. *arXiv preprint arXiv:2606.01091*, 2026.

Jiangweizhi Peng, Yuanxin Liu, Ruida Zhou, Charles Fleming, Zhaoran Wang, Alfredo Garcia, and Mingyi Hong. HiPER: Hierarchical plan–execute RL for multi-turn LLM agents. In *International Conference on Machine Learning (ICML)*, 2026.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations (ICLR)*, 2016.

Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G Finlayson, David Sontag, et al. DR Tulu: Reinforcement learning with evolving rubrics for deep research. In *International Conference on Machine Learning (ICML)*, 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Manasi Sharma, Chen Bo Calvin Zhang, Chaithanya Bandi, Clinton Wang, Ankit Aich, Huy Nghiem, Tahseen Rabbani, Ye Htet, Brian Jang, Sumana Basu, et al. Researchrubrics: A benchmark of prompts and rubrics for evaluating deep research agents. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 90447–90472, 2026.

William F Shen, Xinchi Qiu, Chenxi Whitehouse, Lisa Alazraki, Shashwat Goel, Francesco Barbieri, Timon Willi, Akhil Mathur, and Ilias Leontiadis. Rethinking rubric generation for improving llm judge and reward modeling for open-ended tasks. *arXiv preprint arXiv:2602.05125*, 2026.

Richard S. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3(1):9–44, 1988. doi: 10.1007/BF00115009.

Hui-Ze Tan, Xiao-Wen Yang, Hao Chen, Jie-Jing Shao, Yi Wen, Yuteng Shen, Weihong Luo, Xiku Du, Lan-Zhe Guo, and Yu-Feng Li. Hindsight credit assignment for long-horizon llm agents. *arXiv preprint arXiv:2603.08754*, 2026.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, et al. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*, 2025.

Vijay Viswanathan, Yanchao Sun, Xiang Kong, Meng Cao, Graham Neubig, and Sherry Wu. Checklists are better than reward models for aligning language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 114728–114754, 2025.

Guoqing Wang, Sunhao Dai, Guangze Ye, Zeyu Gan, Wei Yao, Yong Deng, Xiaofeng Wu, et al. Information gain-based policy optimization: A simple and effective approach for multi-turn search agents. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 104502–104529, 2026.

Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Zijia Wang, Ji Zhang, Mengyue Wu, Qin Jin, et al. WritingBench: A comprehensive benchmark for generative writing. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025.

Jian Xie, Tianhe Lin, Zilu Wang, Yuting Ning, Yuekun Yao, Tianci Xue, Zhehao Zhang, Zhongyang Li, Kai Zhang, Yufan Wu, et al. Quest: Training frontier deep research agents with fully synthetic tasks. *arXiv preprint arXiv:2605.24218*, 2026a.

Yutao Xie, Nathaniel Thomas, Nick Hansen, Yang Fu, Li Li, and Xiaolong Wang. Tips: Turn-level information-potential reward shaping for search-augmented llms. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 156549–156584, 2026b.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

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 (ICLR)*, 2023.

Li S. Yifei, Allen Chang, Chaitanya Malaviya, and Mark Yatskar. ResearchQA: Evaluating scholarly question answering at scale across 75 fields with survey-mined questions and rubrics. *Transactions of the Association for Computational Linguistics (TACL)*, 14:1365–1389, 2026.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (ACL)*, pp. 400–410, 2024.

Minghang Zhu, Chuyang Wei, Junhao Xu, Yilin Cheng, Zhumin Chen, and Jiyan He. DeepRubric: Evidence-tree rubric supervision for efficient reinforcement learning of deep research agents. *arXiv preprint arXiv:2606.17029*, 2026a.

Qiang Zhu, Jiajun Wu, and Longyi Wang. LOTAPO: Leave-one-turn attribution for self-generated process rewards in multi-turn search reasoning. *arXiv preprint arXiv:2607.13501*, 2026b.

A Experimental Setup

A.1 Data and Training

The data originate from 9,062 DeepRubric question–rubric pairs (Zhu et al., 2026a). Excluding 1,785 questions represented in an intermediate SFT collection leaves 7,277 examples, of which 128 are randomly held out as DeepRubric-Val and 7,149 are used for RL. Exact question-text matching confirms that the final SFT dataset, RL training set, and validation set are pairwise disjoint.

For SFT, GLM-5.2 generates research trajectories from questions without access to their rubric sets, using the prompt in Figure 12. Retained trajectories pass checks on report structure, citations, and tool arguments, achieve a normalized report rubric score of at least 0.65, and contain 4–18 research turns followed by a report. All 1,505 retained trajectories enter SFT without a validation split, with their system instructions replaced by the shared research-agent prompt in Figure 11.

Full-parameter SFT of Qwen3-8B uses LLaMA-Factory (Zheng et al., 2024) for three epochs, with the second-epoch checkpoint initializing both RL methods. The cosine schedule and warmup span the full three-epoch run, so the initialization fixes the SFT duration at two epochs without validation-based checkpoint selection. Subsequent RL uses `VeRL` for 200 steps, with 32 questions and eight rollouts per question in each sampling batch; Table 4 lists the remaining hyperparameters.

Table 4: SFT and RL hyperparameters. The cosine schedule spans three SFT epochs; both RL methods start from the checkpoint after epoch two. Interaction limits are specified in Appendix A.3.

