FALSIFY BEFORE YOU AMPLIFY: CONSERVATIVE CROSS-FITTED CREDIT FOR SELF-EVOLVING LANGUAGE AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Hindsight skills provide dense supervision for language agents, but explaining a completed trajectory does not establish which actions deserve credit. When a skill is evaluated on the trajectory that produced it, retrospective fit can be mistaken for useful guidance. We introduce EVOCRED, a reinforcement learning method that uses hindsight to refine credit allocation within budgets established by advantages computed from task returns. A frozen policy scores each target's fixed actions with and without a skill generated from another rollout of the same task. Both scoring branches retain the target's original history and response-token prefixes, isolating the effect of skill conditioning. Evidence strength and compatibility with the base advantages govern whether, and to what extent, the skill influences credit allocation. Admitted evidence is standardized within each trajectory and mapped to bounded allocation scores. The resulting positive, bounded multipliers preserve token-level signs, while credit-weighted centering preserves each trajectory's total absolute credit. This construction allows hindsight to distinguish action contributions while keeping its influence anchored to outcome-derived credit. Across ALFWorld, Search-QA, and WebShop with three Qwen backbones, EVOCRED ranks first in ten of twelve aggregate comparisons and outperforms GRPO, SEED, and AgentOPSD in all twelve. On Qwen3-1.7B, it raises ALFWorld macro success from 42.1% to 91.2% and WebShop success from 38.3% to 76.5% relative to GRPO. These gains require no additional environment rollouts or inference-time skills.

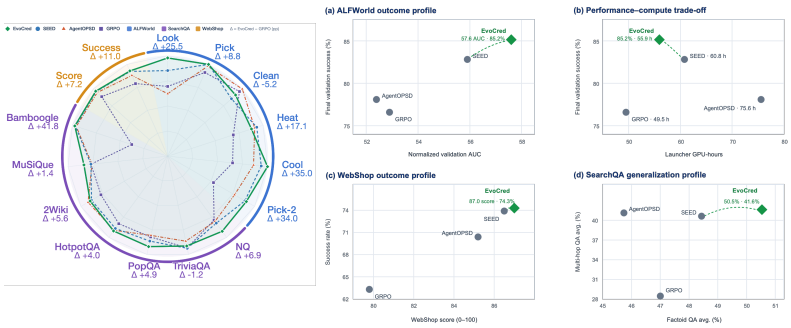


Figure 1: **Performance overview.** The radar and panels (c–d) summarize task-level outcomes for Qwen2.5-3B; panels (a–b) report the separate ALFWorld learning–compute study. *SearchQA* denotes the Search-QA suite. Panel definitions and per-axis radar scales are given in Appendix D.

1 INTRODUCTION

A completed agent trajectory contains information that its final reward alone cannot convey: intermediate observations, action dependencies, and failure patterns that may guide future decisions Li et al. (2024). Recent language-agent methods organize such experience into reusable skills, allowing the agent's knowledge to evolve alongside its policy. SkillRL develops a skill library that co-evolves

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.

with reinforcement learning, while SEED converts completed trajectories into hindsight skills that provide dense supervision during training and are absent at deployment Wu et al. (2026); Wang et al. (2026b); Huang et al. (2024). These approaches offer a promising way to connect sparse outcome feedback with individual decisions in long-horizon interaction Xia et al. (2026); Shi et al. (2026).

Yet a skill's ability to explain a completed trajectory does not establish which actions deserve credit. Consider a skill extracted from a trajectory and used to re-score its actions. Higher action likelihood may reflect useful guidance, but it may also reflect reconstruction of source-specific actions and observations rather than guidance that transfers beyond that trajectory. The trajectory constructs the teacher and justifies its influence on learning Dwork et al. (2015); Kriegeskorte et al. (2009); Jiang et al. (2026b). We call this source-trajectory resubstitution: retrospective compatibility can be mistaken for evidence that a skill should influence credit assignment. This ambiguity can persist as skills and policies co-evolve, with hindsight shaping the policy that generates subsequent experience.

This raises a question about dense supervision: what should a trajectory-derived skill demonstrate before modifying credit? SEED turns evolving hindsight into token-level learning signals, while AgentOPSD develops recursive evidence accumulation and bounded advantage reshaping for sequential credit assignment Wu et al. (2026); Wang et al. (2026b). Our focus is the signal's admissibility and scope: whether support extends beyond the skill's construction trajectory, whether it agrees with base credit, and how much influence it should receive. Temporal aggregation and bounded reweighting alone do not resolve direct reuse of the trajectory that authored the skill Amini et al. (2025); Munkhbat et al. (2025).

We introduce EvoCred, a reinforcement learning method that separates the evidence establishing credit from the evidence redistributing it. Task returns, combining verifier outcomes and prescribed penalties, determine base advantages that fix the signs and total absolute magnitude of credit within each trajectory; hindsight may only redistribute this budget. Generated skills provide candidate allocation evidence under three complementary constraints Gallitto et al. (2025); Jiang et al. (2026a). First, EvoCred evaluates skills away from their source trajectories. Each target receives a skill from another rollout of the same task, assigned before skill-conditioned scores are observed. A frozen policy compares the likelihoods of the target's unchanged actions with and without that skill while keeping the original target histories and response-token prefixes fixed. Reusing existing rollout groups excludes direct source reuse without additional environment interaction. Here, cross-fitting denotes source exclusion; shared tasks and policy snapshots do not provide independent validation or causal identification Zhang et al. (2025); Tu et al. (2026); Zhou et al. (2025).

Second, EvoCred controls admission through evidence strength and directional agreement with base advantages. Weak evidence receives limited influence; insufficient agreement prevents redistribution. Third, admitted evidence defines positive, bounded token-advantage multipliers. Positivity preserves signs; centering transformed allocation scores under the absolute base-advantage measure gives the multipliers unit weighted mean, preserving each trajectory's total absolute credit. These guarantees constrain optimization coefficients without assuming that admitted skills are correct. Skill generation and scoring operate only during training; deployment requires no skill access Tran et al. (2025).

We evaluate EvoCred on ALFWorld Shridhar et al. (2020), Search-QA Jin et al. (2025), and WebShop Yao et al. (2022) using Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, and Qwen3-1.7B-Instruct. Across four aggregate metrics per backbone, EvoCred ranks first in 10 of 12 comparisons in Table 1 and exceeds GRPO, SEED, and AgentOPSD in all 12. On Qwen3-1.7B, it improves ALFWorld macro success from 42.1% to 91.2% and WebShop success from 38.3% to 76.5% relative to GRPO. Component ablations and corrupted-skill interventions examine source exclusion, admission, and credit conservation Qwen et al. (2025); Yang et al. (2025).

Our contributions are threefold:

- We identify source-trajectory resubstitution as a source of ambiguity in assigning credit from trajectory-derived hindsight skills.
- We introduce evaluation on sibling trajectories and verifier-aligned admission to control whether, and to what extent, hindsight may influence credit allocation.
- We formulate conservative redistribution with exact guarantees on advantage signs, credit support, and per-trajectory absolute credit mass, and evaluate the resulting method across three agentic task suites and three model backbones.

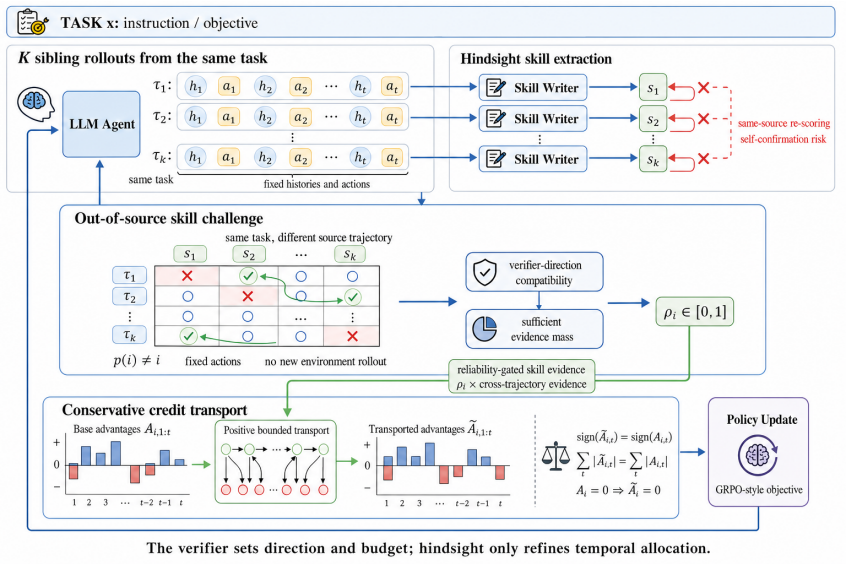


Figure 2: EvoCred**: evaluate before redistributing credit.** A score-blind sibling assignment excludes same-source rescoring. Evidence strength and verifier compatibility determine admission; credit-weighted centering preserves advantage signs and absolute mass. The actor is updated with skill-free histories. All skill-side operations are training-only.

2 Method

EvoCred separates credit authority from allocation: task-return advantages fix credit signs and each trajectory's absolute-credit budget; sibling hindsight may only redistribute it (Figure 2).

2.1 Source-Excluded Evaluation

For each task x, a frozen actor π_{θ^-} samples $K \geq 2$ sibling trajectories τ_i. Returns R_i combine outcome rewards with prescribed invalid-action penalties. All three benchmarks use $A_i^0 = (R_i - \bar{R}_x)/(\hat{\sigma}_x + \epsilon_A)$, where $\bar{R}_x$ and $\hat{\sigma}_x$ are the group mean and standard deviation. Token credit is $A_{ik\ell}^0 = m_{ik\ell} A_i^0$ for turn k, response token ℓ, and binary optimization mask m. Let $M_i^A = \sum_{k,\ell} |A_{ik\ell}^0|$ and, for $M_i^A > 0$, $\mu_{ik\ell} = |A_{ik\ell}^0|/M_i^A$; otherwise retain $\rho_i = 0$, $g_i = 1$, and $\widetilde{A}_i = A_i^0 = 0$ Shao et al. (2024).

