# JUDGING THE PATH, NOT JUST THE ANSWER: TURN-LEVEL VERIFIERS FOR DEEP RESEARCH AGENTS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Autonomous deep research agents tackle complex, open-ended questions by sequentially reasoning and interacting with external environments across extended horizons. The prevailing reinforcement learning paradigm relies on terminal outcome rewards and encounters a fundamental turn-level credit assignment bottleneck. For failed trajectories, terminal penalties uniformly suppress all intermediate actions without distinguishing productive exploratory steps from genuine errors. On challenging queries where all rollouts fail, the learning signal vanishes. While process-level reward paradigms offer finer-grained feedback, they typically evaluate actions against ground-truth answers, limiting their ability to credit intermediate progress along unsuccessful paths and precluding their deployment during test-time search. To this end, we propose **TPV** (**T**urn-level **P**rocess **V**erifier), an unprivileged process verifier that evaluates each intermediate action purely from the interaction context. TPV is trained to predict turn-level grades assigned by an offline, privileged teacher judge that evaluates each action independently of final trajectory success. Empirical evaluations show that TPV reliably discriminates informative from redundant turns even inside failed rollouts, outperforming outcome-based baselines and transferring across diverse model scales. Consequently, our unified verifier serves a dual purpose: it delivers dense, fine-grained credit assignment during reinforcement learning, bypassing the zero-gradient dilemma on all-fail queries, and acts as an effective trajectory reranker at inference time. Extensive experiments on five challenging deep research benchmarks demonstrate that integrating TPV into RL training consistently boosts task success rates over competitive baselines, while deploying it as a test-time reranker yields substantial additional accuracy gains.[1]

## 1 INTRODUCTION

As users increasingly delegate complex, exploratory inquiries to LLM agents, traditional search engines fall short. As illustrated in Figure 1(a), conventional search pipelines merely return ranked candidate links, forcing users to manually shoulder the heavy burden of planning search paths, cross-checking conflicting facts, and synthesizing evidence. In contrast, deep research agents transform information seeking into an autonomous, multi-turn exploration process in open web environments (Du et al., 2026; Nakano et al., 2021). As shown in Figure 1(b), the agent interleaves multi-step reasoning with tool usage (Yao et al., 2022), dynamically generating queries conditioned on past observations and adaptively navigating its investigation across long horizons.

To equip deep research agents with effective decision-making capabilities, the prevailing paradigm relies on post-training via supervised fine-tuning (SFT) followed by reinforcement learning (RL) (Ouyang et al., 2022; Bai et al., 2022; Li et al., 2025). While SFT provides foundational formatting and query patterns through behavioral cloning, it inherently relies on static demonstrations and cannot provide agents with exploration capabilities in dynamic environments (Ross et al., 2011). In practice, outcome-based RL with outcome reward models (ORMs) has become the default mechanism to steer agent exploration (Stiennon et al., 2020). However, as the reasoning and interaction horizon deepens in research tasks, outcome-based feedback encounters a severe credit assignment bottleneck (Arjona-Medina et al., 2019). As illustrated in Figure 1(c), terminal scalar

[1]Code is available at https://anonymous.4open.science/r/ICLR2027_2021-E425/.



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.

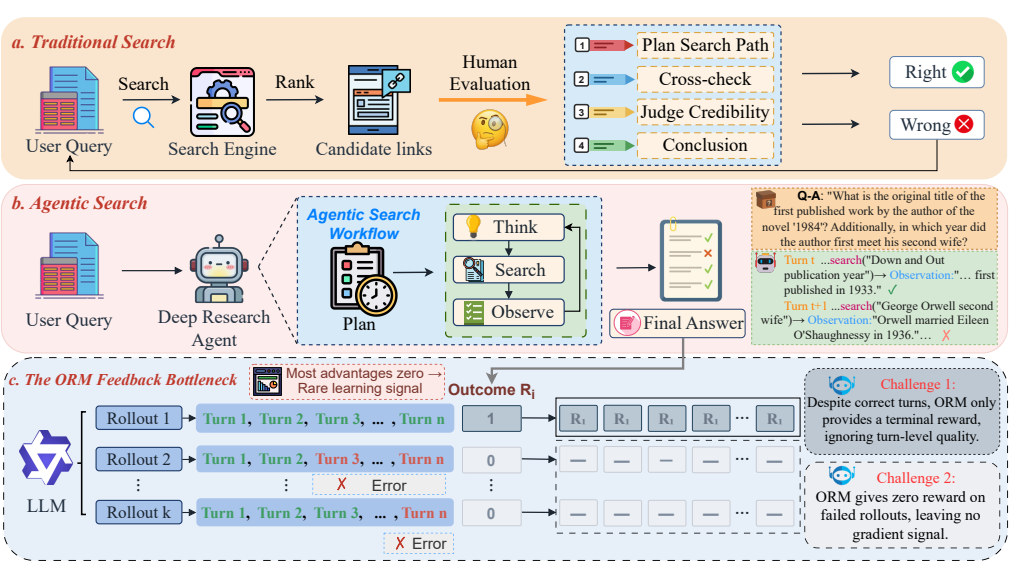



Figure 1: Evolution of search paradigms and the ORM feedback bottleneck. (a) Traditional search relies on human evaluation; (b) agentic search performs autonomous multi-turn actions; (c) outcome-based RL provides only a terminal score and no reward on failed rollouts.

rewards treat the entire interaction trajectory as a uniform unit: when a trajectory succeeds, the sparse outcome reward is diffused evenly across all turns, failing to isolate which intermediate actions matter; conversely, when a trajectory fails, the terminal penalty is propagated uniformly across all turns, suppressing valid exploratory steps alongside genuine errors (Shao et al., 2025). These limitations reveal that conventional outcome-based RL fundamentally lacks turn-level supervision, highlighting the need for fine-grained, process-level reward mechanisms (Yu et al., 2024).

Process reward models (PRMs) have achieved notable success in structured domains like mathematical reasoning, yet conventional approaches heavily rely on explicit, step-level supervision (Lightman et al., 2024). Annotating such intermediate steps is not only expensive but also not well suited for deep research, where agent actions such as web queries and evidence extraction are highly open-ended and dynamic. To alleviate manual labeling, recent advances have explored automated process supervision, primarily through three paradigms: implicit process rewards derived from outcome verifiers (Cui et al., 2025), Monte Carlo rollout value estimation (Wang et al., 2024; Kazemnejad et al., 2024), and likelihood or information-gain metrics (Setlur et al., 2025). While effective for closed-form reasoning tasks (such as math and code), applying these automated paradigms to interactive deep research agents exposes several structural challenges: **1) Implicit rewards: outcome dependence and semantic mismatch.** Methods like PRIME (Cui et al., 2025) derive token-level implicit rewards from log-probability ratios. Because these rewards are normalized against the group outcome, they collapse when all rollouts fail, leaving intermediate turns unguided. Even when they do not collapse, token-level log-ratios primarily capture local fluency rather than the semantic quality of turn-level actions such as query formulation or evidence filtering. **2) Monte Carlo rollout estimation: prohibitive cost and value sparsity.** Estimating turn values via branch rollouts requires repeated simulation (Kazemnejad et al., 2024), each involving expensive LLM generation and web retrieval. When base success rates are low, resampled continuations rarely succeed, so value estimates remain uniformly near-zero and cannot distinguish productive turns from unproductive ones. **3) Likelihood and information-gain signals: ground-truth dependence and test-time failure.** Metrics based on likelihood or information gain (Setlur et al., 2025) are defined strictly against the ground-truth target. Once an exploratory trajectory deviates, these signals collapse, failing to reward valid intermediate turns within overall failed rollouts. Their dependence on ground truth also precludes deployment at test time.

These limitations underscore the clear need for a turn-level process verifier: a dedicated model that evaluates the strategic quality of each tool-calling turn independently of final outcome labels, and can be readily deployed at inference time. To bridge this gap, we introduce **TPV** (**T**urn-level **P**rocess **V**erifier), a lightweight yet effective framework tailored for deep research trajectories. Unlike outcome-based signals or costly simulation rollouts, TPV directly scores intermediate actions

by predicting turn-level grades from interaction context, providing dense turn-level supervision to guide policy optimization. This design rests on a key empirical observation: the judge model reliably uncovers valid turns even within failed rollouts, and this evaluative ability largely persists without access to ground truth. Crucially, this signal is learnable, allowing a student verifier to be trained to anticipate these turn-level judgments solely from the interaction context. TPV is thus trained using the teacher's privileged evaluations as supervisory targets: while the judge grades turns with access to the ground truth, the verifier learns to predict these grades solely from the interaction context. During RL, TPV's predicted grade is composed with rule-based terms for retrieval novelty and format validity. A single frozen verifier thereby serves two roles: delivering informative turn-level rewards when all group rollouts fail, and acting as a trajectory reranker at inference.

Our contributions are fourfold: (1) We propose **TPV**, an independent turn-level process verifier trained via privileged label construction, which provides lightweight evaluation of agent actions without rollouts, relying strictly on the interaction context. (2) We empirically demonstrate that privileged evaluation reliably identifies valid exploration even within globally failed trajectories, and confirm that this supervisory signal can be faithfully recovered by a context-only student model. (3) We show that our label-construction paradigm enables a single frozen verifier to fulfill dual roles: providing turn-level credit assignment during RL training and serving as a test-time trajectory reranker. (4) We establish the effectiveness and generalizability of our approach across multiple deep research benchmarks, demonstrating consistent gains in policy optimization and test-time scaling.