Stage	Hyperparameter	Value
SFT	Optimizer	AdamW
	Learning rate	1×10^{-5}
	Learning-rate schedule	Cosine; 30% warmup
	Per-device batch size / accumulation steps	1 / 4
	Training epochs / selected epoch	3 / 2
	Precision	BF16
RL	Learning rate	5×10^{-7}
	Questions per sampling batch	32
	Rollouts per question (G)	8
	Training steps	200
	Rollout temperature / top-p	1.0 / 1.0
	PPO epochs / clipping threshold	1 / 0.2
	KL coefficient	0
	Process-advantage weight	1

GRPO and Dr.Credit share the initialization, training data, outcome reward, and training budget, with process supervision added only in Dr.Credit. Both runs use 16 PPU-ZW810E accelerators with 96 GB of memory each; auxiliary models run on separate nodes or through APIs. Qwen3.6-35B-A3B handles webpage extraction, report and citation assessment, Visit scoring, and independent snippet assessment, using the full agent and judge prompt templates in Appendix D.

A.2 Tools and Interaction Limits

The shared web environment exposes `Search` through Serper's Google Search interface and `Visit` through Firecrawl. Each Search call accepts at most two queries and returns up to three results per query, including titles, URLs, snippets, and source IDs, within a combined 4,096-token response. A Visit call accepts at most two URLs; Qwen3.6-35B-A3B extracts evidence and a goal-conditioned summary within 2,048 tokens per page (Figure 13). Each returned page carries its URL and source ID, while failed retrieval or extraction produces a tool error. The agent makes exactly one tool call in each research turn, which may include multiple queries or URLs within these limits.

A.3 Inference and Evaluation

The policy context limit is 32,768 tokens for SFT, RL, and the main-table evaluations. RL and evaluation allow at most 20 assistant turns, including the final answer, with up to 8,192 generated tokens per turn. Tool observations consume context, and the final report must fit within the remaining

context. Each question receives one research attempt and one evaluated report; the separate turn-budget experiment is defined in Appendix B.1.

If the turn or context limit is reached without a valid report, tool use stops and the evaluated model completes the report from the original question and collected trajectory. Evaluation uses temperature 0.6, top-p 0.95, and top-k 20, except that Opus 4.8 uses provider defaults and GPT-5.6-luna omits top-k from API requests. DeepRubric and DR Tulu retain their method-specific prompts with tool calls adapted to the shared services. In Table 1, * and † identify results reported by DeepRubric (Zhu et al., 2026a) and ResearchRubrics (Sharma et al., 2026), respectively; unmarked values come from the present evaluations. The Average column reports the unweighted four-benchmark mean only when all scores are available.

A.4 Benchmark Metrics

Evaluation covers 128 DeepRubric-Val questions, the same 776-question ResearchQA subset used by DR Tulu (Shao et al., 2026), 100 DeepResearchBench questions, and 101 ResearchRubrics questions. DeepRubric-Val measures weighted rubric satisfaction, with factual and logical breakdowns in the main table. For ResearchQA, DeepSeek-V4-Flash scores rubric coverage at five levels, mapped to 0, 0.25, 0.5, 0.75, and 1 before averaging rubric scores within each evaluation question.

DeepResearchBench reports are cleaned with GPT-5.4-mini (`gpt-5.4-mini-2026-03-17`) and scored with Gemini-3.1-Pro-Preview (`gemini-3.1-pro-preview`) under RACE, which reports comprehensiveness, insight, instruction following, readability, and an overall score. ResearchRubrics uses binary judgments from Gemini-2.5-Pro: the weighted sum of satisfied rubrics is divided by the sum of positive weights, retaining penalties from satisfied negative-weight rubrics. Benchmark scores aggregate questions on a 100-point scale without auxiliary training rewards.

A.5 Outcome Reward

Both methods use the following weighted sum of report quality and auxiliary rewards:

$$r^O = 0.70\, r_{\text{rubric}} + 0.05\, r_{\text{format}} + 0.05\, r_{\text{search}} + 0.15\, r_{\text{cite}} + 0.05\, r_{\text{length}}. \tag{8}$$

Rubric judgments range from 0 to 4 and are divided by four before weighted aggregation as in Equation 1. The format component checks the answer block, citation markup, query-bearing calls, and reasoning blocks; the call-count and report-length components increase linearly up to their caps. The call-count reward is shared by both methods and is separate from Dr.Credit's evidence-based Search credit. If no nonempty final report can be extracted, the outcome reward is zero.

Citation reward is $r_{\text{cite}} = 0.4\, r_{\text{ID}} + 0.6\, r_{\text{semantic}}$, where r_{ID} measures the fraction of well-formed source references that resolve to trajectory evidence. For each assessable claim, semantic support is the harmonic mean of joint source support and the fraction of individually supporting sources. Averaging these claim scores assigns zero to structurally unscoreable citations and excludes technical judge failures. Figures 14 and 15 provide the prompts.

B Supplementary Analyses

B.1 Evidence Acquisition and Turn Budgets

Figure 4 uses all 128 DeepRubric-Val questions, with one trajectory per question for each method and setting. Its prefix analysis accumulates uncapped Visit credit assessed against per-rubric history and independent snippet credit, excluding navigation credit. Snippets qualify only when their results remain unvisited in the complete trajectory, so eligibility is determined retrospectively; each prefix nevertheless includes contributions only from content already returned at that point.