A frozen writer produces $s_i = G_{\theta^-}(x, \tau_i, S_i)$ from the episode success label $S_i \in \{0, 1\}$ and trajectory. Each target receives $s_{p(i)}$ with $p(i) \neq i$; donors may be reused. Assignment is outcome-aware and *score-blind*, fixed before skill-conditioned target scores are observed. The frozen actor scores identical target actions and prefixes with and without the skill; no action is regenerated. Within each turn, masked token log-likelihood shifts are summed, clipped to $[-8, 8]$, and accumulated in log-odds space with decay $\gamma_b = 0.95$. A sigmoid maps the accumulated state to B_{ik}, with $B_{i0} = 0.5$. Revisions are $\Delta B_{ik} = B_{ik} - B_{i,k-1}$, with $\Delta B_{ik\ell} = m_{ik\ell} \Delta B_{ik}$ for token-level use. The support state is not a calibrated success probability. Appendix B specifies the scores and recursion Yang et al. (2026c); Schulman et al. (2017).

2.2 Admissible Credit Redistribution

Admission. Align evidence with base credit: $q_{ik\ell} = \operatorname{sgn}(A_{ik\ell}^0)\Delta B_{ik\ell}$. Define strength $m_i = \mathbb{E}_{\mu_i}|\Delta B_i|$ and compatibility $C_i = \mathbb{E}_{\mu_i}[q_i]/m_i$ for $m_i > 0$, otherwise $C_i = 0$. The gate is defined by

$$\rho_i = \left[\frac{C_i - \eta}{1 - \eta}\right]_+ \frac{m_i}{m_i + \kappa} \in [0, 1], \tag{1}$$

Algorithm 1 EvoCred training

Require: Task x; frozen actor and writer; verifier $\mathcal{V}$; $K \geq 2$.
Sample K siblings; retain rollout log-probabilities.
Compute success labels, returns, base credit, and M_i^A; initialize $\rho_i = 0$, $g_i = 1$, $\widetilde{A}_i = A_i^0$.
Fix a score-blind donor map $p(i) \neq i$, allowing reuse; generate one skill per rollout from its task, trajectory, and success label.
for each i with $M_i^A > 0$ **do**
  Evaluate: rescore fixed actions with $s_{p(i)}$; sum and clip token shifts per turn, then accumulate support from $B_{i0} = 0.5$ to obtain ΔB_i.
  Admit: compute μ_i, q_i, (m_i, C_i), and ρ_i.
  Redistribute: standardize q_i across turns, apply $\tanh$, and center u_i under μ_i; compute g_i and $\widetilde{A}_i$.
end for
Update the skill-free actor with detached hindsight quantities.

where $\eta \in [0, 1)$ rejects insufficient compatibility, $\kappa > 0$ attenuates weak evidence, and $[u]_+ = \max(u, 0)$.

Allocation. Standardize aligned turn revisions $q_{ik} = \mathrm{sgn}(A_i^0)\Delta B_{ik}$ and apply a bounded transform:

$$z_{ik} = \frac{q_{ik} - \widehat{q}_i}{\sqrt{v_i + 10^{-6}}}, \qquad u_{ik\ell} = m_{ik\ell} \tanh(z_{ik}),$$

where $\widehat{q}_i$ and v_i are the within-trajectory arithmetic turn mean and population variance. Admission retains raw q; allocation centers u under μ_i:

$$\begin{aligned} \bar{u}_i &= \mathbb{E}_{\mu_i}[u_i], & d_{ik\ell} &= (u_{ik\ell} - \bar{u}_i)/2, \\ g_{ik\ell} &= 1 + \lambda \rho_i d_{ik\ell}, & \widetilde{A}_{ik\ell} &= A^0_{ik\ell} g_{ik\ell}, \end{aligned} \tag{2}$$

with $\lambda \in [0, 1)$. Redistribution favors above-average allocation scores while preserving the budget. Centering under μ_i accounts for unequal response lengths Lu et al. (2026); Du et al. (2020); Amini et al. (2025); Xi et al. (2025); Zhang et al. (2025).

This *conservative change of measure* gives $g_{ik\ell} \in [1 - \lambda\rho_i, 1 + \lambda\rho_i]$ and

$$\begin{aligned} \mathrm{sgn}(\widetilde{A}_{ik\ell}) &= \mathrm{sgn}(A^0_{ik\ell}), \\ \sum_{k,\ell} |\widetilde{A}_{ik\ell}| &= \sum_{k,\ell} |A^0_{ik\ell}|. \end{aligned} \tag{3}$$

Even misleading evidence preserves signs and total absolute credit. Rejection or constant aligned turn revisions leaves base credit unchanged. Appendix B.7 gives the proof and coefficient-change bound.

2.3 Training Procedure and Cost

We substitute $\widetilde{A}$ into the unchanged, globally token-averaged GRPO surrogate (Eq. equation 13), using skill-free histories and detached hindsight quantities. Algorithm 1 summarizes one task group Shao et al. (2024); Yu et al. (2025).

Reusing rollout log-probabilities requires K skill generations and at most K conditional target evaluations per group: $O(K)$ scoring pairs rather than $O(K^2)$, with cost also depending on sequence lengths. No additional environment rollouts are needed; deployment uses only the actor.

3 Experiments

Our experiments examine the effectiveness of EvoCred across tasks and model backbones, the development of its gains during training, and the role of its credit constraints. We combine comparisons against outcome-based RL and skill-based learning with component controls, evidence diagnostics, and a corrupted-skill intervention.

Table 1: **Task-level and aggregate performance across three backbones.** Results are grouped into ALFWorld, Search-QA, and WebShop. The two Avg. columns are unweighted macro averages over the six ALFWorld categories and seven QA datasets, respectively. Higher is better. Bold and underline indicate the best and second-best displayed values within each backbone, including ties.

	ALFWorld							Search-QA								WebShop	
Method	Look	Pick	Clean	Heat	Cool	Pick2	Avg.	NQ	Triv	Pop	Hotp	2Wk	MuS	Bam	Avg.	Score	Succ.
Qwen2.5-3B-Instruct																	
Vanilla	11.1	44.4	6.2	15.4	28.6	12.5	19.7	24.6	48.1	31.0	26.3	25.3	7.2	59.7	31.7	6.7	0.8
Skill-Fewshot	66.7	51.7	48.4	0.0	4.3	10.0	30.2	23.7	46.2	30.6	24.4	22.1	7.5	12.5	23.9	0.2	0.8
GRPO	62.5	91.2	96.2	61.9	65.0	47.4	70.7	39.3	60.6	41.1	37.4	34.6	15.4	26.4	36.4	79.8	63.3
GRPO+OPSD	82.4	**100.0**	85.7	75.0	70.0	60.0	78.9	44.9	**61.2**	45.2	40.4	38.5	16.0	66.1	44.6	77.8	66.4
Skill+GRPO	71.4	88.9	58.8	70.6	40.7	29.2	59.9	43.5	58.8	43.0	36.8	32.2	11.7	12.5	34.1	77.3	60.9
RLSD	75.0	87.9	90.9	75.0	73.1	68.4	78.4	41.5	58.6	42.3	40.4	40.2	**16.8**	66.9	43.8	84.4	66.4
SDAR	62.5	97.1	**100.0**	61.9	75.0	**84.2**	80.1	44.8	58.1	44.3	38.6	36.2	15.7	66.1	43.4	85.0	68.0
SEED	76.7	**100.0**	85.5	**84.0**	93.7	80.0	86.7	41.2	60.9	43.2	40.0	39.5	15.4	67.7	44.0	86.5	73.9
AgentOPSD	56.0	98.0	**100.0**	82.0	86.0	69.2	81.9	40.1	56.2	40.3	41.0	**41.3**	15.2	67.1	43.0	85.2	70.4
EvoCred (Ours)	**88.0**	**100.0**	91.0	79.0	**100.0**	81.4	**89.9**	**46.2**	59.4	**46.0**	**41.4**	40.2	**16.8**	**68.2**	**45.5**	**88.4**	**74.3**
Qwen2.5-7B-Instruct																	
Vanilla	22.2	36.1	3.1	0.0	0.0	0.0	10.2	25.2	50.8	29.5	29.0	29.0	10.4	63.7	33.9	5.9	1.6
Skill-Fewshot	50.0	51.7	32.3	5.3	4.3	0.0	23.9	30.9	52.1	32.7	32.7	27.9	12.7	66.1	36.4	1.7	0.8
GRPO	87.5	91.2	96.2	81.0	65.0	57.9	79.8	45.1	63.7	44.0	43.6	43.2	16.8	37.6	42.0	80.9	72.6
GRPO+OPSD	60.0	50.0	22.7	21.4	17.6	9.5	30.2	8.8	8.6	17.5	2.5	4.2	0.5	1.2	6.2	4.5	2.3
Skill+GRPO	66.7	88.5	65.2	61.1	57.7	73.1	68.7	45.2	63.7	45.7	43.1	43.3	19.6	21.4	40.3	80.4	71.9
RLSD	87.5	**100.0**	92.3	58.8	80.0	65.2	80.6	46.8	63.0	44.4	45.5	**48.9**	**21.5**	**73.0**	49.0	87.4	77.3
SDAR	75.0	94.7	**100.0**	86.7	68.2	78.9	83.9	46.3	63.5	48.2	43.8	48.4	19.6	**73.0**	49.0	89.4	**82.8**
SEED	**100.0**	**100.0**	98.5	89.4	70.1	77.8	89.3	46.5	**65.4**	47.1	44.8	46.0	19.3	68.3	48.2	89.7	78.1
AgentOPSD	83.3	91.2	99.2	**94.8**	78.6	82.7	88.3	45.9	63.7	48.4	43.9	44.7	20.8	67.2	47.8	88.4	79.3
EvoCred (Ours)	**100.0**	**100.0**	**100.0**	86.7	**100.0**	**90.2**	**96.2**	**48.2**	64.1	**49.0**	**46.0**	44.9	21.2	**73.0**	**49.5**	**91.2**	81.2
Qwen3-1.7B-Instruct																	
Vanilla	22.2	25.0	3.1	0.0	21.4	4.2	12.7	29.4	46.9	37.0	23.5	19.6	6.4	10.5	24.8	46.5	4.7
Skill-Fewshot	50.0	10.3	16.1	0.0	0.0	5.0	13.6	29.4	46.5	36.2	22.9	20.8	4.3	10.1	24.3	23.0	2.3
GRPO	41.7	71.1	36.4	40.0	31.8	31.6	42.1	40.0	**58.9**	43.5	35.4	30.3	12.0	65.7	40.8	67.3	38.3
Skill+GRPO	54.5	27.6	22.7	27.3	0.0	19.2	25.2	39.2	58.6	43.9	35.2	28.2	11.5	66.1	40.4	73.4	46.1
RLSD	37.5	50.0	61.5	19.0	50.0	21.1	39.9	38.6	57.3	43.0	34.5	34.1	11.5	65.3	40.6	74.0	50.8
SDAR	25.0	73.5	76.9	33.3	40.0	36.8	47.6	39.7	**58.9**	45.3	**35.9**	**35.5**	**12.6**	65.3	**41.9**	76.8	58.6
SEED	98.7	95.8	**99.4**	82.1	81.5	82.5	90.0	39.4	56.1	45.8	34.2	33.6	11.4	66.0	40.9	84.1	72.6
AgentOPSD	96.8	93.5	98.2	77.4	**86.0**	74.9	87.8	38.1	54.7	43.6	35.8	32.4	9.8	65.1	39.9	85.0	68.0
EvoCred (Ours)	**99.2**	**97.1**	98.9	**83.4**	83.0	**85.6**	**91.2**	**40.2**	55.6	**46.7**	35.0	34.1	10.9	**66.8**	41.3	**86.7**	**76.5**