## 2 METHOD

### 2.1 PROBLEM FORMULATION AND RL PRELIMINARIES

**Interactive Deep Research.** Given a question $q$, an agent policy $\pi_\theta$ interacts with an environment across turns $t \in [1, T]$. At each turn, conditioned on history $h_t = (q, a_1, o_1, \ldots, a_{t-1}, o_{t-1})$, $\pi_\theta$ generates an action $a_t = (z_t, c_t)$ consisting of a reasoning trace $z_t$ and a tool command $c_t$ (e.g., `search`, `visit`). The environment returns observation $o_t$, terminating with final answer $\hat{y}$ to form trajectory $\tau = (q, a_1, o_1, \ldots, a_T, o_T, \hat{y})$. Masking observation tokens $o_t$ during updates makes the turn the minimal controllable policy unit. We initialize $\pi_\theta$ via SFT on two publicly available datasets: DeepResearch-9K (Wu et al., 2026) and REDSearcher-SFT-10K (Chu et al., 2026).

**Outcome-Based RL Bottleneck.** In outcome-based RL, a binary reward $R(\tau) = \mathbf{1}[\textsc{Judge}(q, \hat{y}, y^\star) = \text{correct}] \in \{0, 1\}$ is assigned at trajectory terminus against ground truth $y^\star$. Under group-relative policy optimization (Jin et al., 2025), rollout advantage among $K$ trajectories is normalized as (Shao et al., 2024; Guo et al., 2025):

$$\hat{A}_i^{\text{out}} = \frac{R(\tau_i) - \mu}{\sigma + \varepsilon}, \quad \text{where} \quad \mu = \frac{1}{K}\sum_{j=1}^{K} R(\tau_j), \quad \sigma = \text{std}\left(\{R(\tau_j)\}_{j=1}^{K}\right). \tag{1}$$

Eq. 1 suffers two fundamental credit assignment failures: (i) for failed rollouts, uniform penalties suppress productive intermediate exploration together with actual errors; and (ii) for successful rollouts, turn-invariant advantage diffuses credit evenly, failing to isolate decisive actions from redundant or suboptimal turns. This motivates a turn-level reward $r_t = r(a_t \mid h_t)$ conditioned strictly on context, without relying on privileged targets $(y^\star, R(\tau))$ unavailable at inference.

### 2.2 TURN-LEVEL PROCESS VERIFIER (TPV)

**Challenges and Design.** Providing fine-grained credit in deep research presents unique structural challenges: Monte Carlo rollouts (Kazemnejad et al., 2024; Wang et al., 2024) incur prohibitive search latency and value sparsity under low base success rates, while likelihood or info-gain metrics (Setlur et al., 2025; Wang et al., 2025) rely strictly on target $y^\star$, collapsing on deviating paths and failing at test time. To resolve this, a verifier must decouple *credit construction* from *credit prediction*. We instantiate the *learning using privileged information* (LUPI) paradigm (Vapnik & Vashist, 2009; Lopez-Paz et al., 2015): an offline teacher leverages privileged ground truth $y^\star$ to audit individual turns without rollout simulations, while an unprivileged student verifier $v_\phi$ learns to predict these grades purely from interaction context $x_t$ (Figure 2(a)), keeping intermediate rewards dense during RL and directly enabling test-time reranking.

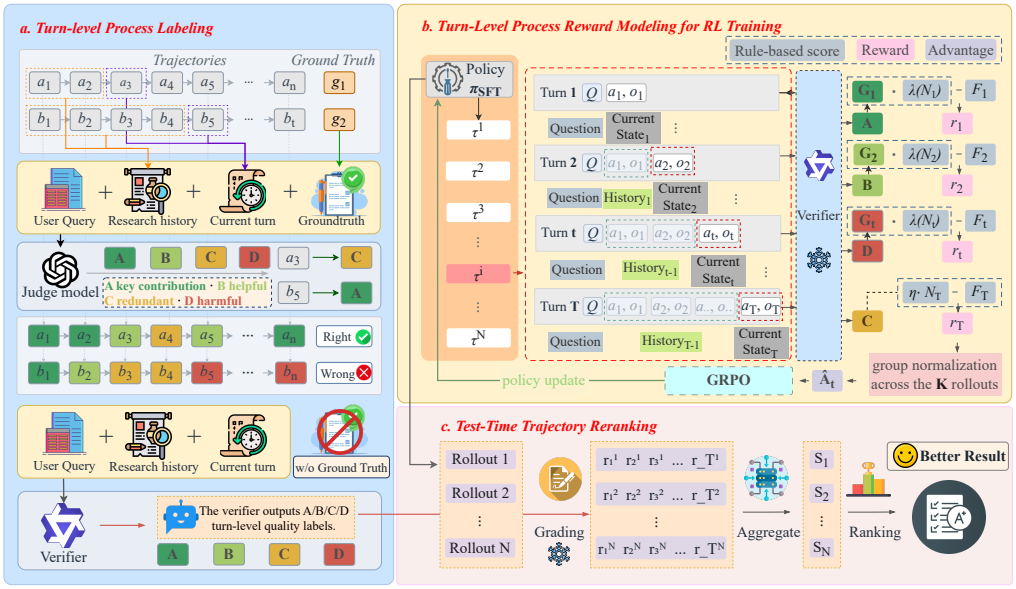



Figure 2: Overview of the TPV framework: (a) a privileged judge grades each turn $\{A, B, C, D\}$ using ground truth, and a verifier learns these grades from context alone; (b, c) the verifier delivers dense GRPO rewards during RL and aggregates scores for test-time reranking.

**Privileged Turn Labeling.** Conditioned on the question $q$, interaction history $h_t$, current turn $(a_t, o_t)$, and ground truth $y^\star$, a judge model $\Phi$ assigns an ordinal contribution grade:

$$g_t \;=\; \Phi\big(q,\, h_t,\, a_t,\, o_t\,;\, y^\star\big) \;\in\; \{A,\, B,\, C,\, D\}, \tag{2}$$

where $A$ (*critical*) uncovers decisive evidence or core reasoning links; $B$ (*helpful*) provides useful context; $C$ (*redundant*) yields uninformative repetition or irrelevant drift; and $D$ (*harmful*) misguides search. Crucially, the judge is blinded to the final trajectory outcome $R(\tau)$, enabling it to penalize redundant turns in successful trajectories ($a_3 \to C$) while accurately crediting valid exploration inside failed rollouts ($b_5 \to A$). Labeling is applied strictly to tool-calling turns (`search` and `visit`), with $h_t$ truncated to the eight most recent turns.

**Training the Verifier.** Because $x_t = (q, h_t, a_t, o_t)$ excludes all privileged targets ($y^\star$, $R(\tau)$, $a_{>t}$), $v_\phi$ operates on identical inputs across training, RL optimization, and inference. We parameterize $v_\phi$ with a four-way classification head on a pretrained language model, trained on $\mathcal{D}_{\text{ver}}$ via class-weighted cross-entropy to balance frequent redundant turns ($C$):

$$\mathcal{L}(\phi) \;=\; -\sum_{(x_t,\, g_t)\in\mathcal{D}_{\text{ver}}} w_{g_t} \log p_\phi(g_t \mid x_t), \qquad w_g \;\propto\; \frac{1}{\sqrt{n_g}}, \qquad \tfrac{1}{4}\sum_{g\in\{A,B,C,D\}} w_g = 1, \tag{3}$$

where $\mathcal{D}_{\text{ver}}$ denotes the teacher-labeled tool turns, $w_g$ is the normalized class weight, and $n_g$ is the class frequency.

**Reward Extraction.** Once trained, the frozen verifier converts its predictive distribution into a turn-level scalar reward $\hat{r}_t$, evaluated either deterministically (*hard*) or in expectation (*soft*):

$$\hat{r}_t \;=\; \begin{cases} r\big(\arg\max_g p_\phi(g \mid x_t),\, N_t,\, F_t\big), & \text{hard}, \\ \sum_g p_\phi(g \mid x_t)\, r\big(g,\, N_t,\, F_t\big), & \text{soft}, \end{cases} \tag{4}$$

where $r(g, N_t, F_t)$ maps the predicted grade $g \in \{A, B, C, D\}$ to a scalar turn reward, adjusted by the rule-based novelty and formatting terms $N_t$ and $F_t$ defined below. Unparsable tool calls bypass $v_\phi$ entirely and directly receive $\hat{r}_t = -F_t$.

**Turn-level Reward.** To account for exploratory novelty and syntax correctness outside $v_\phi$, the composite turn score $r(g, N_t, F_t)$ integrates the predicted grade with a novelty factor $\lambda(N_t)$ and a formatting penalty $F_t$:

$$r\big(g,\, N_t,\, F_t\big) \;=\; \begin{cases} \eta\, N_t \;-\; F_t, & g = C, \\ \nu(g)\cdot\lambda(N_t) \;-\; F_t, & g \in \{A, B, D\}, \end{cases} \qquad \lambda(N_t) = \tfrac{1}{2}\big(1 + N_t\big), \tag{5}$$

where utilities are set to $\nu(A) = 1.0$, $\nu(B) = 0.4$, and $\nu(D) = -0.5$, with $\eta = 0.1$. The novelty score $N_t \in [0, 1]$ discounts redundant tool queries, while $F_t \geq 0$ penalizes syntax or argument violations, escalating with repeated violations (detailed in Appendix B.3). For uninformative turns ($g = C$), the additive term $\eta N_t$ softly encourages novel exploration over redundancy.