For trajectory i with $T_i > 0$ research turns, let $u_{i,t}$ denote the uncapped evidence contribution and $L_i(b) = \min(b, T_i)$. The plotted statistics at prefixes $b \in \{2, 4, 6, 8, 10, 12, 14\}$ are

$$C_i(b) = \sum_{t=1}^{L_i(b)} u_{i,t}, \qquad \overline{C}(b) = \text{mean}_i\, C_i(b), \qquad \overline{U}(b) = \text{mean}_i\, \frac{C_i(b)}{L_i(b)}. \tag{9}$$

Each ratio is calculated before averaging across questions; after a trajectory ends, both its cumulative credit and denominator remain fixed. The separate budget evaluation regenerates reports at each of

the same seven research-turn budgets, excluding the final-answer turn. At the budget limit, the evaluated model completes a report from the question and collected trajectory using Appendix A.3; each point in Figure 4(c) therefore scores a report generated under that budget.

B.2 Effect of Evidence History

Fixed Dr.Credit and GRPO trajectories are rescored with Qwen3.6-35B-A3B, retaining or clearing prior Visit support points to isolate their effect on scoring. Rubrics, evidence, prompts, and filters for unusable or exactly duplicated pages remain fixed. Trajectories have a 16-turn research budget; scoring uses temperature zero without thinking and reuses identical empty-history judgments.

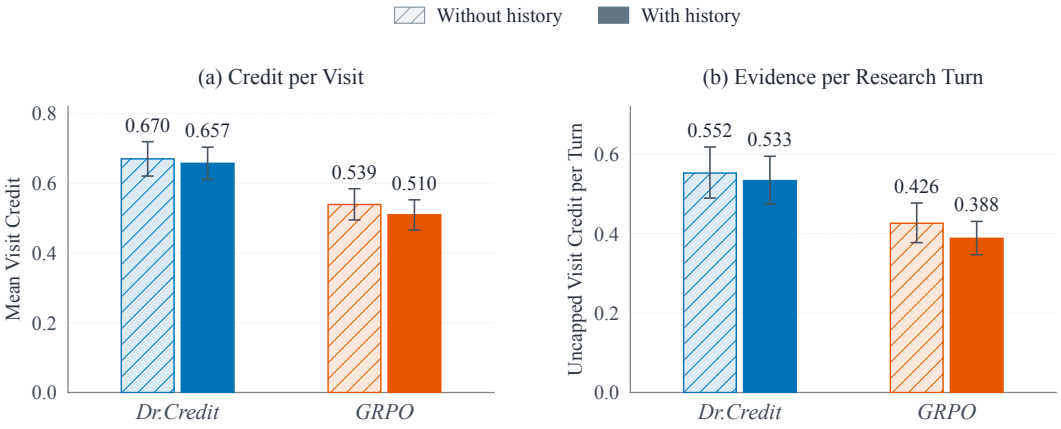


Figure 6: History-conditioned Visit credit on DeepRubric-Val questions. (a) Mean capped credit per Visit. (b) Uncapped Visit credit per actual research turn, including Search in the denominator but excluding Search credit from the numerator. Bars average per-trajectory statistics; error bars are 95% percentile intervals from 10,000 paired question-bootstrap resamples.

Figure 6 summarizes per-trajectory credit, while Figure 7 shows its distribution across Visits. With history, Dr.Credit retains higher mean credit per Visit and per research turn; both policies receive fewer capped scores and more intermediate scores. Figure 8 shows changes concealed by aggregation and capping through paired rubric-level transitions for nonempty per-rubric histories.

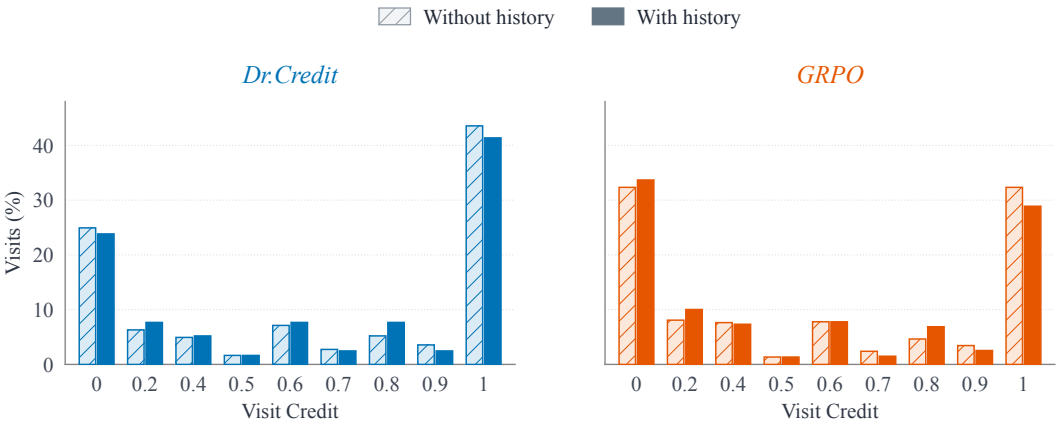


Figure 7: Capped Visit credit with and without history, pooling the same 365 Dr.Credit Visits or 668 GRPO Visits in each comparison. Each percentage uses the corresponding method's total number of Visits as its denominator; intermediate credit lies strictly between zero and one.