3.1 Evaluation Setup

Tasks and metrics. We evaluate three complementary forms of agentic interaction. **ALFWorld** Shridhar et al. (2020) tests goal-directed manipulation across Look, Pick, Clean, Heat, Cool, and Pick2; we report category success rates and their unweighted macro average. **Search-QA** Jin et al. (2025) evaluates question answering on Natural Questions, TriviaQA, PopQA, HotpotQA, 2WikiMultihopQA, MuSiQue, and Bamboogle, reporting each dataset score and the unweighted seven-dataset mean. **WebShop** Yao et al. (2022) evaluates interactive shopping using task score and success rate, capturing graded task performance and complete task fulfillment, respectively.

Backbones and comparison methods. We use Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, and Qwen3-1.7B-Instruct. The Qwen2.5 models test the method at two scales within one family; Qwen3-1.7B evaluates it on a compact backbone from another family. All comparisons are made within each backbone. Vanilla supplies the unadapted policy reference, and Skill-Fewshot tests skill-based prompting. GRPO provides the outcome-based RL reference. OPSD Zhao et al. (2026), Skill+GRPO, RLSD Yang et al. (2026b), SDAR Lu et al. (2026), SEED Wu et al. (2026), and AgentOPSD Wang et al. (2026b) compare alternative ways of using skills and distillation for learning. Table 1 reports the evaluated methods for each backbone.

Reward and reporting. For WebShop, EvoCred uses a terminal success reward of 10 and a penalty of 0.1 per invalid-action turn, without episode-length normalization. The continuous environment score is used to report WebShop Score. Success rates are percentages, and WebShop Score is reported on a 0–100 scale. We express success-rate gains in percentage points (pp) and other score differences in points.

Table 2: **Component ablation.** Qwen2.5-3B; final success and success AUC (%). WebShop also reports score.

	ALFWorld		WebShop		
Variant	Final	AUC	Score	Succ.	AUC
GRPO	70.2	49.6	79.5	63.0	44.8
Self-source	84.3	60.7	84.6	70.1	52.2
w/o admission	86.2	61.9	85.4	71.6	53.7
w/o centering	87.1	61.1	85.9	72.0	52.8
Shuffled alloc.	81.6	57.4	83.2	68.6	49.9
EvoCred	**89.6**	**65.4**	**86.8**	**74.0**	**56.3**

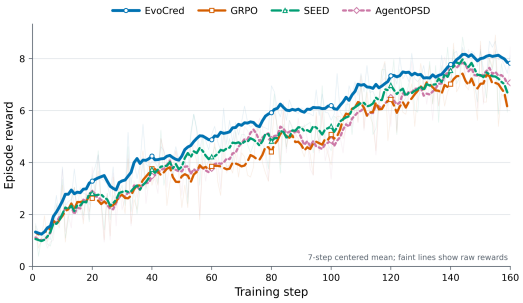


Figure 3: **Training reward.** EvoCred versus GRPO, SEED, and AgentOPSD. Thick curves show seven-step centered means; faint traces show raw rewards.

3.2 Main Results

EvoCred improves every aggregate comparison against the key baselines. Across three backbones and four aggregate metrics, EvoCred outperforms GRPO, SEED, and AgentOPSD in all 12 comparisons (Table 1). It ranks first in ten comparisons and second in the remaining two, achieving the highest ALFWorld macro success and WebShop Score on every backbone. On Qwen2.5-3B, it leads all four metrics, including 89.9% ALFWorld success, 45.5 QA macro score, 88.4 WebShop Score, and 74.3% WebShop success.

The gains extend beyond existing hindsight methods. On Qwen2.5-7B, EvoCred reaches 96.2% ALFWorld macro success, exceeding SEED's 89.3% and AgentOPSD's 88.3% by 6.9 and 7.9 pp. On Qwen3-1.7B, WebShop success reaches 76.5%, improving over SEED and AgentOPSD by 3.9 and 8.5 pp, respectively. These gains show that the complete credit construction improves on both evolving skill supervision and recursive evidence-based reweighting.

Interactive task completion improves across model scales. Relative to GRPO, ALFWorld macro success increases by 19.2, 16.4, and 49.1 pp on Qwen2.5-3B, Qwen2.5-7B, and Qwen3-1.7B; WebShop success increases by 11.0, 8.6, and 38.2 pp. The largest gains occur on the compact backbone, where ALFWorld improves from 42.1% to 91.2% and WebShop success from 38.3% to 76.5%. The same design therefore benefits interactive task completion at both model scales and on the compact Qwen3 backbone.

QA and shopping reveal distinct performance profiles. EvoCred achieves the highest QA macro scores on both Qwen2.5 backbones, improving over GRPO by 9.1 and 7.5 points. On Qwen3-1.7B, its score of 41.3 exceeds SEED's 40.9 and AgentOPSD's 39.9, while SDAR leads at 41.9. For Qwen2.5-7B WebShop, EvoCred achieves the highest Score at 91.2; SDAR leads in success rate at 82.8%, compared with EvoCred's 81.2%. The strongest gains are in interactive task completion, accompanied by consistent QA improvements over the three key baselines.

3.3 Task-Level Analysis

Interactive gains cover nearly all task categories. EvoCred improves over GRPO in 17 of the 18 ALFWorld backbone–category combinations. On Qwen2.5-3B, Look, Cool, and Pick2 improve by 25.5, 35.0, and 34.0 pp. All six categories improve on Qwen3-1.7B, including Clean from 36.4% to 98.9% and Pick2 from 31.6% to 85.6%. The large macro-level gain thus reflects improvements across multiple manipulation tasks.

QA gains combine broad coverage with large dataset-specific improvements. EvoCred improves over GRPO on six of seven QA datasets with Qwen2.5-3B, all seven with Qwen2.5-7B, and four with Qwen3-1.7B: 17 of 21 backbone–dataset comparisons overall. Bamboogle yields the largest gains on the Qwen2.5 models, at 41.8 and 35.4 points. Improvements also extend to NQ, PopQA, and HotpotQA on the 3B model (+6.9, +4.9, and +4.0), and MuSiQue on the 7B model (+4.4).

3.4 Training Dynamics

Figure 3 compares EvoCred with GRPO, SEED, and AgentOPSD over training. EvoCred establishes a reward advantage during the middle of training and maintains it through later updates, finishing at a higher reward. This sustained separation complements the final benchmark results: the improvement develops over the course of policy optimization.

3.5 Mechanistic Analysis

We examine how source exclusion, admission, and credit conservation contribute to learning, using the component controls in Table 2 and the evidence diagnostics in Figure 5.

Component controls test the credit construction. **Self-source** assigns each target its own skill while retaining admission and conservation. **Without admission** fixes $\rho_i = 1$, retaining source exclusion and conservative allocation. **Without centering** replaces $(u_{ik\ell} - \bar{u}_i)/2$ with raw $q_{ik\ell}/2$, bypassing standardization and $\tanh$. Multipliers remain positive and bounded, but credit mass is no longer conserved. **Shuffled allocation** permutes aligned evidence across turns for allocation, keeps the original ρ_i, and recomputes credit-weighted centering. This control tests whether the temporal placement of evidence matters beyond bounded redistribution. Appendix C.1 gives the exact controlled changes.

The full method outperforms every component control. In the component study, the full method achieves 89.6% ALFWorld success and 74.0% WebShop success, outperforming every controlled variant in both final success and success AUC. Self-source scoring reduces these success rates by 5.3 and 3.9 pp, respectively. Removing admission costs 3.4 and 2.4 pp, while raw, uncentered allocation costs 2.5 and 2.0 pp. These comparisons support the full credit construction.

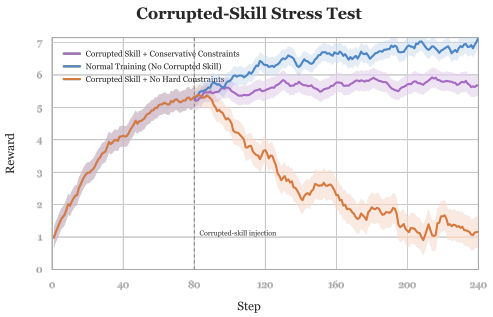


Figure 4: **Corrupted-skill robustness.** Following corruption at step 80, constrained training retains its pre-injection reward level, while unconstrained training deteriorates.

Temporal placement matters beyond bounded reweighting. Shuffling the allocation produces the largest degradation among the component controls: ALFWorld and WebShop success fall by 8.0 and 5.4 pp, and their success AUCs decrease by 8.0 and 6.4 points. Because this control retains admission and credit-mass conservation, the result identifies the temporal placement of evidence as a contributor to both final performance and learning progress.

Evidence diagnostics connect skill provenance to credit influence. Figure 5(a) compares self-source, same-task sibling, and misleading-control skills on identical target histories and actions at early, middle, and late checkpoints. The comparison tracks how skill source affects fixed-action likelihood gains as training progresses. Panel (b) reports the late-stage distribution of sibling-skill admission weights ρ_i and the fraction with $\rho_i > 0$, separately for positive and negative base advantages. Together, the panels distinguish the likelihood evidence produced by a skill from the influence admitted into credit allocation.

3.6 Robustness to Corrupted Skills

We test whether the credit constraints preserve learning progress when skill guidance becomes unreliable. Figure 4 compares uncorrupted training with two corrupted-skill conditions, with and without the constraints. Corruption begins at step 80, when the reward traces are at comparable levels.