**Advantage Assignment.** We embed turn rewards $\hat{r}_t$ at the end token $e_t$ of each turn and the outcome reward $R(\tau_i)$ at the terminal token $e_{\text{end}}$ into a token-level reward vector $u_i$. To prevent dense process rewards from overwhelming sparse outcomes, both channels are standardized independently within the group of $K$ rollouts (Ye et al., 2025; Xu et al., 2025):

$$\tilde{A}_{i,k} = \frac{u_i[k] - \mu_{\mathcal{P}}}{\sigma_{\mathcal{P}} + \varepsilon}, \quad \mathcal{P} \in \{\mathcal{P}^{\text{out}}, \mathcal{P}^{\text{turn}}\}, \qquad \hat{A}^{\text{turn}}_{i,\,e_t} = s \cdot \tilde{A}_{i,\,e_t}, \tag{6}$$

where $\mu_{\mathcal{P}}$ and $\sigma_{\mathcal{P}}$ are channel-specific group statistics, and $s$ dynamically scales turn gradients against the outcome magnitude ($s$ formula and clipping bounds detailed in Appendix B.3).

Per-turn credit is then propagated backward from the outcome advantage $\hat{A}_{i,T+1} \equiv \tilde{A}_{i,e_{\text{end}}}$ with discount $\gamma = 0.95$, clipped, and broadcast to the tokens of each turn:

$$\hat{A}_{i,t} = \hat{A}^{\text{turn}}_{i,\,e_t} + \gamma\, \hat{A}_{i,\,t+1}, \qquad \hat{A}_{i,k} = \text{clip}\left(\hat{A}_{i,t},\, -c,\, c\right) \quad \forall k \in \text{turn } t. \tag{7}$$

Finally, $\hat{A}_{i,k}$ replaces the trajectory-constant outcome advantage $\hat{A}^{\text{out}}_i$ in the clipped GRPO objective (with $\pi_{\text{ref}} = \pi_{\text{SFT}}$ and KL weight $\beta = 0.1$; see Appendix B.3), effectively steering multi-turn exploration with dense, unprivileged credit assignment.

### 2.3 Test-Time Scaling with the Same Verifier

Because $v_\phi$ is strictly context-conditioned, the exact same verifier that provides intermediate RL training signals can directly serve as a test-time reranker. At test time, we sample $N$ candidate trajectories per query, evaluate each rollout using the frozen verifier, and select the optimal trajectory via Best-of-$N$ (BoN) selection.

Specifically, we score each intermediate `search` and `visit` turn using the deterministic hard extraction rule from Eq. 4 (recomputing novelty $N_t$ and format penalty $F_t$ under identical rules), and aggregate turn-level rewards via mean pooling:

$$f(\tau) = \frac{1}{|\mathcal{S}(\tau)|} \sum_{t \in \mathcal{S}(\tau)} \hat{r}_t, \qquad \tau^\star = \underset{i \in \{1,\ldots,N\}}{\arg\max}\; f(\tau_i), \tag{8}$$

where $\mathcal{S}(\tau)$ denotes the index set of intermediate tool-calling turns in trajectory $\tau$. Mean pooling naturally normalizes across variable trajectory lengths, penalizing trajectories that accumulate uninformative drift; comprehensive comparisons against alternative aggregation rules are provided in Appendix D.3. Crucially, averaging per-turn utilities also smooths out isolated verifier misclassifications, yielding a robust trajectory-level scoring proxy without introducing length bias.

## 3 Experiments

We evaluate our framework across five deep-research benchmarks, investigating five core questions: (1) Does turn-level process verification outperform outcome-based and information-gain rewards? (2) Do turn-level labels reflect genuine turn quality rather than the trajectory outcome? (3) Can the verifier recover the judge's turn-level quality without privileged information? (4) Does the framework generalize across policy architectures and verifier scales? (5) Can the frozen verifier effectively guide test-time trajectory reranking?

### 3.1 Experimental Setup

**Datasets.** We evaluate on five deep-research benchmarks spanning multilingual search, fixed corpora, and live web environments: **GAIA** (Mialon et al., 2024) is a general AI assistant benchmark; we use its 127 text questions across varying complexity, **BrowseComp** (Wei et al., 2025) provides 1,266 hard web-exploration queries, **BrowseComp-Plus** (Chen et al., 2025b) extends BrowseComp

Table 1: Overall accuracy across five deep research benchmarks. All arms share the same SFT initialization, prompt, tool stack, and GRPO implementation; only the per-turn reward differs. Best results are in **bold**.

| Model / Method | GAIA (Text-Only) | | | | BrowseComp -plus | Browse Comp | xBench -DS-2510 | Deep SearchQA |
|---|---|---|---|---|---|---|---|---|
| | L1 | L2 | L3 | all | | | | |
| Qwen3-4B-Thinking | 35.71% | 15.15% | 5.26% | 20.47% | 5.42% | 3.40% | 7.00% | 9.11% |
| - GRPO (RL-Only) | 33.33% | 18.18% | 5.26% | 21.26% | 6.14% | 4.34% | 6.50% | 9.00% |
| + SFT | 64.29% | 48.48% | 21.05% | 49.61% | 28.64% | 21.48% | 22.00% | 27.11% |
| - GRPO (ORM) | 59.52% | 46.97% | 26.32% | 48.03% | 32.84% | 19.83% | 19.50% | 24.22% |
| - GRPO (IGPO) | 66.67% | 48.48% | 21.05% | 50.39% | 33.86% | 22.12% | **25.50%** | 28.56% |
| - GRPO (Uni-TPV) | 61.90% | 53.03% | 26.32% | 51.97% | 35.18% | 21.56% | 24.00% | 25.56% |
| - GRPO (TPV) | **76.19%** | **57.58%** | **31.58%** | **53.54%** | **37.23%** | **23.85%** | **25.50%** | **28.89%** |

with a fixed, human-verified corpus of about 100K documents, enabling fair and reproducible comparisons without relying on live web search APIs, **xbench-DeepSearch** (Chen et al., 2025a) targets complex multi-hop information retrieval in Chinese, and **DeepSearchQA** (Gupta et al., 2026) contains multi-step tasks testing the balance between search recall and precision. Environment tooling employs Google Serper for `search` and `trafilatura` (Barbaresi, 2021) for web `visit`; full implementation details are in Appendix A.

**Metrics.** The performance is reported by *Accuracy*, determined by an LLM judge against ground-truth references. For the verifier, we evaluate classification fidelity using *macro-F1* and *balanced accuracy* across the four quality grades, and assess trajectory-level discrimination via *ROC-AUC*. For test-time scaling, we report *best-of-$N$* accuracy against unguided *random@1* and *pass@$N$*.

**Baselines.** To isolate reward effects, all RL runs share identical prompts, tool environments, and GRPO hyperparameters, varying only the reward mechanism: (1) **Base Model**: The un-finetuned foundation model (Qwen3-4B-Thinking) (Team, 2025); (2) **RL-Only**: GRPO applied directly to the base model, bypassing SFT; (3) **SFT**: The base policy fine-tuned on demonstration trajectories without RL; (4) **ORM**: GRPO with terminal outcome rewards broadcast uniformly across tokens; (5) **IGPO** (Team et al., 2026): GRPO with dense turn-level rewards derived from policy information gain; (6) **Uni-TPV**: Receives identical cumulative TPV rewards, but redistributes them uniformly across turns to isolate credit placement from density; and (7) **TPV**: Our proposed policy optimized with turn-level process verifier rewards (Section 2).

### 3.2 Main Results: Advantage of Turn-Level Verification (RQ1)

To verify the effectiveness of turn-level process verification, we evaluate TPV across all five deep-research benchmarks, comparing it against outcome-based GRPO (ORM), information-gain rewards (IGPO), and a uniform ablation (Uni-TPV). Table 1 reports overall accuracy.

**TPV improves every benchmark.** TPV is the best-performing arm on all five benchmarks. Relative to ORM, it raises GAIA from $48.03\%$ to $53.54\%$, BrowseComp-Plus from $32.84\%$ to $37.23\%$, BrowseComp from $19.83\%$ to $23.85\%$, xBench-DS-2510 from $19.50\%$ to $25.50\%$, and DeepSearchQA from $24.22\%$ to $28.89\%$. The gains are consistent across diverse environments rather than concentrated on a single task.

**Outcome-based credit is the bottleneck.** ORM broadcasts one trajectory reward to every token, so all turns in a failed trajectory receive identical credit. On GAIA, it even underperforms plain SFT ($48.03\%$ vs. $49.61\%$), showing that a sparse or misattributed signal can be counterproductive. The gap to TPV is largest on long-trajectory benchmarks (xBench-DS-2510, DeepSearchQA), where misattribution compounds over multiple turns.

**Information-gain rewards are competitive but insufficient.** IGPO improves over ORM on most benchmarks and matches SFT on GAIA, but remains below TPV everywhere. A likely reason is that IGPO requires ground-truth answers to compute $\log \pi(y^\star \mid h)$, making it unusable when answers are unknown and measures progress toward the answer rather than the quality of retrieved evidence.

**Per-turn credit placement outperforms uniform distribution.** The Uni-TPV ablation receives the same total reward as TPV but spreads it evenly across turns. While it improves over ORM on all five benchmarks—confirming that denser credit helps—it still falls short of TPV everywhere, effectively isolating the effect of credit placement from pure reward density. This gap confirms that penalizing unproductive web drift while selectively amplifying decisive retrieval turns is essential, as uniform smoothing still inadvertently reinforces redundant search loops.