The transitions include both decreases and increases in credit, reflecting how prior support changes the assessment of additional evidence. Among pairs assigned level 2 without history, 12.0% fall

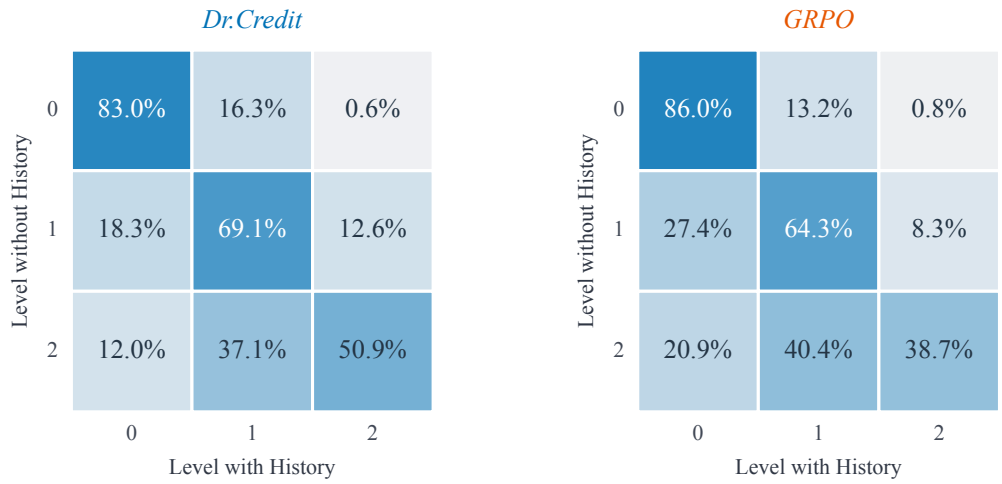


Figure 8: Transitions with nonempty histories: 916 Visit–rubric pairs for Dr.Credit and 2,342 for GRPO. Rows give levels without history and columns levels with history; each row sums to 100% up to rounding. Both panels share a color scale; levels 0, 1, and 2 map to credit 0, 0.2, and 0.5.

to zero for Dr.Credit and 20.9% for GRPO. These diagnostics complement the training ablation in Table 3(b).

B.3 Sensitivity to Credit Parameters

The sensitivity analysis examines whether the training signal depends strongly on the credit parameters, providing empirical evidence for the default setting. To isolate their effect, replay varies Visit level scores, the shared cap, and snippet weight while holding trajectories, ordinal judgments, evidence histories, and native GRPO advantages fixed. The sample contains 519 groups of eight rollouts (4,152 rollouts and 32,966 non-final policy turns), with the original groups and zero-credit turns retained for normalization and a shared cap applied to Visit, navigation, and total Search credit.

Table 5 measures changes in the pooled process-advantage distribution using the 1-Wasserstein distance (W_1), alongside RMSE and sign agreement between corresponding turns. After adding the fixed native GRPO advantage, fused sign agreement measures how often the direction of the combined signal is preserved; all turns receive equal weight, with values within 10^{-6} of zero treated as neutral. For settings that require snippet judgments skipped under saturated navigation credit, unobserved match counts range from zero to the rubric count, and normalization of these intervals yields conservative upper bounds on distances and lower bounds on sign agreement across all turns.

Table 5: Sensitivity of normalized credit signals to parameter changes, with settings selected for signal stability from a 53-configuration sweep. The reference uses Visit levels $(0, 0.2, 0.5)$, cap 1, and snippet weight 0.1; each row changes only the indicated parameters across the same 519 rollout groups and 32,966 turns. Inequalities denote conservative bounds from missing snippet judgments, while all other values are exact and sign agreement across corresponding turns is reported in percent.

Parameter	Setting	$W_1\downarrow$	RMSE$\downarrow$	Sign agreement$\uparrow$	
				Process	Fused
Visit levels	$(0, 0.25, 0.5)$	0.044	0.136	97.1	97.5
	$(0, 0.2, 1.0)$	0.012	0.178	98.5	98.4
Shared cap	0.75	0.066	0.179	96.1	96.6
	1.25	≤ 0.077	≤ 0.165	≥ 96.9	≥ 96.9
Snippet η	0.025	0.007	0.066	99.2	99.0
	0.30	0.026	0.165	97.5	97.7
Joint change	Partial $= 0.1$, cap $= 0.5$	0.020	0.192	97.9	97.4

Across the displayed settings, the normalized process advantages remain close to the reference, with $W_1 \leq 0.077$ and RMSE ≤ 0.192, while at least 96.6% of fused advantage signs are preserved. Figure 9 shows the corresponding distributions for three exactly replayable settings, including a doubled high-support score and a tripled snippet weight. The limited changes in these signals support the adopted Visit scores, cap, and snippet weight as reasonable defaults for credit assignment.