After injection, unconstrained training shows a sustained reward decline. The constrained condition retains its pre-injection reward level and continues to improve modestly, while uncorrupted training

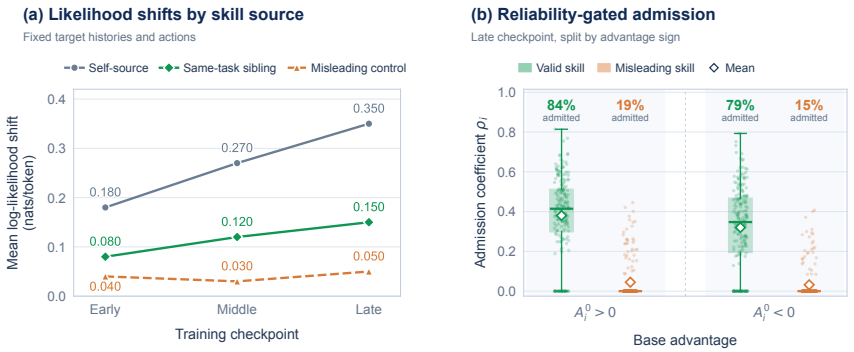


Figure 5: **Diagnostics of skill evidence and admission.** (a) Mean fixed-action log-likelihood shifts by skill source. (b) Late-stage admission coefficients, split by advantage sign; percentages indicate $\rho_i > 0$. Boxes show quartiles, whiskers use the 1.5-IQR rule, and diamonds denote means.

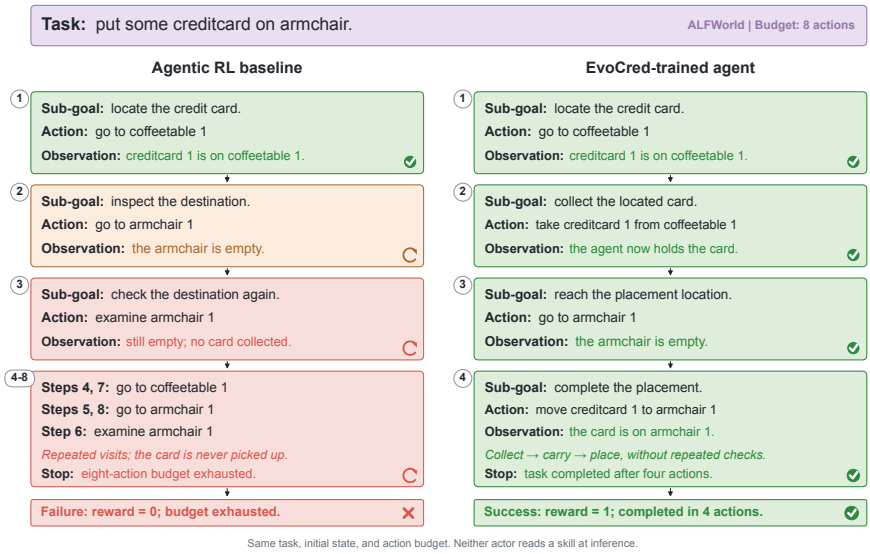


Figure 6: **Task execution in an ALFWorld case.** Under the same initial state and eight-action budget, the baseline exhausts its budget without collecting the card; the EvoCred-trained agent completes the task in four actions. Both agents act without inference-time skills.

reaches higher rewards. The intervention shows that the full credit mechanism limits the degradation caused by misleading skills and helps preserve progress already achieved by the policy.

3.7 Qualitative Behavior

Figure 6 illustrates how improved task completion appears in an ALFWorld trajectory. Both agents locate a credit card, but the baseline repeatedly checks the destination without collecting it. The EvoCred-trained agent executes the required collect–carry–place sequence and completes the task. The case highlights the transition from recognizing a subgoal to taking the actions needed to satisfy it.

4 Related Work

Privileged teachers and fine-grained credit. Skill-conditioned self-distillation provides dense training guidance that is absent at deployment. Skill-SD Wang et al. (2026a) transfers trajectory-derived skills through a synchronized teacher; OPID Yang et al. (2026c) combines episode- and step-level skill-conditioned likelihood shifts with outcome advantage; and SEED Wu et al. (2026) writes a hindsight skill from each on-policy trajectory before re-scoring that trajectory's fixed actions.

These methods demonstrate the value of evolving privileged teachers. For trajectory-authored skills, however, a same-source likelihood increase may reflect hindsight reconstruction, leaving its value for credit assignment ambiguous. EVOCRED makes provenance an explicit constraint: each target is scored with a sibling-authored skill under an assignment fixed before the conditioned scores are observed. This separates the trajectory used to construct a skill from the trajectory on which it supplies credit evidence Zhao et al. (2026); Hübotter et al. (2026); Ye et al. (2026).

AgentOPSD Wang et al. (2026b) is closest in how likelihood evidence enters a sequential update. It aggregates token-level teacher–student gaps into turn evidence, recursively updates a bounded support state, aligns state revisions with the outcome advantage, and converts them into bounded positive multipliers. EVOCRED complements this sequential construction with source exclusion and an explicit admission gate that suppresses weak or insufficiently aligned evidence. Its additional conservation constraint comes from centering under the absolute base-advantage measure: positive multipliers preserve tokenwise signs, and weighted centering further preserves each trajectory's L_1 credit mass Yang et al. (2026a); Lu et al. (2026); Zhang et al. (2026).

Experience-derived skills and memory. Language agents increasingly compress trajectories into reusable abstractions. SkillRL Xia et al. (2026) builds a skill bank that co-evolves with the policy, EMPO^2 Liu et al. (2026); Chen et al. (2026) uses self-generated memory to support exploration, and MEM1 Zhou et al. (2026) consolidates long interaction histories into a bounded state. Policy-internalization and workflow training further transfer prompt- or trajectory-level knowledge into model parameters. This literature studies how experience should be compressed, retrieved, or internalized Yang et al. (2026d); Zhao et al. (2024); Wang et al. (2024); Wu et al. (2025); Ye et al. (2026). Our question is complementary: when a trajectory-derived skill is used to assign credit to its own source trajectory, what evidence should authorize its influence?

Agentic reinforcement learning with verifiable outcomes. Agentic RL extends single-turn reasoning to long-horizon planning, tool use, and environment interaction. Representative systems improve rollout stability, executable environments, and verifier design: AgentFlow Li et al. (2026) and AgentGym-RL Xi et al. (2025) develop multi-turn training frameworks, while Gaia2 Russell et al. (2025) and MedAgentGym Xu et al. (2025) construct dynamic or domain-specific environments with fine-grained verification. Other methods target concrete reward-path failures, such as invalid tool turns or partial credit for executable programs. These works improve the environment, rollout process, or outcome signal Xue et al. (2026); Applis et al. (2026). EVOCRED complements these efforts by controlling how training-time linguistic hindsight may redistribute advantages already computed from the verifier.

Adaptive data reuse and constrained influence. Sample splitting and cross-fitting motivate separating an object's construction from its evaluation. EVOCRED applies this principle to skill provenance by excluding the target trajectory as its own skill source. Sibling rollouts share a task and frozen policy snapshot, so this source exclusion alone establishes neither statistical independence nor classical inferential guarantees. Our use of "conservative" concerns a different object from trust-region or clipped policy optimization Yu et al. (2022); Ivison et al. (2024). Those methods control policy updates; EVOCRED constrains the scalar credit coefficients supplied to the optimizer. Its conservation guarantee holds at the coefficient level and does not imply invariance of the aggregate policy gradient Engstrom et al. (2019).

5 CONCLUSION

EVOCRED uses self-generated hindsight to refine temporal credit allocation within budgets established by task-return advantages. Source-excluded evaluation tests skills on sibling trajectories, admission controls their influence, and positive, credit-centered multipliers preserve token-level signs and each trajectory's total absolute credit. Across three Qwen backbones and three agentic suites, EVOCRED ranks first in ten of twelve aggregate comparisons and outperforms GRPO, SEED, and AgentOPSD in all twelve, without additional environment rollouts or inference-time skills. These results support a broader design principle: self-generated supervision can improve learning through evidence-guided allocation of outcome-derived credit, with explicit constraints on its influence.

Reproducibility Statement

Appendix A documents the data partitions, training configurations, hyperparameters, and skill-generation prompt. Appendix B specifies fixed-action scoring, evidence accumulation, credit redistribution, and the training objective, together with proofs of the credit guarantees. Appendix C defines metric aggregation and component controls, while Appendix D explains figure-specific evaluation scopes and plotting conventions.

AI Use Statement

Generative AI tools assisted with language editing, LaTeX preparation, and code for figure design. The authors take full responsibility for the accuracy and integrity of the final text, equations, code, figures, and empirical claims.

References

Massih-Reza Amini, Vasilii Feofanov, Loic Pauletto, Lies Hadjadj, Emilie Devijver, and Yury Maximov. Self-training: A survey. *Neurocomputing*, 616:128904, 2025.

Leonhard Applis, Yuntong Zhang, Shanchao Liang, Nan Jiang, Lin Tan, and Abhik Roychoudhury. Unified software engineering agent as ai software engineer. In *Proceedings of the 2026 IEEE/ACM 48th International Conference on Software Engineering*, pp. 907–918, 2026.

Shuhang Chen, Yunqiu Xu, Junjie Xie, Aojun Lu, Tao Feng, Zeying Huang, Zhang Ning, Yi Sun, Yi Yang, and Hangjie Yuan. Cogflow: Bridging perception and reasoning through knowledge internalization for visual mathematical problem solving. In *International Conference on Learning Representations*, volume 2026, pp. 67538–67570, 2026.

Yunshu Du, Wojciech M. Czarnecki, Siddhant M. Jayakumar, Mehrdad Farajtabar, Razvan Pascanu, and Balaji Lakshminarayanan. Adapting auxiliary losses using gradient similarity, 2020. URL https://arxiv.org/abs/1812.02224.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toni Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. *Advances in neural information processing systems*, 28, 2015.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation matters in deep rl: A case study on ppo and trpo. In *International conference on learning representations*, 2019.

Giuseppe Gallitto, Robert Englert, Balint Kincses, Raviteja Kotikalapudi, Jialin Li, Kevin Hoffschlag, Ulrike Bingel, and Tamas Spisak. External validation of machine learning models—registered models and adaptive sample splitting. *GigaScience*, 14:giaf036, 2025.

Xu Huang, Weiwen Liu, Xiaolong Chen, Xingmei Wang, Hao Wang, Defu Lian, Yasheng Wang, Ruiming Tang, and Enhong Chen. Understanding the planning of llm agents: A survey. *arXiv preprint arXiv:2402.02716*, 2024.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A Smith, Yejin Choi, and Hannaneh Hajishirzi. Unpacking dpo and ppo: Disentangling best practices for learning from preference feedback. *Advances in neural information processing systems*, 37: 36602–36633, 2024.