**SFT provides the foundation.** RL-only stays close to the base model (21.26% vs. 20.47% on GAIA) and far below SFT (49.61%). Dense rewards alone cannot convey basic output formats or tool-use skills; SFT provides that foundation, upon which TPV delivers secondary gains.

Overall, turn-level externally grounded credit outperforms both outcome-based and information-gain rewards across all benchmarks. Crucially, this gain is driven by precise credit attribution rather than reward density alone.

### 3.3 Validity of Process-Level Verification (RQ2)

The verifier is trained on turn-level grades (A/B/C/D) produced by an LLM judge with access to the ground truth. A natural concern is whether the judge simply reproduces trajectory outcomes, assigning high grades to every step in successful paths and low grades to failed ones. If so, the verifier would act as a pseudo-outcome reward rather than a true process-level signal. Below, we analyze the judge's labeling behavior across three dimensions (detailed statistics in Appendix D).

**Grade distribution.** The most direct test is examining grade distributions within correct and incorrect trajectories on the train split of DeepResearch-9K(Wu et al., 2026). If the judge were dominated by outcome, correct trajectories would consist almost entirely of grade A and incorrect ones of grade D. Instead, correct trajectories contain 45.9% A/B and 54.1% C/D turns, and incorrect ones retain 22.0% A/B turns, confirming that effective steps are recognized even when the overall trajectory fails (Table 2).

Table 2: Grade distribution over turns in correct (Tier-1) vs. incorrect (Tier-2) trajectories.

| Grade | Tier-1 | Tier-2 |
|---|---|---|
| A | 33.58% | 10.30% |
| B | 12.34% | 11.70% |
| C | 50.05% | 69.83% |
| D | 4.04% | 8.16% |
| **Total** | 51,415 | 99,636 |

**Label validity.** We further evaluate whether turn grades track externally verified evidence on BrowseComp-Plus (Chen et al., 2025b), which provides manually annotated evidence documents. Defining $P(\text{evidence} \mid g)$ as the probability that a turn with grade $g$ retrieves at least one gold evidence document, if the judge merely followed trajectory outcomes, $P(\text{evidence} \mid g)$ would be flat across grades. Instead, it aligns monotonically with retrieval quality: $P(\text{evidence} \mid A) = 0.949$ drops steeply to $P(\text{evidence} \mid D) = 0.010$, achieving a turn-level AUC of 0.905. This indicates that grades reflect actual retrieval utility rather than trajectory outcomes (Table 5).

**Privilege ablation.** To test whether grades are merely driven by the teacher seeing the ground truth, we re-judge all turns with the ground-truth answer hidden. Withholding the answer reduces turn-level AUC by 8.6 points (0.906 $\rightarrow$ 0.820), yet per-grade evidence agreement remains remarkably stable: grade A still indicates evidence retrieval 92.8% of the time (vs. 92.0% originally; Table 6). While the verifier yields a lower offline AUC (0.777) due to operating without ground-truth answers, it consistently preserves the crucial within-group rank ordering ($A > B > C$).

### 3.4 Turn Quality Beyond Outcome Supervision (RQ3)

While RQ2 established the validity of turn-level labels, a critical question remains: can a verifier without access to ground-truth answers genuinely distinguish turn quality from trajectory outcomes? We evaluate on 17,581 held-out search and visit turns, excluding

Table 3: Verifier performance on held-out test turns under outcome conflicts. Best in **bold**.

| Verifier | Macro-F1 | Bal. Acc. | $\text{Acc}_{\text{bad}\mid\text{succ}}$ | $\text{Acc}_{\text{good}\mid\text{fail}}$ |
|---|---|---|---|---|
| ORM | 0.293 | 0.339 | NA | NA |
| 0.6B-Base | 0.559 | 0.577 | 0.779 | 0.464 |
| 1.7B-Base | 0.531 | 0.578 | 0.791 | 0.459 |
| 4B-Base | 0.615 | 0.635 | 0.819 | 0.527 |
| 8B-Base | 0.643 | 0.668 | 0.835 | **0.624** |
| 14B-Base | **0.663** | **0.669** | **0.873** | 0.565 |

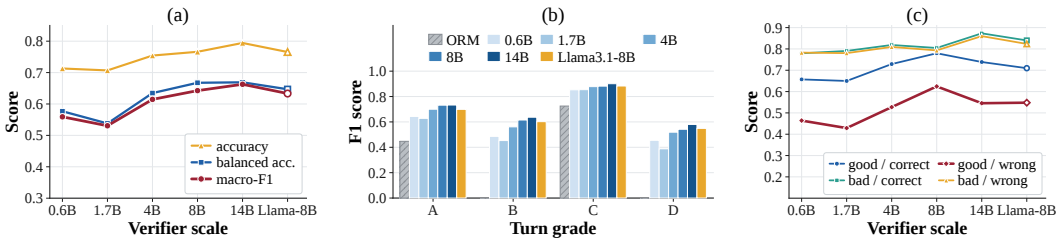



Figure 3: Scaling behavior of the verifier across model sizes. (a) Metrics improve with scale. (b) Per-grade F1 across verifier scales and backbones (c) Accuracy grouped by turn quality and outcome.

final-answer and unparsable turns (Appendix B.2), across macro-F1, balanced accuracy, and two scenarios with mismatched outcomes: bad turns in successful trajectories ($\text{Acc}_{\text{bad|succ}}$) and good turns in failed trajectories ($\text{Acc}_{\text{good|fail}}$; case study in Appendix D.4).

**Results & Analysis.** By uniformly broadcasting outcome rewards, ORM fails on both mismatched scenarios, capping its macro-F1 at 0.293. In contrast, our 8B verifier outperforms ORM in both macro-F1 (0.643) and balanced accuracy (0.668), while achieving strong accuracy under these mismatched settings (0.835 on $\text{Acc}_{\text{bad|succ}}$, 0.624 on $\text{Acc}_{\text{good|fail}}$). This confirms that the verifier directly evaluates turn utility rather than imitating overall outcomes. Notably, 8B achieves the highest $\text{Acc}_{\text{good|fail}}$ (0.624 vs. 0.565 for 14B), excelling at identifying valid turns from failed rollouts.

### 3.5 Scaling Behavior and Cross-Architecture Transfer (RQ4)

**Overall scaling trend (Fig. 3(a)).** Evaluating five Qwen3 sizes (0.6B–14B) on 17,581 test turns shows that discrimination scales consistently with parameter count. Beyond a slight dip at 1.7B, macro-F1 reaches 0.663 at 14B, far surpassing ORM (0.293). Because gains taper off above 8B (macro-F1 0.643), 8B offers the best precision-efficiency trade-off.

**Per-grade discrimination (Fig. 3(b)).** Breaking down performance by grade shows that scale improvements are heavily concentrated on subtle turns. While dominant grades (A and C) approach saturation early at 0.6B, intermediate grades (B and D) drive virtually all scaling gains. In contrast, ORM fails completely on both intermediate classes ($F_1 = 0$).

**Disentangling turn quality from outcome (Fig. 3(c)).** Splitting turns by their label and by whether the trajectory they belong to succeeded confirms that capacity enables the verifier to distinguish local utility from final results. The hardest setting, recovering good turns within failed rollouts (*good / wrong*), improves most with scale, peaking at 0.624 for 8B and suggesting that larger models learn intrinsic turn quality rather than surface correlations with the outcome.

**Cross-architecture transferability.** Process verification transfers across model families. Switching to a Llama-3.1-8B backbone holding the training recipe fixed yields 0.648 macro-F1, matching Qwen3-8B (0.643) within a 0.005 gap and mirroring its per-grade profile (Fig. 3b). This indicates that performance stems from our process supervision rather than biases tied to a particular backbone.

### 3.6 Test-Time Scaling and Reranking Aggregation (RQ5)

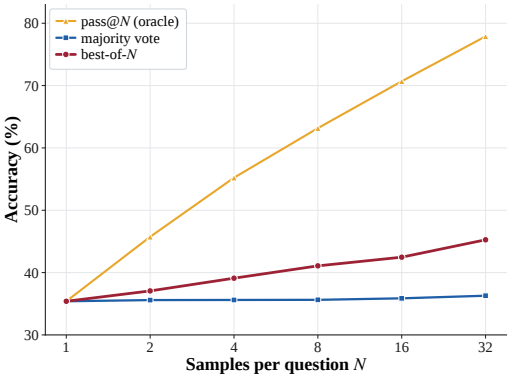



Figure 4: Test-time scaling of the verifier reranking $N$ trajectories

**Test-time scaling.** We freeze the 8B verifier as a trajectory reranker over $N$ sampled rollouts per question. As shown in Figure 4, process reranking converts additional compute into consistent accuracy gains, rising from 35.40% at $N$=1 to 45.26% at $N$=32 (+9.86%). In contrast, majority voting remains nearly flat (36.29%, +0.89%) because long tool-use answer strings rarely match,

so majority votes carry little signal. Against the oracle upper bound ($77.89\%$), the verifier recovers $23.2\%$ of the achievable selection gap, confirming that the primary bottleneck at test time is trajectory selection, not sampling coverage.

**Aggregation and generalization.** To aggregate per-turn scores into a trajectory scalar, we evaluate seven pooling rules across $842$ held-out BrowseComp-Plus trajectories (Appendix D.3). Mean pooling achieves the highest performance and loses only $0.036$ AUC when oracle labels are replaced with verifier predictions ($0.822 \rightarrow 0.785$). Mean pooling is the most noise tolerant: under $40\%$ random label corruption, its AUC degrades least ($-0.059$ vs. $-0.082$ to $-0.141$ for other variants). This robustness stems from its averaging nature: it dilutes misjudged turns across the entire rollout, whereas min, max, or last-3 pooling allow a single noisy turn to dominate the trajectory score.