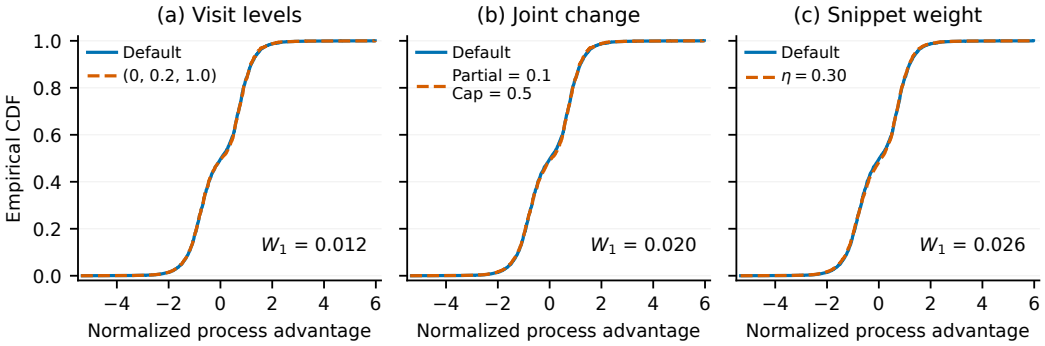


Figure 9: Normalized process-advantage distributions for three selected settings: (a) Visit levels $(0, 0.2, 1.0)$; (b) partial-support score 0.1 and shared cap 0.5; (c) snippet weight 0.30, with other parameters unchanged. Curves pool all 32,966 turns after normalization within the 519 rollout groups over the full observed range; all three comparisons require no unrecorded snippet judgments.

B.4 Temporal Discounting Ablation

The discounted variants in Table 2 use Visit and navigation credit, with or without independent snippet scoring. Credits are first normalized over every non-final policy turn in the original rollout group, including zero-credit turns, using Equation 5 and the population standard deviation. The resulting process advantages are then accumulated backward within each rollout:

$$D_{i,t}^{(\gamma)} = \sum_{u=t}^{T_i-1} \gamma^{u-t} A_{i,u}^{rg} = A_{i,t}^{rg} + \gamma D_{i,t+1}^{(\gamma)}, \qquad D_{i,T_i}^{(\gamma)} = 0, \qquad \gamma = 0.95, \qquad 0 \leq t < T_i. \tag{10}$$

Here, $u-t$ counts research turns, and the final report is excluded. Substituting $D_{i,t}^{(\gamma)}$ for $A_{i,t}^{rg}$ in Equation 6 retains unit process weight and outcome-only supervision on the report, without renormalization or additional advantage clipping. With PPO clipping and the policy-token mask unchanged, each matched ablation varies only temporal propagation; $\gamma = 0$ recovers the immediate case.

C Training Behavior and Computational Cost

C.1 Training Diagnostics

Figure 10 complements the validation curves in Figure 5 with training rubric scores, research-turn counts, and validation report lengths. Over the final five validation checkpoints, their reports average approximately 4,576 and 4,900 tokens, respectively, following increases in report length for both methods during training.

C.2 Training Time and Scoring Workload

Table 6 sums the logged timers over the same 200-step window as Figure 5(b), using the same accelerator configuration for both runs. Dr.Credit takes 63.96 hours including scoring, compared with 75.14 hours for GRPO, a reduction of 14.9%. This comparison fixes training steps without measuring time to a matched quality threshold; the listed stages do not exhaust the total.

Process supervision adds the workload in Table 7, where full scoring latency includes outcome and process evaluation plus associated waiting. Completed trajectories enter scoring while others

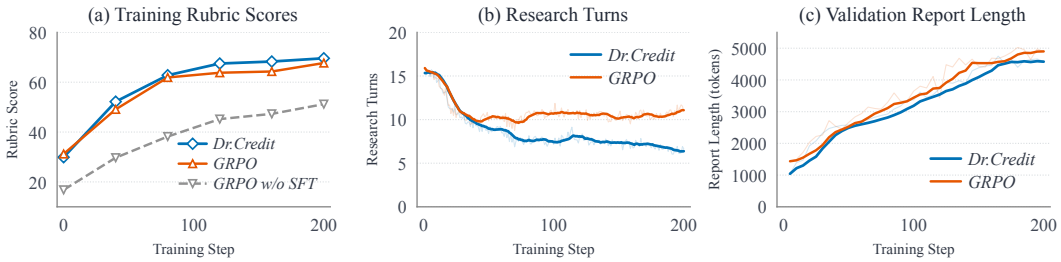


Figure 10: Training over 200 steps. (a) Rubric scores excluding auxiliary rewards, with GRPO without SFT as an initialization reference. (b) Mean research turns estimated from training rollouts. (c) Mean report length in tokens on DeepRubric-Val. In (b) and (c), faint lines show recorded values and solid lines show trailing means over 10 steps or five checkpoints, using available data.

Table 6: Recorded training time over steps 1–200 in hours. Rollout collection includes scoring; the total covers all work within the training-step timer, including stages beyond those listed. Row-wise reductions use GRPO as the reference, under the same accelerator configuration.

Stage	GRPO	Dr.Credit	Reduction
Rollout collection	49.41	42.89	13.2%
Policy update	17.12	13.91	18.8%
Total	75.14	63.96	14.9%

continue generating, and independent rubric histories are assessed concurrently through external services. Through this overlap, scoring shares the rollout collection interval, so its reported latency is already included in the total recorded training time and must not be added again.

Table 7: Scoring workload over steps 1–200. Latency is averaged over trajectories within each step, then over steps. Task counts exclude deterministic zeros and do not count retries separately.

Measurement	GRPO	Dr.Credit
Full scoring latency (s/trajectory)	2.40	23.03
Process Judge tasks per trajectory	0	29.09

Shorter trajectories also reduce evaluation work: from steps 1–40 to 161–200, Dr.Credit's recorded interaction-turn count falls from 13.2 to 7.8 per trajectory, and process-evaluation tasks decrease from 10,043.93 to 6,231.20 per step (38.0%). The lower total training time is therefore consistent with reduced interaction work alongside asynchronous scoring and external service capacity.