Yubo Jiang, Yitong An, Xin Yang, Abudukelimu Wuerkaixi, Xuxin Cheng, Fengying Xie, Zhiguo Jiang, Cao Liu, Ke Zeng, and Haopeng Zhang. Breaking the illusion: When positive meets negative in multimodal decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4210–4220, June 2026a.

Yubo Jiang, Xin Yang, Abudukelimu Wuerkaixi, Zheming Yuan, Xuxin Cheng, Cao Liu, Ke Zeng, Fengying Xie, Zhiguo Jiang, and Haopeng Zhang. Global context or local detail? adaptive visual grounding for hallucination mitigation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 15133–15152, 2026b.

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.

Nikolaus Kriegeskorte, W Kyle Simmons, Patrick SF Bellgowan, and Chris I Baker. Circular analysis in systems neuroscience: the dangers of double dipping. *Nature neuroscience*, 12(5):535–540, 2009.

Xinyi Li, Sai Wang, Siqi Zeng, Yu Wu, and Yi Yang. A survey on llm-based multi-agent systems: workflow, infrastructure, and challenges. *Vicinagearth*, 1(1):9, 2024.

Zhuofeng Li, Haoxiang Zhang, Seungju Han, Sheng Liu, Jianwen Xie, Yu Zhang, Yejin Choi, James Y Zou, and Pan Lu. In-the-flow agentic system optimization for effective planning and tool use. In *International Conference on Learning Representations*, volume 2026, pp. 50524–50570, 2026.

Zeyuan Liu, Jeonghye Kim, Xufang Luo, Dongsheng Li, and Yuqing Yang. Exploratory memory-augmented llm agent via hybrid on-and off-policy optimization. In *International Conference on Learning Representations*, volume 2026, pp. 111799–111820, 2026.

Zhengxi Lu, Zhiyuan Yao, Zhuowen Han, Zi-Han Wang, Jinyang Wu, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, et al. Self-distilled agentic reinforcement learning. *arXiv preprint arXiv:2605.15155*, 2026.

Tergel Munkhbat, Namgyu Ho, Seo Hyun Kim, Yongjin Yang, Yujin Kim, and Se-Young Yun. Self-training elicits concise reasoning in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 25127–25152, 2025.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. Gaia-2: A controllable multi-view generative world model for autonomous driving. *arXiv preprint arXiv:2503.20523*, 2025.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

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

Yaorui Shi, Yuxin Chen, Zhengxi Lu, Yuchun Miao, Shugui Liu, Qi Gu, Xunliang Cai, Xiang Wang, and An Zhang. Skill1: Unified evolution of skill-augmented agents via reinforcement learning. *arXiv preprint arXiv:2605.06130*, 2026.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. *arXiv preprint arXiv:2010.03768*, 2020.

Hieu Tran, Zonghai Yao, and Hong Yu. Exploiting tree structure for credit assignment in rl training of llms. *arXiv preprint arXiv:2509.18314*, 2025.

Songjun Tu, Jiahao Lin, Qichao Zhang, Xiangyu Tian, Linjing Li, Xiangyuan Lan, and Dongbin Zhao. Learning when to think: Shaping adaptive reasoning in r1-style models via multi-stage rl. *Advances in Neural Information Processing Systems*, 38:15181–15207, 2026.

Hao Wang, Guozhi Wang, Han Xiao, Yufeng Zhou, Yue Pan, Jichao Wang, Ke Xu, Yafei Wen, Xiaohu Ruan, Xiaoxin Chen, et al. Skill-sd: Skill-conditioned self-distillation for multi-turn llm agents. *arXiv preprint arXiv:2604.10674*, 2026a.

Zi-Han Wang, Zhengxi Lu, Zhiyuan Yao, Jinyang Wu, Jie Wu, Zhengzhou Cai, Yueqing Sun, Ziang Ye, Linji Hao, Qi Gu, et al. Agentopsd: Recursive self-distillation for agentic reinforcement learning. *arXiv preprint arXiv:2608.05987*, 2026b.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. *arXiv preprint arXiv:2409.07429*, 2024.

Jinyang Wu, Shuo Yang, Zhengxi Lu, Fan Zhang, Yuhao Shen, Lang Feng, Haoran Luo, Zheng Lian, Shuai Zhang, Zhengqi Wen, et al. Seed: Self-evolving on-policy distillation for agentic reinforcement learning. *arXiv preprint arXiv:2607.14777*, 2026.

Rong Wu, Xiaoman Wang, Jianbiao Mei, Pinlong Cai, Daocheng Fu, Cheng Yang, Licheng Wen, Xuemeng Yang, Yufan Shen, Yuxin Wang, et al. Evolver: Self-evolving llm agents through an experience-driven lifecycle. *arXiv preprint arXiv:2510.16079*, 2025.

Zhiheng Xi, Jixuan Huang, Chenyang Liao, Baodai Huang, Honglin Guo, Jiaqi Liu, Rui Zheng, Junjie Ye, Jiazheng Zhang, Wenxiang Chen, et al. Agentgym-rl: Training llm agents for long-horizon decision making through multi-turn reinforcement learning. *arXiv preprint arXiv:2509.08755*, 2025.

Peng Xia, Jianwen Chen, Hanyang Wang, Jiaqi Liu, Kaide Zeng, Yu Wang, Siwei Han, Yiyang Zhou, Xujiang Zhao, Haifeng Chen, et al. Skillrl: Evolving agents via recursive skill-augmented reinforcement learning. *arXiv preprint arXiv:2602.08234*, 2026.

Ran Xu, Yuchen Zhuang, Yishan Zhong, Yue Yu, Xiangru Tang, Hang Wu, May Dongmei Wang, Peifeng Ruan, Donghan Yang, Tao Wang, et al. Medagentgym: Training llm agents for code-based medical reasoning at scale. In *The Second Workshop on GenAI for Health: Potential, Trust, and Policy Compliance*, 2025.

Zhenghai Xue, Longtao Zheng, Qian Liu, Yingru Li, Xiaosen Zheng, Zejun Ma, and Bo An. Simpletir: End-to-end reinforcement learning for multi-turn tool-integrated reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 8424–8449, 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.

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026a.

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr, 2026b. URL `https://arxiv.org/abs/2604.03128`.

Shuo Yang, Jinyang Wu, Zhengxi Lu, Yuhao Shen, Fan Zhang, Lang Feng, Shuai Zhang, Haoran Luo, Zheng Lian, Zhengqi Wen, et al. Opid: On-policy skill distillation for agentic reinforcement learning. *arXiv preprint arXiv:2606.26790*, 2026c.

Zonghan Yang, Shengjie Wang, Kelin Fu, Wenyang He, Weimin Xiong, Yibo Liu, Yibo Miao, Bofei Gao, Yejie Wang, Yanhao Li, et al. Kimi-dev: Agentless training as skill prior for swe-agents. In *International Conference on Learning Representations*, volume 2026, pp. 148369–148436, 2026d.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in neural information processing systems*, 35:24611–24624, 2022.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

Guibin Zhang, Hejia Geng, Xiaohang Yu, Zhenfei Yin, Zaibin Zhang, Zelin Tan, Heng Zhou, Zhongzhi Li, Xiangyuan Xue, Yijiang Li, et al. The landscape of agentic reinforcement learning for llms: A survey. *arXiv preprint arXiv:2509.02547*, 2025.

Yanfei Zhang, Xu Lin, and Chenglin Wu. Stepopsd: Step-aware online preference distillation for agent reinforcement learning. *arXiv preprint arXiv:2605.27140*, 2026.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19632–19642, 2024.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Wenxuan Zhou, Shujian Zhang, Lingxiao Zhao, and Tao Meng. T-reg: Preference optimization with token-level reward regularization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27876–27889, 2025.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Bryan Kian Hsiang Low, and Paul Liang. Mem1: Learning to synergize memory and reasoning for efficient long-horizon agents. In *International Conference on Learning Representations*, volume 2026, pp. 58413–58438, 2026.

Table 3: Data partitions and periodic evaluation schedule. Counts refer to runnable games, goal identities, or questions, rather than rollout multiplicity.

Benchmark	Training data	Evaluation data
ALFWorld	3,553 runnable games from the training split, covering six task types.	140 runnable games from `valid_seen`; 128 sampled episodes every 5 updates.
Search QA	169,615 questions: NQ (79,168) and HotpotQA (90,447).	51,713 questions: NQ (3,610), TriviaQA (11,313), PopQA (14,267), HotpotQA (7,405), 2WikiMultiHopQA (12,576), MuSiQue (2,417), and Bamboogle (125). Intermediate NQ evaluation every 10 updates.
WebShop	1,000-product catalog with 6,910 generated goals; fixed global training indices $[500, 6910)$ (6,410 goals).	Fixed global evaluation indices $[0, 500)$ (500 goals); 128 sampled episodes every 5 updates.

WebShop uses one fixed, globally shared partition of goal identities. Worker-level sampling preserves split membership, so training and evaluation goals are disjoint across workers. ALFWorld and WebShop evaluation episodes are resampled within their respective evaluation partitions across checkpoints.

A Experimental Settings

Data and evaluation. Table 3 summarizes the data partitions and evaluation schedule. ALFWorld and WebShop tasks are sampled by their environment managers; their Parquet files contain scheduling placeholders rather than a fixed set of 16 training tasks. Periodic evaluation uses 128 episodes per checkpoint for these two environments, with one rollout per sampled environment instance. These evaluation batches are separate from the training task groups. Search QA uses a fixed 128-question NQ subset for intermediate evaluation and the complete 51,713-question evaluation set at step 160.

WebShop goal-level separation. The WebShop results in Tables 1 and 2 use a global partition over stable goal identities. Let g_j denote goal j in the shared goal list. We use

$$\mathcal{G}_{\text{train}} = \{g_j : 500 \le j < 6910\}, \qquad \mathcal{G}_{\text{eval}} = \{g_j : 0 \le j < 500\}, \qquad \mathcal{G}_{\text{train}} \cap \mathcal{G}_{\text{eval}} = \varnothing.$$

All workers share this same mapping from goal identity to split. Worker-specific sampling or shuffling preserves that mapping and cannot move goals between the training and evaluation partitions. Training rollouts and the hindsight skills constructed from them use only $\mathcal{G}_{\text{train}}$; evaluation rollouts use only $\mathcal{G}_{\text{eval}}$. Resampling at successive checkpoints changes which held-out goals are evaluated, not their split membership. This is a goal-level separation over the shared 1,000-product catalog.