## 4 Related Works

Process reward modeling for LLM agents has emerged to mitigate the sparse credit assignment of outcome rewards in long-horizon tasks. Unlike standard PRMs designed for structured reasoning (Uesato et al., 2022; Wang et al., 2024) (e.g., mathematical deduction and code synthesis), deep research agents execute open-ended actions (Jin et al., 2025; Zheng et al., 2025b) such as query formulation and evidence extraction. Since outcome rewards are too sparse to guide multi-turn exploration, automated process supervision leverages turn-level feedback to steer policy learning (Xi et al., 2026). We review related works along three paradigms: (1) implicit process rewards, (2) Monte Carlo rollout value estimation, and (3) likelihood and information-gain metrics.

**Implicit process rewards.** Rather than learning an intermediate verifier, implicit process reward methods obtain dense credit by reparameterizing terminal outcomes as implicit rewards (Rafailov et al., 2023; Yuan et al., 2024). PRIME (Cui et al., 2025) iteratively update this implicit reward during reinforcement learning, and iStar (Liu et al., 2026) extends the formulation to multi-turn trajectory optimization. To bypass reward updates, TRACE (Tao et al., 2026) exploits differences in ground-truth log-probabilities under a frozen reference model to extract turn-level credit.

**Monte Carlo rollout estimation.** A classic paradigm estimates intermediate step values by measuring the success rate of sampled continuations, as adopted in OmegaPRM (Luo et al., 2024) and VinePPO (Kazemnejad et al., 2024). Building on this, tree-based reinforcement learning frameworks share prefix rollouts to amortize simulation costs, deriving step-level feedback across branching reasoning paths and agentic retrieval steps (Hou et al., 2025; Ding & Ye, 2026; Zhang et al., 2026). By aggregating outcomes from multiple exploration trajectories originating from the same intermediate state, these methods estimate turn-level values without requiring step-level supervision.

**Likelihood and information-gain metrics.** PAVs (Setlur et al., 2025) reward progress in success probability under a prover policy, while IGPO (Wang et al., 2025) and StepSearch (Zheng et al., 2025a) reward search turns by information gain toward the ground-truth answer or gold documents. These signals are dense and avoid rollout, and are computed against $y^\star$ or gold documents.

**How TPV bridges the gap.** Learned agent verifiers (Chae et al., 2026; Lee et al., 2026) can support test-time scaling (Snell et al., 2024), yet their step-level labels still require annotation or rollouts. TPV instead adopts learning with privileged information: an unprivileged verifier $v_\phi$ is trained on context alone to predict turn scores assigned by a privileged judge. This eliminates branch rollouts, credits productive turns within all-fail groups, and yields a single frozen verifier that both guides RL optimization and serves as a test-time reranker.

## 5 Conclusion

In this work, we propose **TPV**, an independent turn-level process verifier designed for deep research agents. By training an unprivileged verifier on supervision from a privileged judge, TPV evaluates intermediate search, visit, and reasoning actions solely from the interaction context without expensive continuation rollouts. Integrated into policy training via GRPO, TPV delivers dense turn-level credit assignment, notably resolving the zero-signal dilemma on queries where all trajectory rollouts fail. Extensive evaluations across five benchmarks show that TPV accurately estimates turn utility, scales effectively across policy architectures, and outperforms standard outcome-based methods in test-time trajectory reranking, establishing turn-level process verification as a practical foundation for multi-turn search agents.

## AI USE STATEMENT

In this work, we used Claude Opus 5 for dataset cleaning and formatting, assisting with the implementation of secondary experiment scripts, and drafting and editing parts of the manuscript to improve readability. We have not used generative AI tools for core theoretical conceptualization, main experimental design, primary model training, or data analysis and result interpretation, and the remaining required disclosure tasks are not applicable. All AI-assisted text, data processing pipelines, and code snippets were thoroughly reviewed, manually verified, and tested by the authors. We take full responsibility for the final content and claims of this work.

## ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. Our study focuses on turn-level process verification for research agents and utilizes five publicly available academic benchmarks, without collecting private user data or conducting human-subject experiments. Four benchmarks (GAIA, BrowseComp, xbench-DeepSearch, and DeepSearchQA) require live web retrieval, which is performed through third-party search APIs under their standard terms of service; the fifth benchmark, BrowseComp-Plus, operates on a fixed, publicly released corpus. No personally identifiable information (PII) or harmful content was gathered or disseminated in any of these settings.

## REPRODUCIBILITY STATEMENT

All datasets used in this study are publicly available, with sources referenced in the main text. To ensure reproducibility, we provide comprehensive algorithmic workflows and mathematical formulations in Section 2. Full implementation details, including model configurations, reinforcement learning hyperparameters, prompt templates, and evaluation pipelines across all five benchmarks, are documented in Appendix B, and hyperparameter sensitivity analyses are reported in Appendix D. An anonymized repository containing our source code, environment configurations, and evaluation scripts is included in the supplementary materials and is available at https://anonymous.4open.science/r/ICLR2027_2021-E425/.

## REFERENCES

Jose A Arjona-Medina, Michael Gillhofer, Michael Widrich, Thomas Unterthiner, Johannes Brandstetter, and Sepp Hochreiter. Rudder: Return decomposition for delayed rewards. *Advances in Neural Information Processing Systems*, 32, 2019.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Adrien Barbaresi. Trafilatura: A web scraping library and command-line tool for text discovery and extraction. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing: System demonstrations*, pp. 122–131, 2021.

Hyungjoo Chae, Seonghwan Kim, Junhee Cho, Seungone Kim, Seungjun Moon, Gyeom Hwangbo, Dongha Lim, Minjin Kim, Yeonjun Hwang, Minju Gwak, et al. Web-shepherd: Advancing prms for reinforcing web agents. *Advances in Neural Information Processing Systems*, 38:63314–63356, 2026.

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.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, et al. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent. *arXiv preprint arXiv:2508.06600*, 2025b.

Zheng Chu, Xiao Wang, Jack Hong, Huiming Fan, Yuqi Huang, Yue Yang, Guohai Xu, Chenxiao Zhao, Cheng Xiang, Shengchao Hu, et al. Redsearcher: A scalable and cost-efficient framework for long-horizon search agents. *arXiv preprint arXiv:2602.14234*, 2026.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, et al. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*, 2025.

Zheng Ding and Weirui Ye. Treegrpo: tree-advantage grpo for online rl post-training of diffusion models. In *International Conference on Learning Representations*, volume 2026, pp. 10877–10889, 2026.

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

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

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.

Zhenyu Hou, Ziniu Hu, Yujiang Li, Rui Lu, Jie Tang, and Yuxiao Dong. Treerl: Llm reinforcement learning with on-policy tree search. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12355–12369, 2025.

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.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. Vineppo: Unlocking rl potential for llm reasoning through refined credit assignment. 2024.

Jaewoo Lee, Archiki Prasad, Justin Chen, Zaid Khan, Elias Stengel-Eskin, and Mohit Bansal. Prints: Reward modeling for long-horizon information seeking. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 34120–34138, 2026.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 5420–5438, 2025.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Xiaoqian Liu, Ke Wang, Yuchuan Wu, Fei Huang, Yongbin Li, Jianbin Jiao, and Junge Zhang. Agentic reinforcement learning with implicit step rewards. In *International Conference on Learning Representations*, volume 2026, pp. 129271–129291, 2026.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Meiqi Guo, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, et al. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*, 2024.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. Gaia: a benchmark for general ai assistants. In *International Conference on Learning Representations*, volume 2024, pp. 9025–9049, 2024.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. Rewarding progress: Scaling automated process verifiers for llm reasoning. In *International Conference on Learning Representations*, volume 2025, pp. 60808–60838, 2025.

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.

Zhihong Shao, Yuxiang Luo, Chengda Lu, ZZ Ren, Jiewen Hu, Tian Ye, Zhibin Gou, Shirong Ma, and Xiaokang Zhang. Deepseekmath-v2: Towards self-verifiable mathematical reasoning. *arXiv preprint arXiv:2511.22570*, 2025.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Leitian Tao, Baolin Peng, Wenlin Yao, Tao Ge, Hao Cheng, Mike Hang Wang, Jianfeng Gao, and Sharon Li. Trace: Turn-level reward assignment via credit estimation for long-horizon agents. *arXiv preprint arXiv:2607.13988*, 2026.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Venus Team, Sunhao Dai, Yong Deng, Jinzhen Lin, Yusheng Song, Guoqing Wang, Xiaofeng Wu, Yuqi Zhou, Shuo Yang, Zhenzhe Ying, et al. Dr-venus: Towards frontier edge-scale deep research agents with only 10k open data. *arXiv preprint arXiv:2604.19859*, 2026.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process-and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural networks*, 22(5-6):544–557, 2009.

Guoqing Wang, Sunhao Dai, Guangze Ye, Zeyu Gan, Wei Yao, Yong Deng, Xiaofeng Wu, and Zhenzhe Ying. Information gain-based policy optimization: A simple and effective approach for multi-turn llm agents. *arXiv e-prints*, pp. arXiv–2510, 2025.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, 2024.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. Browsecomp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*, 2025.

Tongzhou Wu, Yuhao Wang, Xinyu Ma, Xiuqiang He, Shuaiqiang Wang, Dawei Yin, and Xiangyu Zhao. Deepresearch-9k: A challenging benchmark dataset of deep-research agent. *arXiv preprint arXiv:2603.01152*, 2026.