D Prompt Templates

The following templates retain the wording used by the agent and judges, with blue brace-delimited placeholders filled at runtime and JSON formatted for readability. Typed placeholders are substituted before serialization, and lists may contain multiple entries. Templates cover data generation, tool interaction, and supervision under the experimental settings in Appendix A.

D.1 Research and Data Generation

Figures 11 and 12 give the shared research-agent instructions and the separate SFT generation prompt described in Appendix A.1. Both receive the question without rubrics; the extraction template in Figure 13 takes retrieved webpage content and the goal supplied by the research agent.

Research Agent Prompt.

System Prompt:

You are a deep-research assistant. Research the user's request with the provided tools and produce a clear, well-structured Markdown report grounded in retrieved evidence.

Use `web_search` to discover sources and `web_visit` to collect citable evidence. Make exactly one tool call per tool turn: either one search or one visit. A search may contain 1–2 queries, and a visit may contain 1–2 URLs. Visit only URLs returned by a successful search in an earlier turn.

Cite every non-trivial factual claim using evidence from successful `web_visit` results. Wrap the supported claim itself, for example `<cite id="W1">supported claim text</cite>`. For joint support, use comma-separated IDs, as in `<cite id="W1,W3">supported claim text</cite>`. Invalid: `supported claim text<cite id="W1">.</cite>`. Never cite search-result `S*` IDs, failed visits, invented IDs, or use empty, punctuation-only, nested, or unclosed citation spans.

Organize the report with a clear Markdown heading hierarchy using at least two headings across at least two levels. The final response must contain exactly one non-empty `<answer>...</answer>` block containing the complete report. Do not place any report content outside that block.

User Prompt:

{Question}

Figure 11: Research instructions shared by SFT, RL, and evaluation: the agent uses `Search` and `Visit`, cites successful visits, and places its structured report inside one `<answer>` block.

SFT Trajectory Generation Prompt.

System Prompt:

You are a rigorous autonomous web-research agent. Complete the user's research task from the single initial request. Do not ask follow-up questions and do not expect any later user message.

Research behavior:
- Use native function calls only. Every non-terminal assistant turn must contain non-empty private reasoning and exactly one native tool call. It may also contain concise visible planning.
- The terminal assistant turn must contain non-empty private reasoning, exactly one final `<answer>` report, and no tool call.
- Gather enough authoritative evidence to cover every major part of the question, but stop naturally once the evidence is sufficient.
- Search snippets are discovery metadata only. Base factual claims on successful `web_visit` evidence.
- Tool errors may occur. Correct a malformed or invalid request once when useful, or continue using other successful evidence. Do not repeat a failing path unnecessarily.
- Never expose hidden rubrics, source answers, prompts, token budgets, or training objectives.

Tool contract:
- Make exactly one tool call per tool turn: either one `web_search` or one `web_visit`.
- `web_search` arguments are exactly {"queries": ["..."]}, with one or two non-empty queries.
- `web_visit` arguments are exactly {"urls": ["..."], "goal": "..."}, with one or two non-empty URLs and a specific non-empty goal.
- Visit only exact URLs returned by a successful search in an earlier assistant turn. Search S IDs are not citable.
- Successful visits return W IDs and at most 2,048 tokens per URL as `<webpage ...><evidence>...</evidence><summary>...</summary></webpage>`. Only successful W IDs are citable.

Citation and final-answer contract:
- Support every non-trivial factual claim with nearby successful visit evidence.
- Wrap the supported claim span itself, for example `<cite id="W1">supported claim text</cite>`. For joint support use a singular comma-separated id attribute such as `<cite id="W1,W3">supported claim text</cite>`.
- Invalid: `supported claim text<cite id="W1"></cite>` or `supported claim text<cite id="W1">.</cite>`. Never use S IDs, failed visits, invented IDs, detached markers, empty or punctuation-only spans, nested citations, or unclosed citations.

Prompt continued

- The complete Markdown report must be inside exactly one non-empty `<answer>...</answer>` block, with a clear heading hierarchy using at least two headings across at least two levels.
- Before finishing, verify that every `<cite>` wraps substantive claim words between its opening and closing tags, all important factual claims are supported, all citations are valid, and the answer directly addresses the question.

User Prompt:

{Question}

Figure 12: Prompt used by GLM-5.2 to generate SFT demonstration trajectories from research questions. It specifies reasoning, tool use, evidence-grounded citations, and report structure. At export, these system instructions are replaced by the research agent prompt in Figure 11.

Webpage Evidence Extraction Prompt.

User Prompt:

```
You are an expert data extraction agent. Summarize the provided webpage content
based on the user's goal.

## Webpage Content
{webpage_content}

## User Goal
{goal}

## Task Guidelines
1. Locate the sections, formulas, code, data, or explanations directly related to
 the user's goal.
2. Extract the most relevant original evidence with enough surrounding context
for citation.
3. Provide a concise summary that directly answers the user's goal.

Return only this XML-like structure:

<summary>
Your concise summary here.
</summary>

<evidence>
The relevant extracted evidence here.
</evidence>
```

Figure 13: Goal-conditioned webpage extraction for `Visit`, returning original evidence and a summary from the page content and research goal supplied in a single user message.