Training configuration. The main experiments use Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, and Qwen3-1.7B-Instruct, matching Table 1. Training uses 160 updates, eight rollouts per task ($K = 8$), and environment seed 0. Benchmark-specific settings appear in Table 4. We perform full-parameter optimization with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$), a learning rate of 10^{-6}, weight decay 0.01, gradient clipping at 1.0, and one PPO epoch per collected batch. The entropy and reference-policy KL coefficients are 10^{-3} and 10^{-2}, respectively. KL regularization is applied in the loss. Training uses FSDP with BF16 mixed precision, gradient checkpointing, and vLLM rollouts. Sampling temperatures are 1.0 for training and 0.4 for evaluation, with top-$p = 1.0$ and no top-k truncation.

The learning rate remains constant after warmup. Search QA retrieves the top three passages per query using E5-base-v2 and a FAISS flat index over the Wikipedia-2018 corpus (21,015,324 passages). Evaluation reports environment success for ALFWorld, normalized exact match for Search-QA, and both task score and success for WebShop. Metric aggregation is defined in Appendix C.1. Table 4 retains γ_{RL} as a trainer configuration entry; γ_b is the decay used in the evidence recursion below. WebShop episode returns are summed without length normalization as specified in Eq. equation 4.

Skill generation and EvoCred hyperparameters. A frozen snapshot of the current rollout policy generates one episode-level skill from each completed training trajectory using Figure 7. The writer receives the task, trajectory, and success label. Its four skill fields are concatenated as labeled clauses; the episode summary is excluded from rescoring. Each target is scored with a skill from another

Table 4: Benchmark-specific training settings. Qwen2.5 model names denote their Instruct checkpoints.

Setting	ALFWorld	Search QA	WebShop
Training tasks per batch	16	128	16
Training trajectories per batch	128	1,024	128
Maximum environment turns	30	4	15
History length (turns)	5	4	2
Maximum prompt tokens	2,048	4,096	6,000
Maximum response tokens	512	512	4,096
PPO minibatch size (configured)	256	128	128
PPO microbatch per GPU	8	16 (3B); 8 (7B)	8
PPO clip $(\epsilon_{\text{low}}, \epsilon_{\text{high}})$	$(0.2, 0.2)$	$(0.2, 0.24)$	$(0.2, 0.2)$
Learning-rate warmup fraction	0	0.1	0
Configured RL discount γ_{RL}	0.95	1.0	0.95
Invalid-action penalty coefficient	0.1	0.01	0.1
EvoCred group-return normalization	Mean/std.	Mean/std.	Mean/std.
H800 80GB allocation	8 training	4 training + 4 retrieval	8 training

Table 5: EvoCred and skill-generation hyperparameters shared across benchmarks.

Hyperparameter	Value
Initial support / evidence decay	$B_0 = 0.5$; $\gamma_b = 0.95$
Turn-evidence clipping	Sum token log-likelihood shifts, then clip to $[-8, 8]$
Compatibility margin / evidence-strength scale	$\eta = 0.05$; $\kappa = 0.1$
Reliability floor / variance stabilizer	0; $\epsilon_z = 10^{-6}$
Allocation transform / amplitude	Within-trajectory standardization, then $\tanh$; $\lambda = 0.5$
Donor assignment	Self-excluded, outcome-aware; donor reuse allowed; fixed before conditional scoring
Reliability / credit budget	One-sided gate; exact-budget centering
Skill decoding	Greedy ($T = 0$), one completion per trajectory
Skill input / output token limits	10,240 / 2,048
Auxiliary distillation / skill-generation loss	Both disabled

trajectory of the same task. Donors may be reused, but a trajectory cannot receive its own skill. The writer and scorer remain frozen for the current collection and scoring cycle and are refreshed with the rollout policy across updates. Skills are used only during training. Table 5 lists the shared settings.

B Method Details and Credit Guarantees

This appendix expands the method in Section 2 and Algorithm 1. The computation uses one sequence: token log-likelihood shifts are summed within each turn and clipped before support accumulation; raw aligned support revisions determine admission; standardized, $\tanh$-transformed revisions determine allocation. The distinction between raw evidence q and allocation scores u is retained throughout the objective, proofs, and component controls. Donor assignment excludes the target itself and permits reuse across targets.

B.1 Training Returns and Base Credit

For each task x, the frozen rollout actor π_{θ^-} samples K trajectories $\tau_i = \{(h_{ik}, a_{ik}, o_{i,k+1})\}_{k=1}^{T_i}$. Here h_{ik} is the original interaction history, a_{ik} is the sampled response, and $y_{ik\ell}$ is its ℓth token. Training returns R_i incorporate the task outcome and the benchmark's prescribed invalid-action penalties before any hindsight evidence is introduced.

Prompt for On-Policy Skill Generation

User Instruction:
Analyze the following agent episode and return ONLY one valid JSON object.

Include `episode_summary` plus the four skill fields. Each skill field must be a short policy-facing clause grounded in the trajectory:
- `precondition`: when the rule applies, using observed state rather than hindsight.
- `action_rule`: what to do and in what order.
- `expected_observation`: what concrete feedback confirms progress.
- `avoid`: one concrete failure mode or warning sign.

Do not add an `episode_skill` field. Do not mention unsupported object locations, generic advice such as "search thoroughly", or retrospective phrases such as "the agent failed".

Output Format (JSON):
Return exactly this shape:

```
{
  "episode_summary":  "one concise sentence",
  "precondition":  "string",
  "action_rule":  "string",
  "expected_observation":  "string",
  "avoid":  "string"
}
```

Episode Context:
- Task description: {`task_description`}
- `episode_success`: {`success / failure`}
- Interaction trajectory: {`trajectory`}

Figure 7: The shared user-message template for skill generation, with formatting adapted for readability. The trajectory contains each step's observation, model response, and action-validity flag. The backbone's chat template is applied without an additional task-specific system instruction.

WebShop reward and reported score. Let $S_i \in \{0, 1\}$ indicate episode success and let N_i^{invalid} count turns with invalid actions. The environment assigns its success reward when the episode terminates successfully:

$$R_i^{\text{env}} = 10S_i, \qquad R_i = 10S_i - 0.1N_i^{\text{invalid}}. \tag{4}$$

The episode return is summed without dividing by episode length. The continuous task score, stored in `info['task_score']`, is used to report WebShop Score. Thus the environment outcome reward has two values, while the training return also reflects action validity.

Group-relative advantages. All three benchmarks subtract the task-group mean and divide by the group return standard deviation:

$$\bar{R}_x = \frac{1}{K}\sum_{j=1}^{K} R_j, \qquad A_i^0 = \frac{R_i - \bar{R}_x}{\widehat{\sigma}_x + \epsilon_A},$$
$$A_{ik\ell}^0 = m_{ik\ell} A_i^0. \tag{5}$$

Here $\widehat{\sigma}_x$ is the group return standard deviation, $\epsilon_A > 0$ is a numerical stabilizer, and $m_{ik\ell}$ is the trainer's binary response-token optimization mask. The sign of A_i^0 expresses performance relative to the task group, including penalties; it is not determined by the success label alone.

The absolute-credit measure. For a trajectory with positive base-credit mass, define

$$M_i^A = \sum_{k,\ell} |A_{ik\ell}^0|, \qquad \mu_{ik\ell} = \frac{|A_{ik\ell}^0|}{M_i^A} \quad (M_i^A > 0). \tag{6}$$

This measure determines the relative weight of each coefficient in the credit budget. Let $n_{ik} = \sum_\ell m_{ik\ell}$ and $n_i = \sum_k n_{ik}$. For the episode-level advantage broadcast used here,

$$M_i^A = |A_i^0| n_i, \qquad \mu_{ik\ell} = \frac{m_{ik\ell}}{n_i}, \qquad \sum_\ell \mu_{ik\ell} = \frac{n_{ik}}{n_i}.$$

Centering is therefore uniform over optimized tokens and weights each turn by its optimized-token count. This makes the conservation rule consistent with the token-level training objective even when response lengths differ. The general definition in Eq. equation 6 also accommodates nonuniform token advantages determined before hindsight. For $M_i^A = 0$, evidence evaluation is skipped and we set $\rho_i = 0$, $g_{ik\ell} = 1$, and $\widetilde{A}_{ik\ell} = A^0_{ik\ell} = 0$ without forming μ_i.

B.2 Source-Excluded Skill Assignment

The writer generates $s_i = G_{\theta^-}(x, \tau_i, S_i)$ from the success label and trajectory using Figure 7. Only the four policy-facing skill fields condition rescoring. For each target i, a donor map selects one sibling:

$$p : \{1, \dots, K\} \longrightarrow \{1, \dots, K\}, \qquad p(i) \neq i. \tag{7}$$

The map need not be injective: multiple targets may receive the same skill. Assignment is outcome-aware and is fixed before observing skill-conditioned target scores. This is the sense in which assignment is *score-blind*; it excludes score-based selection among candidate skills without requiring reward-blind assignment. Source exclusion prevents a skill from being evaluated on its construction trajectory. It concerns skill provenance, rather than statistical independence of same-task rollouts. The credit guarantees below hold separately for each target and do not require distinct donors across targets.

B.3 Fixed-Action Scores and Sequential Evidence

Both scoring branches use π_{θ^-} with the target's recorded actions and response-token prefixes. Only the skill insertion H changes the conditioning context:

$$\delta_{ik\ell} = \mathrm{sg}\left[\log \frac{\pi_{\theta^-}(y_{ik\ell} \mid H(h_{ik}, s_{p(i)}), y_{ik,<\ell})}{\pi_{\theta^-}(y_{ik\ell} \mid h_{ik}, y_{ik,<\ell})}\right]. \tag{8}$$

Here sg denotes stop-gradient. No target action is regenerated. Ordinary log-probabilities are reused from the rollout under the same frozen actor and scoring convention.

Turn evidence. We sum log-likelihood shifts over the optimized response tokens and clip the total:

$$e_{ik} = \mathrm{clip}\left(\sum_\ell m_{ik\ell}\delta_{ik\ell}, -8, 8\right). \tag{9}$$

The sum represents the change in log-likelihood of the scored response; clipping bounds the evidence entering the sequential update. No turn-length average or $\tanh$ transform is applied at this stage.