Zhiheng Xi, Chenyang Liao, Guanyu Li, Zhihao Zhang, Wenxiang Chen, Binghai Wang, Senjie Jin, Yuhao Zhou, Jian Guan, Wei Wu, et al. Agentprm: Process reward models for llm agents via step-wise promise and progress. In *Proceedings of the ACM Web Conference 2026*, pp. 4184–4195, 2026.

Peiran Xu, Zhuohao Li, Xiaoying Xing, Guannan Zhang, Debiao Li, and Kunyu Shi. Hybrid reward normalization for process-supervised non-verifiable agentic tasks. *arXiv preprint arXiv:2509.25598*, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Chenlu Ye, Zhou Yu, Ziji Zhang, Hao Chen, Narayanan Sadagopan, Jing Huang, Tong Zhang, and Anurag Beniwal. Beyond correctness: Harmonizing process and outcome rewards through rl training. *arXiv preprint arXiv:2509.03403*, 2025.

Fei Yu, Anningzhe Gao, and Benyou Wang. Ovm, outcome-supervised value models for planning in mathematical reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 858–875, 2024.

Lifan Yuan, Wendi Li, Huayu Chen, Ganqu Cui, Ning Ding, Kaiyan Zhang, Bowen Zhou, Zhiyuan Liu, and Hao Peng. Free process rewards without process labels. *arXiv preprint arXiv:2412.01981*, 2024.

Wenlin Zhang, Xiangyang Li, Kuicai Dong, Yichao Wang, Pengyue Jia, Xiaopeng Li, Yingyi Zhang, Derong Xu, Zhaocheng Du, Huifeng Guo, et al. Process vs. outcome reward: Which is better for agentic rag reinforcement learning. *Advances in Neural Information Processing Systems*, 38: 58701–58729, 2026.

Xuhui Zheng, Kang An, Ziliang Wang, Yuhang Wang, and Yichao Wu. Stepsearch: Igniting llms search ability via step-wise proximal policy optimization. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21816–21841, 2025a.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 414–431, 2025b.

# Technical Appendix

## TABLE OF CONTENTS

**A Dataset Details** **15**
A.1 GAIA . . . 15
A.2 BrowseComp . . . 15
A.3 xBench-DS-2510 . . . 15
A.4 DeepSearchQA . . . 15
A.5 BrowseComp-Plus . . . 15

**B Implementation details** **16**
B.1 Search Tool: Serper API . . . 16
B.2 Evaluation Set Construction . . . 16
B.3 Reinforcement Learning and Reward Formulation . . . 16

**C Prompts and Input Formats** **17**
C.1 Teacher Grading Prompt . . . 18
C.2 Verifier Input Format . . . 18

**D Supplementary Experimental Analyses** **19**
D.1 Empirical Validity of Privileged Turn Supervision . . . 19
D.2 Ablation on Privileged Ground-Truth Access . . . 19
D.3 Trajectory Reranking: Aggregation Rules and Noise Robustness . . . 19
D.4 Qualitative Case Study: Turn-Level Process Verification in Failed Trajectories . . . 20

# A Dataset Details

We evaluate on five deep-research benchmarks that differ along three axes: answer format (single-value vs. set-valued), language (English vs. Chinese), and retrieval environment (live web vs. local corpus). Table 4 summarizes them.

| Benchmark | Split used | $N$ | Difficulty strata | Lang. | Retrieval |
|---|---|---|---|---|---|
| GAIA | text-only | 127 | L1 (42) / L2 (66) / L3 (19) | en | live web |
| BrowseComp | full | 1266 | — | en | live web |
| xbench-DS-2510 | full | 100 | — | zh | live web |
| DeepSearchQA | full | 900 | — | en | live web |
| BrowseComp-Plus | full | 830 | — | en | local corpus |

Table 4: Evaluation benchmarks. Only GAIA§A.1 has official difficulty levels.

## A.1 GAIA

GAIA (Mialon et al., 2024) is a general AI-assistant benchmark whose questions require multi-step tool use. Because our task focuses on text search, we retain only the questions that are answerable from text and web search, discarding those that ship an attached image, spreadsheet, or audio file. This leaves 127 of the validation questions, distributed over the three released difficulty levels as 42/66/19 for L1/L2/L3. Levels reflect the number of steps and the variety of tools the original authors deemed necessary, not reasoning depth alone.

## A.2 BrowseComp

BrowseComp (Wei et al., 2025) targets questions whose answers are hard to find but easy to verify, constructed by inverting a known answer into an obfuscated multi-hop description. We evaluate on the full 1,266-question benchmark spanning the ten released topic categories: TV shows & movies, Science & technology, Art, History, Sports, Music, Video games, Geography, Politics, and Other. BrowseComp ships no difficulty labels, so we report results only in aggregate and do not use the topic field for training.

## A.3 xBench-DS-2510

xBench-DeepSearch (Chen et al., 2025a) is a Chinese deep-search benchmark refreshed periodically to limit contamination; we use the 2510 release in full (100 questions). It carries no difficulty annotation, so all questions form a single stratum. This is the only all-Chinese benchmark in our suite and therefore the only one that exercises the agent's non-ASCII tool-call path, which we flag as a distinct evaluation condition.

## A.4 DeepSearchQA

DeepSearchQA (Gupta et al., 2026) probes research-style questions across domains. We evaluate on the full 900-question set spanning 15 subject categories, the largest being Politics & Government, Finance & Economics, Science, Geography, and Health. DeepSearchQA carries no released difficulty labels. Unlike the other four benchmarks, a majority of its answers are set-valued rather than single-valued (*Set Answer* vs. *Single Answer*), which makes exact string matching unusable and motivates judge-based scoring throughout.

## A.5 BrowseComp-Plus

BrowseComp-Plus (Chen et al., 2025b) pairs BrowseComp-style questions with a *fixed*, curated corpus of roughly 100K web documents, so retrieval is reproducible and decoupled from the live web. The corpus contains both human-verified evidence documents sufficient to answer each query and mined hard negatives that keep the task challenging. We use all 830 queries against the released corpus of 100,195 passages. Each query carries human-annotated evidence documents (6.1 per

query on average, ranging from 1 to 13 documents), of which 2.9 are marked gold (ranging from 1 to 9 documents).

This benchmark plays a second role in our analysis. Since every retrieved document has a stable identifier, we can log which documents each turn surfaces and opens, and thus measure whether a turn actually acquired annotated evidence. This is what makes the label-validity audit of §3.3 possible: it is the only benchmark in our suite with human-annotated, turn-level ground truth about evidence acquisition, as opposed to only final-answer ground truth.

We serve the corpus locally with a dense retriever, embedding passages with `e5-base-v2` (Wang et al., 2022) and searching a flat inner-product index. The tool interface is byte-identical to the live-web tools (§B.1), including result formatting and content truncation, so the agent is not exposed to a distribution shift in tool output. Since evaluation on this benchmark consumes no search-API quota, the local retriever can be deployed and queried quickly, making repeated experiments practical.

# B Implementation Details

## B.1 Search Tool: Serper API

For all live-web experiments (GAIA, BrowseComp, xBench-DS-2510, and DeepSearchQA), the agent accesses external web search via the Google Serper API.[2] Serper acts as a lightweight wrapper that provides structured, real-time Google search results. Upon receiving a natural language query, the search tool executes an HTTP POST request to the endpoint and extracts the top-10 organic search results, including their titles, site URLs, and text snippets. To prevent context window explosion and maintain standardized input lengths across all benchmarks, we enforce a strict content truncation limit on the raw snippets returned per query. The resulting structured text is then formatted into a standardized JSON response that matches the exact schema used by our local corpus retriever (§A.5), thereby eliminating distribution shifts in tool-call outputs between live-web and local setups.

## B.2 Evaluation Set Construction

DeepResearch-9K is split by trajectory into train, dev, and test; the test split contains 867 trajectories and 18,424 turns in total. For verifier evaluation we exclude two categories of turns, leaving 17,581 search and visit turns.

**Final-answer turns.** Each trajectory ends with a final-answer turn whose grade is a function of the trajectory outcome $R(\tau)$. Including it would leak outcome information into the evaluation and contradict the judge's blindness to $R(\tau)$, so these turns are removed.

**Unparsable turns.** Turns whose tool call fails to parse are already fully penalized by $F_t$ and carry no semantic content for the verifier to grade. They are removed for the same reason they are excluded from verifier training.

Together these two categories account for 843 turns. The remaining 17,581 turns are all search or visit calls drawn from trajectories with ground-truth annotations, and form the basis for all verifier-quality metrics reported in Section 3.

## B.3 Reinforcement Learning and Reward Formulation

**Rule-Based Components in Turn Rewards.** As formulated in Section 2.2, our composite turn reward incorporates two rule-based terms outside the neural verifier $v_\phi$: a novelty factor $N_t$ and a formatting penalty $F_t$.

- **Novelty Factor ($N_t$)**: To discourage the agent from repeatedly issuing identical queries or browsing redundant URLs to exploit intermediate rewards, we maintain a history of executed commands

[2] https://serper.dev

up to turn $t$. $N_t$ is discretized into three tiers based on string and argument matching:

$$N_t = \begin{cases} 1.0, & \text{if } c_t \text{ is a completely new query string or unvisited URL,} \\ 0.3, & \text{if } c_t \text{ exhibits partial overlap with previous calls (e.g., modified keywords),} \\ 0.0, & \text{if } c_t \text{ is an exact duplicate of an earlier action in } h_t. \end{cases} \tag{9}$$

For non-neutral turns ($g \in \{A, B, D\}$), the multiplicative discount $\lambda(N_t) = \frac{1}{2}(1+N_t) \in [0.5, 1.0]$ ensures that rediscovering identical evidence receives at most half the reward of an initial discovery. For neutral turns ($g = C$), the additive term $\eta N_t$ ($\eta = 0.1$) gives a minor bonus to exploratory attempts over redundant loops.