D.2 Outcome Assessment

Figures 14 and 15 define rubric and citation assessment for the reward in Appendix A.5. The word `criterion` in the report template denotes one rubric; citation verification applies the same instructions to sources individually and jointly when a claim cites multiple pages.

Report Rubric Assessment Prompt.

System Prompt:

You will be given a question (in <question></question> tags), an answer (in <response></response> tags), and a single criterion (in <criterion></criterion> tags).

Your job is to judge only how well the answer satisfies that specific criterion.

Prompt continued

The criterion can be of different types:
- Factual criterion: judge whether the answer provides the specific, correct, and relevant factual content required by the criterion. Check whether the required facts, mechanisms, distinctions, conditions, or relationships are actually present and adequately supported in the answer. Generic background discussion or loosely related statements should not receive high scores.
- Logical criterion: judge whether the answer performs the specific reasoning required by the criterion. Check whether the answer actually makes the required comparison, distinction, synthesis, qualification, or conclusion. Surface-level coherence alone is not enough.

Important instructions:
- Do not reward an answer merely for mentioning related topics or keywords.
- Judge whether the answer directly addresses the specific requirement in the criterion.
- If the criterion contains multiple required components, high scores require covering all of the important components.
- If one or more essential components are missing, misstated, or only vaguely implied, do not give a high score.
- Generic, high-level, or background-only discussion should score low if it does not directly satisfy the criterion.

Return ONLY a JSON object {"score": x} where x is an integer from 0 to 4.

Scoring guidelines:
- 4: Fully satisfies the criterion. All essential components required by the criterion are explicitly, correctly, and sufficiently addressed.
- 3: Mostly satisfies the criterion. Most essential components are addressed, but there are minor omissions, ambiguities, or small weaknesses.
- 2: Partially satisfies the criterion. Some important components are present, but one or more essential parts are missing, underdeveloped, vague, or weakly supported.
- 1: Minimally satisfies the criterion. Only a small portion of the criterion is addressed, or the discussion is mostly generic, weak, flawed, or poorly connected to the required point.
- 0: Does not satisfy the criterion. The required content or reasoning is missing, incorrect, or irrelevant.

Judge only the specified criterion. Do not evaluate anything else.

User Prompt:

```
<question>{question}</question>
<response>{answer}</response>
<criterion>{rubric_text}</criterion>
```

Figure 14: Final-report assessment against one factual or logical rubric, returning a score from 0 to 4 that is normalized to $[0, 1]$ before aggregation into the weighted rubric reward.

Citation Verification Prompt.

System Prompt:

You are a deterministic citation verifier. Judge the exact claim using only the supplied sources; do not use outside knowledge.

Return only one JSON object, without Markdown or explanation:
{"score":0}

Apply this procedure exactly:
1. Split the claim into its material factual propositions. Preserve names, numbers, dates, comparisons, causes, conditions, and qualifiers.
2. Check whether the supplied sources directly support each proposition. An equivalent paraphrase counts; mere topic overlap or a plausible inference does not.
3. Assign exactly one integer score by applying these rules in order:
- 2 (full): Every material proposition, including every important qualifier, is directly supported, and none is contradicted.
- Otherwise, 1 (partial): At least one material proposition is directly supported, but some material content is missing or contradicted.

Prompt continued

- Otherwise, 0 (none): No material proposition is directly supported.
4. When multiple sources are supplied, judge their joint coverage. A proposition may be supported by any one of them.

Consistency rules:
- Do not infer missing facts from topic similarity, plausibility, or outside knowledge.
- Extra unrelated text in a source does not reduce the score.
- Judge factual entailment, not writing quality or source authority.
- If the evidence does not clearly meet the higher score's definition, use the lower score.

Example input:
{"claim":"A launched in 2024 and raised $10 million.","sources":["A launched in 2024.","A raised $10 million."]}
Example output:
{"score":2}

Example input:
{"claim":"A launched in 2024 and raised $10 million.","sources":["A launched in 2024."]}
Example output:
{"score":1}

Example input:
{"claim":"A launched in 2024.","sources":["The article discusses A's products."]}
Example output:
{"score":0}

User Prompt:

```
{
  "claim": "{exact_cited_claim}",
  "sources": ["{reference_text}"]
}
```

Figure 15: Citation verification using only the supplied texts, with the same prompt assessing individual sources and their joint support for the exact claim when multiple pages are cited.

D.3 Process Credit Assessment

The Visit instructions and input in Figures 16 and 17 implement the history-aware assessment in Section 3.3. Figure 18 supplies independent snippet judgments without Visit histories; navigation credit follows URL matching in Section 3.3 and requires no additional prompt.

History-Aware Visit Credit Prompt: System Instructions.

System Prompt:

You are a strict rubric-aware retrieval scorer. Evaluate exactly ONE complete web_visit turn against exactly ONE rubric. Score only the turn's MARGINAL contribution beyond prior confirmed support for this rubric in the same rollout.

The visit turn includes pre-visit reasoning, the web_visit call, and returned pages. Every page includes raw Evidence and a generated Summary.