Support revisions. The support state starts from the fixed value $B_{i0} = 0.5$ and accumulates evidence in log-odds space:

$$\begin{aligned} B_{i0} &= \tfrac{1}{2}, & c_{i0} &= 0, \\ c_{ik} &= \gamma_b c_{i,k-1} + e_{ik}, & B_{ik} &= \sigma(c_{ik}), \\ \Delta B_{ik} &= B_{ik} - B_{i,k-1}, & \Delta B_{ik\ell} &= m_{ik\ell}\Delta B_{ik}. \end{aligned} \tag{10}$$

Here σ is the logistic function and $\gamma_b = 0.95$ discounts previous evidence. The fixed initialization corresponds to zero initial log-odds and is independent of the group rewards. Since $B_{ik} \in (0, 1)$, each revision lies in $[-1, 1]$. A revision measures the change in accumulated support, including evidence decay, rather than the isolated likelihood shift of the current turn. The state serves as bounded hindsight evidence, not a calibrated success probability.

B.4 Evidence Strength and Compatibility

For $M_i^A > 0$, we align each revision with its base-credit sign and measure its magnitude and directional agreement:

$$q_{ik\ell} = \mathrm{sgn}(A^0_{ik\ell})\Delta B_{ik\ell}, \qquad m_i = \sum_{k,\ell} \mu_{ik\ell}|\Delta B_{ik\ell}|,$$

$$C_i = \begin{cases} \dfrac{\sum_{k,\ell} \mu_{ik\ell} q_{ik\ell}}{m_i}, & m_i > 0, \\ 0, & m_i = 0. \end{cases} \tag{11}$$

On the credit support, $|q_{ik\ell}| = |\Delta B_{ik\ell}|$. Therefore,

$$\left|\mathbb{E}_{\mu_i}[q_i]\right| \le \mathbb{E}_{\mu_i}[|q_i|] = m_i \le 1,$$

so $C_i \in [-1, 1]$. The admission weight in Eq. equation 1 uses $\eta = 0.05$ and $\kappa = 0.1$: compatibility determines whether the gate opens, while evidence strength continuously attenuates its influence. When $m_i = 0$ or $C_i \le \eta$, $\rho_i = 0$. Large but conflicting revisions and small but aligned revisions are thus handled by different factors of the same gate.

B.5 Standardized Evidence for Allocation

Admission uses the raw revisions in Eq. equation 11. Allocation first standardizes the aligned turn values within each trajectory. Because base advantages are broadcast across response tokens, their aligned turn value is $q_{ik} = \mathrm{sgn}(A^0_i)\Delta B_{ik}$. Define

$$\widehat{q}_i = \frac{1}{T_i}\sum_{k=1}^{T_i} q_{ik}, \qquad v_i = \frac{1}{T_i}\sum_{k=1}^{T_i}(q_{ik} - \widehat{q}_i)^2,$$

$$z_{ik} = \frac{q_{ik} - \widehat{q}_i}{\sqrt{v_i + 10^{-6}}}, \qquad u_{ik} = \tanh(z_{ik}), \qquad u_{ik\ell} = m_{ik\ell}u_{ik}. \tag{12}$$

The arithmetic mean and variance operate over trajectory turns. The $\tanh$ transform is applied here, after standardization, without an additional temperature. It produces bounded allocation evidence $u_{ik\ell} \in [-1, 1]$.

We then center under the absolute-credit measure:

$$\bar{u}_i = \mathbb{E}_{\mu_i}[u_i], \qquad d_{ik\ell} = \frac{u_{ik\ell} - \bar{u}_i}{2}, \qquad g_{ik\ell} = 1 + \lambda\rho_i d_{ik\ell}.$$

These two centering operations have distinct roles. Turn-level standardization sets the scale before the nonlinear transform; credit-weighted centering after $\tanh$ enforces the budget constraint. A zero arithmetic mean before $\tanh$ does not imply a zero credit-weighted mean afterward. The admission weight continues to use the unstandardized q and ΔB, retaining their original strength and compatibility information.

B.6 Skill-Free Policy Optimization

The current-to-frozen policy ratio uses the original, skill-free history:

$$r_{ik\ell}(\theta) = \frac{\pi_\theta(y_{ik\ell} \mid h_{ik}, y_{ik,<\ell})}{\pi_{\theta^-}(y_{ik\ell} \mid h_{ik}, y_{ik,<\ell})}.$$

The full objective retains the base trainer's clipping, reference-policy KL term, and entropy regularization:

$$\mathcal{L}_{\mathrm{ours}}(\theta) = -\frac{1}{N}\sum_{i,k,\ell} m_{ik\ell}\Phi\Big(r_{ik\ell}, \mathrm{sg}[\widetilde{A}_{ik\ell}]\Big) + \beta_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}} - \beta_{\mathrm{ent}}\mathcal{H},$$

$$\Phi(r, A) = \min(rA, \mathrm{clip}(r, 1 - \epsilon_{\mathrm{low}}, 1 + \epsilon_{\mathrm{high}})A)\,. \tag{13}$$

Here $N = \sum_{i,k,\ell} m_{ik\ell}$ counts optimized tokens over the update batch, with i ranging over trajectories from all task groups. We use $\beta_{\mathrm{KL}} = 10^{-2}$ and $\beta_{\mathrm{ent}} = 10^{-3}$; clipping parameters are listed in Table 4. All skill-side scores, admission weights, and transported advantages are detached. Return aggregation and loss reduction act at different stages: WebShop returns are not divided by episode length, whereas the policy objective is averaged over optimized tokens. Setting $\lambda = 0$ or closing all gates recovers the base GRPO objective.

Training and deployment cost. Each task group requires K skill generations and at most K conditional target–skill evaluations, reusing ordinary rollout log-probabilities. Donor reuse does not introduce additional target comparisons. The number of scoring pairs is $O(K)$, compared with $K(K-1)$ for all sibling pairs; scoring cost also depends on the lengths of trajectories and skills. No extra environment rollouts are needed. Deployment uses the trained actor alone, without skill generation, conditional rescoring, or admission.

B.7 Derivation of Conservative Redistribution

Fix a trajectory with $M_i^A > 0$. Then $\sum_{k,\ell} \mu_{ik\ell} = 1$ and $u_{ik\ell} \in [-1, 1]$. Writing $\bar{u}_i = \mathbb{E}_{\mu_i}[u_i]$ gives $|d_{ik\ell}| = |u_{ik\ell} - \bar{u}_i|/2 \le 1$ and

$$\mathbb{E}_{\mu_i}[d_i] = \tfrac{1}{2}\big(\mathbb{E}_{\mu_i}[u_i] - \bar{u}_i\big) = 0.$$

The multiplier is exactly $g_{ik\ell} = 1 + \lambda\rho_i(u_{ik\ell} - \bar{u}_i)/2$, with $\lambda = 0.5$ in our experiments.

Positivity and sign preservation. For $\lambda \in [0, 1)$ and $\rho_i \in [0, 1]$,

$$0 < 1 - \lambda\rho_i \le g_{ik\ell} \le 1 + \lambda\rho_i.$$

Thus multiplying by $g_{ik\ell}$ preserves both the sign of every nonzero coefficient and the zero-credit support.

Exact absolute-credit conservation. Positivity and weighted centering yield

$$\begin{aligned}\sum_{k,\ell} |\widetilde{A}_{ik\ell}| &= \sum_{k,\ell} |A^0_{ik\ell}|(1 + \lambda\rho_i d_{ik\ell}) \\ &= M_i^A \left(1 + \lambda\rho_i \sum_{k,\ell} \mu_{ik\ell} d_{ik\ell}\right) = M_i^A.\end{aligned}$$

Equivalently, $\widetilde{\mu}_{ik\ell} = \mu_{ik\ell} g_{ik\ell}$ is a probability measure on the same support as μ_i. Weighting the center by absolute base credit is essential: a uniform mean over turns would generally leave a nonzero token-weighted residual when response lengths differ. The construction conserves the budget directly, without a subsequent clipping or normalization of advantages. PPO clipping in Eq. equation 13 acts on policy ratios within the loss.

Bounds on coefficient changes. Each coefficient satisfies

$$|\widetilde{A}_{ik\ell} - A^0_{ik\ell}| = \lambda\rho_i |A^0_{ik\ell}||d_{ik\ell}| \le \lambda\rho_i |A^0_{ik\ell}|.$$

The total relative change has the tighter bound

$$\begin{aligned}D_i := \frac{\sum_{k,\ell} |\widetilde{A}_{ik\ell} - A^0_{ik\ell}|}{M_i^A} &= \mathbb{E}_{\mu_i}[|g_i - 1|] \\ &= \frac{\lambda\rho_i}{2}\mathbb{E}_{\mu_i}[|u_i - \bar{u}_i|] \le \frac{\lambda\rho_i}{2}.\end{aligned} \tag{14}$$

The last inequality follows from Cauchy–Schwarz and $\mathrm{Var}_{\mu_i}(u_i) \le 1$. In particular,

$$\mathrm{TV}(\widetilde{\mu}_i, \mu_i) = \tfrac{1}{2} D_i \le \tfrac{1}{4}\lambda\rho_i.$$

Thus admission controls the maximum change as well as whether evidence can influence the update. With $\lambda = 0.5$, the relative coefficient change is at most $0.25\rho_i$.

Fallback and interpretation. If $\rho_i = 0$ or $\lambda = 0$, base credit is unchanged. Constant aligned turn evidence yields $z_i = u_i = 0$ and also leaves base credit unchanged. The explicit zero-mass branch satisfies these properties without constructing a measure. For negative base advantages, negative support revisions produce positive aligned evidence. Standardization and $\tanh$ preserve the ordering of aligned turn values. When $\lambda\rho_i > 0$, positions with $u_{ik\ell} > \bar{u}_i$ receive stronger negative coefficients, balanced by weaker penalties elsewhere.

These guarantees constrain the advantage coefficients supplied to the policy objective. They do not fix its gradient, since score gradients vary across tokens. At $\theta = \theta^-$, the change in the unclipped policy-loss gradient is

$$-\frac{1}{N}\sum_{i,k,\ell} A^0_{ik\ell}(g_{ik\ell} - 1)\ \nabla_\theta \log \pi_\theta(y_{ik\ell} \mid h_{ik}, y_{ik,<\ell})|_{\theta=\theta^-}\ .$$

Scalar credit conservation does not force this vector sum to vanish; it bounds how hindsight reallocates the coefficients driving learning.