- **Formatting Penalty ($F_t$)**: To enforce strict environment protocol compliance, $F_t$ penalizes syntactic and schema malformations in tool commands:

$$F_t = \begin{cases} 0, & \text{valid syntax and arguments,} \\ 0.5, & \text{valid command format but missing/empty required arguments,} \\ \min\left(1 + \frac{1}{2}m_t,\, 3\right), & \text{unparsable tool call or broken JSON/API schema,} \end{cases} \tag{10}$$

where $m_t$ counts the number of prior unparsable calls within the current trajectory. The penalty escalates with $m_t$ because a static penalty would be normalized away under group standardization once invalid calls dominate an exploration group. Unparsable calls bypass the verifier and directly receive $\hat{r}_t = -F_t$.

**Two-Channel Advantage Standardization and Dynamic Scaling.** Because research trajectories consist of tens of intermediate tool turns alongside a single terminal outcome label, joint standardization would allow the denser process channel to overwhelm the sparse outcome signal. Standardizing both channels independently via Eq. 6 isolates their relative scales.

To balance their contributions before backward discount propagation, we dynamically calibrate the turn-level scale factor $s$:

$$s = \min\left(\frac{\max\left(\overline{|\tilde{A}^{\text{out}}|},\, \delta\right)}{\overline{|\tilde{A}^{\text{turn}}|} + \varepsilon},\; s_{\max}\right), \tag{11}$$

where $\overline{|\tilde{A}^{\text{out}}|}$ and $\overline{|\tilde{A}^{\text{turn}}|}$ denote the mean absolute advantage values across the respective positions within the group. We set the floor to $\delta = 0.3$ and the cap to $s_{\max} = 0.8$. The floor keeps the process channel alive when every rollout in a group fails ($\overline{|\tilde{A}^{\text{out}}|} = 0$), preventing gradient collapse and steering intermediate exploration. Conversely, the cap prevents a lone success from causing disproportionately violent process gradient updates.

**GRPO Objective with Dense Turn Credit.** After backward credit propagation with discount factor $\gamma = 0.95$ (Eq. 7), the turn-level advantage is clipped to $[-c, c]$ (with $c = 3$) and broadcast to all active action tokens within turn $t$, yielding $\hat{A}_{i,k}$ for token $k \in$ turn $t$. The policy is then updated using the clipped surrogate GRPO objective:

$$\mathcal{L}_{\text{RL}}(\theta) = -\mathbb{E}\left[\frac{1}{Z}\sum_{i=1}^{K}\sum_{k=1}^{Z_i}\min\left(w_{i,k}\,\hat{A}_{i,k},\; \text{clip}(w_{i,k},\, 1-\epsilon,\, 1+\epsilon)\,\hat{A}_{i,k}\right)\right] + \beta\,\mathbb{D}_{\text{KL}}\left[\pi_\theta \,\|\, \pi_{\text{ref}}\right], \tag{12}$$

where $w_{i,k} = \frac{\pi_\theta(y_{i,k}|\cdot)}{\pi_{\theta_{\text{old}}}(y_{i,k}|\cdot)}$ denotes the importance sampling ratio, $Z_i$ is the number of unmasked action tokens in trajectory $i$, $Z = \sum_{i=1}^{K} Z_i$ is the total normalization factor, and $\pi_{\text{ref}} = \pi_{\text{SFT}}$. We set the clipping hyperparameter to $\epsilon = 0.2$ and the KL penalty coefficient to $\beta = 0.1$. Maintaining $\beta = 0.1$ prevents the policy from drifting into excessively long or degenerate exploratory loops under dense intermediate reinforcement.

## C Prompts and Input Formats

Both input templates are reproduced verbatim below.

**Teacher Grading Prompt Template**

**System Role & Context:**

```
Role: Turn-Level Research Auditor
Task: Given a question, the ground truth, the research history, and
a new step (action + observation), assign a grade A/B/C/D to the new
step.
```

**Input Context:**

```
[Question]: {question}
[Ground Truth]: {ground_truth}
[Research History (steps 0 to t-1)]: {history}
[New Step t]:
  Action: {action}
  Observation: {observation}
```

**Grading Instructions:**

Knowing the ground truth, assess whether this new step provides information that genuinely moves closer to it.

**A (Critical):** Finds decisive evidence directly related to the correct answer, or confirms a key link in the reasoning chain.

**B (Helpful):** Finds relevant supporting context that aids progress, but does not directly confirm the target answer.

**C (Redundant):** Finds nothing new, repeats prior information, fails, or explores an irrelevant direction.

**D (Harmful):** Introduces misleading, conflicting, or false information that steers the agent away from the correct answer.

**Strict Output Format:**

```
Reply with a single letter (A/B/C/D).
```

Figure 5: The teacher grading prompt template used to label each `search`/`visit` turn in DeepResearch-9K. For the unprivileged ablation, the `[Ground Truth]` block is removed and all other content is unchanged.

**Answer-Blind Verifier Input Sequence Format**

**Input Context:**

```
[Question]: {question}
[Research History]: {history}
[Current Step]:
  Action: {action}
  Observation: {observation}
```

Figure 6: The unprivileged input format fed to the verifier during both training and inference. It shares the same compact history rendering as the teacher prompt but strictly excludes system instructions and ground-truth answers.

### C.1 Teacher Grading Prompt

The teacher grading prompt is queried once per `search`/`visit` turn with `temperature`$= 0$ and `max_tokens`$= 5$; the first character is taken as the grade and anything other than A–D is discarded. Search and visit turns share the template. Inputs are truncated as follows: question 600 characters, ground truth 300, action 400, and current observation 600; the history context keeps the eight most recent turns, each rendered as `Step` $t$`: <action[:80]> -> <observation[:120]>`. For the privilege ablation (Section 3.3), the `[Ground Truth]` block is removed while keeping all other instructions intact.

### C.2 Verifier Input Format

The verifier $v_\phi$ is a sequence classifier rather than an instruction-following model. Its input format contains no instruction text and no ground-truth answer; `history` follows the identical compact rendering as the teacher prompt.

# D SUPPLEMENTARY EXPERIMENTAL ANALYSES

This appendix provides extended empirical evaluations on privileged turn supervision and test-time reranking dynamics, complementing the findings reported in Section 3.

## D.1 EMPIRICAL VALIDITY OF PRIVILEGED TURN SUPERVISION

To verify that the privileged LLM judge assigns grades based on genuine research utility rather than merely hallucinating or echoing the final trajectory outcome, we evaluate its turn-level predictions against the human-annotated ground-truth evidence documents from BrowseComp-Plus (Chen et al., 2025b).

For each query, BrowseComp-Plus specifies *evidence documents* sufficient to deduce the answer, alongside topically relevant *hard negatives*. We define an intermediate tool turn (`search` or `visit`) as evidence-positive if it retrieves at least one verified evidence document. As reported in Table 5, the probability of discovering verified evidence $P(\text{evidence} \mid g)$ decreases monotonically from Grade $A$ (0.949) to Grade $D$ (0.010). The resulting turn-level AUC of 0.905 demonstrates that the privileged judge reliably captures whether an exploratory action advances the investigation toward the ground truth.

Table 5: Alignment between privileged judge grades and human-annotated evidence documents on BrowseComp-Plus. $P(\text{evidence} \mid g)$ denotes the fraction of grade-$g$ turns retrieving verified evidence. Turn-level AUC measures the separation between evidence-positive and evidence-negative actions using the ordinal grade as the score.

| Judge Grade | Evidence Retrieval Precision $P(\text{evidence} \mid g)$ |
|---|---|
| A (*Critical*) | 0.949 |
| B (*Helpful*) | 0.575 |
| C (*Redundant*) | 0.161 |
| D (*Harmful*) | 0.010 |
| Turn-level AUC | 0.905 |

## D.2 ABLATION ON PRIVILEGED GROUND-TRUTH ACCESS

To isolate the necessity of privileged information during data construction, we conduct an ablation by withholding the ground truth $y^\star$ from the teacher judge. The identical backbone model, prompt template, and interaction trajectories are evaluated, omitting strictly the `[Ground Truth]` segment.

As shown in Table 6, stripping privileged ground truth causes the teacher's discriminatory Turn-level AUC to drop sharply from 0.906 to 0.820. Although $P(\text{ev} \mid A)$ slightly rises ($0.920 \rightarrow 0.928$), this is an artifact of decision boundary shrinkage: without ground truth, the judge becomes overly conservative and reserves Grade $A$ exclusively for turns with unmistakable keyword matches, sacrificing recall on nuanced reasoning links and severely degrading global turn-ranking discrimination. For the unprivileged verifier $v_\phi$, per-grade evidence rates are calculated based on its hard predictions $\hat{g} = \arg\max_g p_\phi(g \mid x_t)$, confirming that $v_\phi$ effectively learns the teacher's grading distribution without expensive branch rollouts.