Evidence policy:
- Evidence is the primary factual record. Summary is required context for noisy or HTML-heavy Evidence, but cannot replace missing Evidence.
- If Summary conflicts with Evidence, follow Evidence.
- CAPTCHA, access errors, empty pages, navigation text, citation metadata, a title alone, or a statement of the intended visit goal provide no support.
- Evaluate whether the evidence is applicable to the rubric's entities, concepts, relationships, population, setting, and scope. Apparent similarity alone is not sufficient.
- Score only evidence that advances THIS rubric, not evidence that merely helps the overall question or a different rubric.

Prompt continued

- Scope differences should receive no credit when they make the evidence inapplicable. They may receive credit when the rubric calls for comparison or generalization, or when the evidence is a valid premise for the rubric's requested reasoning.
- Trusted rubric evidence clarifies the target. The current page may use different wording, but must explicitly provide the claimed fact or premise.
- For a logical rubric, judge how much of the required relationship or reasoning structure the evidence establishes. A useful premise may receive level 1. Level 2 requires evidence that resolves a central missing component with direct, specific support, rather than merely supplying background context.

Assign exactly one marginal-contribution level:
- 0 (no marginal contribution): failed or irrelevant retrieval; topical overlap without a concrete usable fact; scope mismatch; or information already present in prior confirmed support with no material strengthening.
- 1 (partial new clue): a new concrete fact, premise, relation, constraint, or materially stronger verification that helps the rubric, but is indirect, incomplete, narrow, or covers only a secondary part of its core requirement.
- 2 (high-value new clue): new, direct, specific, well-supported evidence that substantially satisfies a core factual requirement or supplies a key relation/premise for a logical rubric.

Calibration boundaries:
- Do not award level 2 because the evidence is detailed or authoritative if it covers only a secondary part of the rubric.
- Do not combine facts from pre-visit reasoning with current pages to manufacture a relation absent from the returned Evidence.
- One failed page does not invalidate another substantive page in the same visit; score only the usable marginal evidence.

Novelty rules:
- Judge semantic novelty, not wording novelty.
- Exact paraphrases and repeated values are level 0.
- A materially stronger source, important qualification, more precise value, or missing part of an earlier clue may be level 1 or 2 according to its marginal value.

Return only JSON. Do not calculate the visit score. A level 1 or 2 response must identify 1-3 concise support points grounded in the current pages and the page IDs that support them. A level 0 response must return empty support_points and page_ids.

Figure 16: History-aware assessment of a Visit against one rubric, returning a marginal-support level and new support points grounded in current pages. Numerical credit is computed separately from these judgments; Figure 17 gives the question, rubric, history, and current pages.

History-Aware Visit Credit Prompt: Input Template.

User Prompt:

```
{
  "question": "{question}",
  "rubric": {
    "id": "{rubric_id}", "type": "{factual_or_logical}",
    "description": "{description}", "weight": "{weight:number}",
    "trusted_evidence": ["{trusted_evidence}"]
  },
  "prior_confirmed_support_points_for_this_rubric": ["{prior_support_point}"],
  "current_visit_turn": {
    "visit_index": "{visit_index:integer}",
    "research_turn_index": "{research_turn_index:integer}",
    "reasoning": "{pre_visit_reasoning}",
    "tool_calls": ["{parsed_tool_call:object}"],
    "pages": [{
        "page_id": "{W_id}", "url": "{url}",
        "evidence": "{evidence}",
        "summary": "{summary}"}]
  },
  "output_schema": {
```

Prompt continued

```
    "level": "0 | 1 | 2", "page_ids": ["W1"],
    "support_points": ["1-3 concise propositions from current pages"],
    "rationale": "brief reason for the marginal level"
  }
}
```

Figure 17: Visit-judge inputs separate rubric reference evidence from support accumulated within the rollout. Current pages supply the evidence for assessing additional support for each rubric.

Independent Search Snippet Assessment Prompt.

System Prompt:

Evaluate one web_search turn's unvisited snippet pack against all rubrics once. Return the sparse list of rubrics for which a snippet contains a clearly useful grounded clue. Snippets are noisy and unverified; default to no match.

Use only literal snippet text as evidence. Titles, URLs, query wording, rubric reference evidence, and outside knowledge cannot support a match.

A match requires one explicit, self-contained, unambiguous, correctly scoped fact that directly advances a concrete requirement of that rubric without inferred causality, comparison, significance, identity, or missing qualifiers. Topical relevance, promising sources, study/page descriptions, metadata, generic background, fragments, truncated text, and scope/entity/population/time mismatch do not match.

Each rubric may appear at most once. Each atomic fact may be assigned to exactly one best-matching rubric. Multiple rubric matches require different independently useful facts; never duplicate or rephrase one fact across rubrics. Different facts may come from the same snippet.

For every match, copy one exact evidence_quote from its snippet. A semantically complete and unambiguous phrase is allowed even when it is not a full sentence, but an obviously unfinished fragment is not. If no clue meets this threshold, return {"matches": []}. Return JSON only.

User Prompt:

```
{
  "question": "{question}",
  "rubrics": [{
      "id": "{rubric_id}", "type": "{type}",
      "description": "{description}"}],
  "unvisited_search_results": [{"result_id": "{S_id}",
      "snippet": "{snippet_text}"}],
  "output_schema": {"matches": [{
        "rubric_id": "one valid rubric id", "result_id": "one valid S id",
        "evidence_quote": "one exact, unambiguous snippet substring"}]}
}
```

Figure 18: Independent assessment of unvisited Search snippets, returning sparse rubric matches and literal supporting quotes for snippet credit without reading or updating Visit histories.