C Evaluation and Diagnostic Details

C.1 Metric Aggregation and Component Controls

Benchmark-level aggregation. We report task-category and dataset scores separately from their suite-level aggregates. Let S_c denote the reported success percentage for ALFWorld category c, and let Q_d denote the reported answer accuracy for QA dataset d. The aggregate scores are

$$\mathrm{Avg.}_{\mathrm{ALF}} = \frac{1}{6}\sum_{c=1}^{6} S_c, \qquad \mathrm{Avg.}_{\mathrm{QA}} = \frac{1}{7}\sum_{d=1}^{7} Q_d.$$

The ALFWorld categories are Look, Pick, Clean, Heat, Cool, and Pick2. The QA datasets are Natural Questions (NQ), TriviaQA (Triv), PopQA (Pop), HotpotQA (Hotp), 2WikiMultiHopQA (2Wk), MuSiQue (MuS), and Bamboogle (Bam). Each category or dataset receives equal weight, regardless of its size. These macro averages operate across benchmark components; they are distinct from aggregation across evaluation repetitions or training seeds.

Scores for one evaluation pass. For a fixed checkpoint and evaluation pass r, let $N_{c,r}$ be the number of evaluated ALFWorld episodes in category c and $n^{\mathrm{succ}}_{c,r}$ the number of successful episodes. For QA dataset d, metrics use deduplicated evaluation trajectories: each evaluated response contributes one binary normalized exact-match outcome, irrespective of its number of interaction turns. Let $N_{d,r}$ count these responses and $n^{\mathrm{correct}}_{d,r}$ count correct answers. The corresponding single-pass scores are

$$S_{c,r} = 100\frac{n^{\mathrm{succ}}_{c,r}}{N_{c,r}}, \qquad Q_{d,r} = 100\frac{n^{\mathrm{correct}}_{d,r}}{N_{d,r}}.$$

The index r identifies an evaluation pass at a fixed checkpoint. Dataset cardinalities in Table 3 describe the evaluation sets; a single pass evaluating every question exactly once has $N_{d,r}$ equal to that dataset's cardinality. Repeated evaluation of the same checkpoint and evaluation of separately trained policies are distinct sources of variation.

WebShop metrics. For evaluation pass r containing $N_{\mathrm{WS},r}$ episodes, let $v_{rj} \in [0, 1]$ be episode j's continuous task score and $S_{rj} \in \{0, 1\}$ its success indicator. We define

$$\mathrm{Score}_r = \frac{100}{N_{\mathrm{WS},r}}\sum_j v_{rj}, \qquad \mathrm{Success}_r = \frac{100}{N_{\mathrm{WS},r}}\sum_j S_{rj}.$$

The continuous score is read from `info['task_score']`; success requires termination with the original environment score equal to 1.0. These metrics describe graded task fulfillment and complete success, respectively. They are separate from the penalized training return in Eq. equation 4. The periodic 128-episode evaluation batch in Table 3 specifies an evaluation schedule and does not by itself specify how final table entries are aggregated.

Numerical comparisons. The main and ablation tables report point estimates. Their rankings and differences summarize observed performance, without a claim of statistical significance. Success-rate differences are reported in percentage points and WebShop Score differences in points.

Learning-curve area. Success AUC summarizes the success curve over the recorded training coordinate u. For success expressed as a percentage $s(u)$, the interval-normalized area is

$$\mathrm{AUC}_{\mathrm{norm}} = \frac{1}{u_1 - u_0} \int_{u_0}^{u_1} s(u)\, du.$$

This normalization retains the success-percentage scale and measures performance throughout the interval. Comparisons within the component study use a common coordinate and interval. The separate ALFWorld learning–compute study below has its own evaluation records.

Component controls. The controls in Table 2 use the full method's base advantages, masks, and hyperparameters, changing the specified operation. Every variant retains zero credit when $M_i^A = 0$. *Self-source* substitutes s_i for $s_{p(i)}$ and recomputes the fixed-action evidence, admission, and centered allocation. *Without admission* sets $\rho_i = 1$ for nonzero-credit trajectories while retaining sibling evidence and centered allocation.

Without centering uses $d_{ik\ell}^{\mathrm{nc}} = u_{ik\ell}/2$. Its multipliers remain positive and bounded:

$$1 - \lambda\rho_i/2 \le g_{ik\ell}^{\mathrm{nc}} \le 1 + \lambda\rho_i/2.$$

However, the absolute-credit budget becomes

$$\frac{\sum_{k,\ell} |\widetilde{A}_{ik\ell}^{\mathrm{nc}}|}{M_i^A} = 1 + \frac{\lambda\rho_i}{2}\bar{u}_i. \tag{15}$$

The budget can increase or decrease according to $\bar{u}_i$. Admission is based on raw q, so it does not determine the sign of the mean after standardization and $\tanh$. This control isolates credit-weighted centering while retaining source exclusion, admission, and the allocation transform.

Shuffled allocation computes the original evidence and admission weight, then permutes the transformed allocation values u_{ik} among turns with nonzero credit support. The reassigned value is broadcast to each receiving turn's credited tokens, producing $u_{ik\ell}^{\mathrm{sh}}$. Allocation uses the original μ_i and ρ_i and a recomputed center:

$$d_{ik\ell}^{\mathrm{sh}} = \big(u_{ik\ell}^{\mathrm{sh}} - \mathbb{E}_{\mu_i}[u_i^{\mathrm{sh}}]\big)/2, \qquad g_{ik\ell}^{\mathrm{sh}} = 1 + \lambda\rho_i d_{ik\ell}^{\mathrm{sh}}.$$

Admission is not recomputed after shuffling. The control preserves the multiset of transformed turn values and exact absolute-credit mass while disrupting their temporal placement. When turn lengths differ, it need not preserve the token-weighted distribution of evidence.

C.2 Fixed-Target Evidence and Admission Diagnostics

Figure 5(a) compares self-source, same-task sibling, and misleading-control skills at early, middle, and late checkpoints. Within each checkpoint, the frozen actor, target histories, actions, response-token prefixes, and scoring masks are held fixed. The measured log-likelihood shifts therefore compare skill-conditioned scores of the same recorded actions. The plotted mean summarizes these scores; the sequential update uses the turn-wise sum and clipping in Eq. equation 9. The sequential revisions and admission weights are computed subsequently, as defined in Eqs. equation 10 and equation 11.

Panel (b) separates nonzero-credit trajectories into $\mathcal{I}_+ = \{i : M_i^A > 0,\ A_i^0 > 0\}$ and $\mathcal{I}_- = \{i : M_i^A > 0,\ A_i^0 < 0\}$. For each group, the admitted fraction is

$$f_\pm = \frac{1}{|\mathcal{I}_\pm|} \sum_{i \in \mathcal{I}_\pm} \mathbf{1}[\rho_i > 0].$$

Rejected trajectories remain in the distributions at $\rho_i = 0$, including trajectories with zero evidence strength. Zero-base-credit trajectories are outside these sign groups. Boxes show quartiles, whiskers use the 1.5-IQR rule, and diamonds mark means. Admission summarizes the influence permitted by the gate; the amount of credit actually moved also depends on evidence variation, as expressed by D_i in Eq. equation 14.

D Figure Definitions and Additional ALFWorld Results

D.1 Task-Suite Panels and Radar Scales

Figure 1 combines task-level and aggregate views. Its radar uses the Qwen2.5-3B values in Table 1; displayed differences compare EvoCred with GRPO. Panel (c) plots WebShop Score against success rate. Panel (d) plots the unweighted mean of NQ, TriviaQA, and PopQA against the unweighted mean of HotpotQA, 2WikiMultiHopQA, MuSiQue, and Bamboogle. The graphic's *SearchQA* label denotes the seven-dataset suite; the two grouped means in panel (d) summarize subsets of that suite.

For a metric value v on a radar spoke with display maximum a, the radial position is v/a. Table 6 lists these maxima. Comparisons between methods are made on the same spoke; the polygon area is not used as an aggregate score. WebShop Score differences are score points, while differences in percentage-valued metrics are percentage points.

Table 6: **Display maxima for the per-axis-normalized radar.** These are plotting scales, not empirical upper bounds.

Suite	Axes (clockwise within each suite)	Maxima
ALFWorld	Look, Pick, Clean, Heat, Cool, Pick2	90, 100, 100, 90, 100, 90
Search-QA	NQ, TriviaQA, PopQA, HotpotQA	50, 65, 50, 45
	2Wiki, MuSiQue, Bamboogle	45, 20, 70
WebShop	Score, Success	90, 80

D.2 Separate Learning–Compute Study

Panels (a–b) of Figure 1 report a separate base-initialized, single-training-seed ALFWorld study with Qwen2.5-3B. Table 7 summarizes final success, normalized learning-curve area, and launcher GPU-hours. EvoCred reaches 85.2% final success, exceeding GRPO, AgentOPSD, and SEED by 8.6, 7.1, and 2.4 pp, respectively. It also has the highest normalized AUC, 57.6, while using 55.9 launcher GPU-hours: more than GRPO's 49.5 and fewer than AgentOPSD's 75.6 and SEED's 60.8. The 85.2% endpoint belongs to this study; the 89.9% ALFWorld macro score in Table 1 comes from the main evaluation.

Table 7: **Additional Qwen2.5-3B ALFWorld study.** Final success, learning-curve area, and compute for the study in Figure 1(a–b).

Method	Final success (%)	Normalized AUC	Launcher GPU-hours
GRPO	76.6	52.9	49.5
AgentOPSD	78.1	52.4	75.6
SEED	82.8	55.9	60.8
EvoCred	85.2	57.6	55.9

D.3 Training Curves, Corruption, and Behavioral Case

Figure 3 plots training reward. Thick curves are seven-step centered moving averages, with faint traces showing the raw series. For an interior recorded point t, the displayed average is $\frac{1}{7}\sum_{j=-3}^{3}\bar{R}_{t+j}$, where $\bar{R}_t$ denotes the recorded training reward at point t. Figure 4 introduces corruption at step 80 and compares ordinary training with constrained and unconstrained corrupted-skill conditions. These reward trajectories complement the benchmark metrics and component results in Tables 1 and 2.

Figure 6 compares two independent execution trajectories from the same initial state under an eight-action budget. The baseline exhausts the budget without collecting the credit card, while the EvoCred-trained agent completes the task in four actions. Both actors execute without inference-time skills. This case budget is specific to the displayed comparison; the training turn limits appear in Table 4. Fixed-action rescoring is a training operation and leaves the recorded target trajectory unchanged.