## D.3 TRAJECTORY RERANKING: AGGREGATION RULES AND NOISE ROBUSTNESS

Test-time reranking requires aggregating turn-level utilities into a unified scalar trajectory score $f(\tau)$. We systematically investigate the choice of aggregation function across two key dimensions: rank discrimination and robustness to misclassification noise.

**Aggregation Pooling Strategies.** We evaluate seven distinct pooling rules across 842 held-out research trajectories from the test split (retaining all trajectories containing at least one intermediate tool-calling turn from the full 867 instances): mean, min, max, trailing last-3 mean, sum, $\sqrt{T}$-normalized sum, and exponentially discounted sum ($\gamma = 0.95$). As summarized in Table 7, length-

Table 6: Privilege ablation: impact of withholding ground-truth supervision from the judge during turn labeling. Evidence precision $P(\text{ev} \mid g)$ denotes the proportion of turns assigned grade $g$ that retrieve verified evidence documents. Turn-level AUC measures discriminatory accuracy between evidence-positive and negative turns.

| Setting | Turn-level AUC | $P(\text{ev} \mid A)$ | $P(\text{ev} \mid B)$ | $P(\text{ev} \mid C)$ |
|---|---|---|---|---|
| Teacher + Privileged Gold | 0.906 | 0.920 | 0.525 | 0.110 |
| Teacher − Unprivileged | 0.820 | 0.928 | 0.534 | 0.140 |
| Verifier ($v_\phi$, Context-Only) | 0.777 | 0.844 | 0.568 | 0.154 |

normalized mean pooling achieves the strongest trajectory-level discrimination ($0.822$ AUC under oracle labels). Replacing oracle teacher grades with predictions from the frozen verifier $v_\phi$ degrades trajectory AUC by only $0.037$ ($0.822 \rightarrow 0.785$), demonstrating that the vast majority of the credit assignment signal translates directly to test-time trajectory reranking.

Table 7: Robustness of pooling strategies to uniform random label corruption on BrowseComp-Plus. Each cell reports the trajectory-level AUC averaged over five random seeds. Corruption denotes the percentage of intermediate tool turns within each trajectory whose grades are replaced with a uniformly random grade $\{A, B, C, D\}$. *Norm. Sum* represents $\sqrt{T}$-normalized sum; *Disc.* denotes exponentially discounted sum with $\gamma = 0.95$.

| Noise Level | Mean | Min | Last-3 | Max | Sum | Norm. Sum | Disc. |
|---|---|---|---|---|---|---|---|
| 0% | 0.822 | 0.754 | 0.778 | 0.603 | 0.542 | 0.715 | 0.635 |
| 20% | 0.792 | 0.708 | 0.736 | 0.518 | 0.473 | 0.655 | 0.583 |
| 40% | 0.763 | 0.662 | 0.696 | 0.500 | 0.401 | 0.583 | 0.514 |
| $\Delta$ | $-0.059$ | $-0.092$ | $-0.082$ | $-0.103$ | $-0.141$ | $-0.132$ | $-0.121$ |

**Robustness to Turn Label Noise.** To evaluate whether mean pooling remains robust when intermediate verifier scores contain prediction noise, we inject uniform random corruption into turn labels. At each corruption level ($20\%$ or $40\%$), the corresponding percentage of intermediate tool turns in each trajectory are randomly reassigned one of the four grades $\{A, B, C, D\}$. As shown in Table 7, mean pooling exhibits superior noise resilience, dropping by only $-0.059$ AUC from $0\%$ to $40\%$ noise, compared to substantial degradations of $-0.082$ to $-0.141$ observed in non-linear or unnormalized baselines. Averaging inherently attenuates isolated turn mispredictions across long horizons, whereas min or max pooling allows single outlier errors to dominate the entire trajectory evaluation.

### D.4 Qualitative Case Study: Turn-Level Process Verification in Failed Trajectories

To intuitively illustrate how TPV provides effective credit assignment where conventional outcome-based reward models (ORMs) fail, Figure 7 presents a representative research trajectory on a deep research query from the validation set.

In this trajectory, although the agent performs valid exploratory turns, its investigation fails due to an inverted temporal reasoning step during its final retrieval turn, leading to an incorrect final answer with terminal reward $R(\tau) = 0$. Under standard outcome-based RL (ORM), this terminal failure broadcasts an indiscriminate non-positive advantage across all intermediate turns, actively suppressing productive exploratory actions alongside flawed ones. In contrast, TPV assesses each intermediate `search` and `visit` turn conditioned purely on interaction context, accurately isolating the turn-level utility of intermediate actions:

- **Turn 1 (Suboptimal Exploration):** The agent attempts to search for film release dates to gather general entity context rather than targeting the directors' identities. TPV detects this semantic drift and predicts turn-level grade **C** (*Redundant*), assigning a modest baseline reward of $\hat{r}_1 = \eta N_1 =$

$+0.10$ (with novelty $N_1 = 1.0$), which avoids rewarding off-target retrieval while preserving benign exploratory attempts.

- **Turn 2 (Decisive Discovery):** Correcting its path, the agent issues a precise search to retrieve John G. Adolfi's birth year, acquiring critical grounding evidence. TPV awards this decisive progress turn-level grade **A** (*Critical*), yielding a high positive reward of $\hat{r}_2 = \nu(A) \cdot \lambda(N_2) = +1.00$. Even inside a globally failed trajectory, this valid exploratory action is explicitly protected and reinforced.
- **Turn 3 (Misguided Reasoning in Tool Turn):** Although the agent retrieves Charles J. Hunt's birth year via search, its internal reasoning step conflates "born earlier" with "younger", introducing a corrupted deduction that derails subsequent answer derivation. TPV penalizes this misleading turn with turn-level grade **D** (*Harmful*), assigning $\hat{r}_3 = \nu(D) \cdot \lambda(N_3) = -0.50$.

After backward credit propagation (Eq. 7), the resulting turn-level advantages successfully differentiate the productive search in Turn 2 from the flawed deduction in Turn 3, directly demonstrating how TPV resolves the credit assignment bottleneck on failed rollouts.

**Case Study: Turn-Level Process Verification vs. ORM on a Failed Deep Research Trajectory**

**Question:** Which film whose director is younger, *College Lovers* or *The Dixie Flyer*?
**Ground Truth ($y^\star$):** *College Lovers* **Trajectory Outcome ($R(\tau)$): 0 (Incorrect)**

**Turn 1: Suboptimal Exploration** **ORM Signal:** $\hat{A}_1^{\text{out}} \leq 0$ (Uniform Indiscriminate Penalty)
**[Thought]:** To answer this question, I first need to gather general context and check the release dates of both films...
**[Action]:** `search("release date College Lovers film")`
**[Observation]:** The release date of *College Lovers* is October 5, 1930...

**TPV Turn-Level Evaluation:** Grade **C** (*Redundant*) Novelty $N_1 = 1.0$, Penalty $F_1 = 0$
$\rightarrow$ **Turn-Level Reward:** $\hat{r}_1 = \eta N_1 - F_1 = 0.1 \times 1.0 - 0 = +\mathbf{0.10}$ *(Benign exploration preserved)*

**Turn 2: Valid Evidence Discovery** **ORM Signal:** $\hat{A}_2^{\text{out}} \leq 0$ (Uniform Indiscriminate Penalty)
**[Thought]:** Next, I need to identify the directors. The director of *College Lovers* is John G. Adolfi; I should find his birth year...
**[Action]:** `search("John G. Adolfi director birth year")`
**[Observation]:** John G. Adolfi was born on February 19, 1888...

**TPV Turn-Level Evaluation:** Grade **A** (*Critical*) Novelty $N_2 = 1.0$, Penalty $F_2 = 0$, $\lambda(N_2) = \frac{1}{2}(1 + 1.0) = 1.0$
$\rightarrow$ **Turn-Level Reward:** $\hat{r}_2 = \nu(A) \cdot \lambda(N_2) - F_2 = 1.0 \times 1.0 - 0 = +\mathbf{1.00}$ ***(Decisive exploration reinforced)***

**Turn 3: Corrupted Reasoning in Tool Turn** **ORM Signal:** $\hat{A}_3^{\text{out}} \leq 0$ (Uniform Indiscriminate Penalty)
**[Thought]:** Now I search for the director of *The Dixie Flyer* (Charles J. Hunt) to compare ages...
**[Action]:** `search("Charles J. Hunt director birth year")`
**[Observation]:** Charles J. Hunt was born on April 8, 1881...
**[Intermediate Inference]:** Charles J. Hunt was born in 1881. Since Hunt was born earlier, he is younger than Adolfi (born 1888).

**TPV Turn-Level Evaluation:** Grade **D** (*Harmful*) Novelty $N_3 = 1.0$, Penalty $F_3 = 0$, $\lambda(N_3) = 1.0$
$\rightarrow$ **Turn-Level Reward:** $\hat{r}_3 = \nu(D) \cdot \lambda(N_3) - F_3 = -0.5 \times 1.0 - 0 = -\mathbf{0.50}$ ***(Fallacious reasoning penalized)***

Figure 7: Qualitative walkthrough of turn-level TPV scoring on a failed deep research trajectory from BrowseComp-Plus. When trajectories fail ($R(\tau) = 0$), standard ORM uniformly suppresses all intermediate actions ($\hat{A}_t^{\text{out}} \leq 0$). In contrast, TPV operates strictly over tool-calling turns (`search`/`visit`), crediting the breakthrough evidence retrieval in Turn 2 ($\hat{r}_2 = +1.00$) while penalizing the misleading deduction in Turn 3 ($\hat{r}_3 = -0.50$), thereby steering policy learning with fine-grained turn-level credit attribution.