UCOB: LEARNING TO UTILIZE AND EVOLVE AGENTIC SKILLS VIA CREDIT-AWARE ON-POLICY BIDIRECTIONAL SELF-DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Skills can improve agentic reinforcement learning by reusing past experience as textual guidance, but retrieved skills are not always reliable: they may help in one state while misleading in another. This makes the common privileged-teacher assumption fragile, namely that a skill-conditioned prompt can be treated as a fixed teacher for the no-skill prompt. We propose UCOB, a framework that first learns when to utilize agentic skills via credit-aware on-policy bidirectional self-distillation, then evolves them through utility-aware skill updates and reflection self-training. UCOB constructs two on-policy context views from skill-conditioned and no-skill prompts, compares return-to-go among rollouts sharing the same task and anchor state, and uses the higher-return view as the local teacher. The resulting local credit signal internalizes useful skill-conditioned behavior, corrects misleading skill usage, and further updates the dual-granularity skill memory, informs utility-aware retrieval, and supports reflection self-training. Experiments on ALFWORLD, WEBSHOP, and SEARCH-QA show that UCOB outperforms skill-free RL baselines, skill-augmented methods, and self-distillation baselines across model scales, achieving up to **23.5** and **18.0** absolute success-rate gains over SOTA baselines on ALFWORLD and WEBSHOP, respectively. Ablations and analyses further validate its core mechanisms, continual adaptation across environments, and modest training overhead.

Code page: `https://anonymous.4open.science/r/UCOB-7137`.

1 INTRODUCTION

Agentic reinforcement learning (RL) has become a central paradigm for improving language agents in long-horizon interactive tasks. Unlike single-turn reasoning, these environments require agents to act under partial observations, recover from local mistakes, and assign sparse outcome rewards to many intermediate decisions. A natural way to reduce this difficulty is to reuse past experience: agents can reflect on trajectories, store reusable textual skills, retrieve relevant skills in later states, and condition future actions on this external memory (Shinn et al., 2023; Wang et al., 2024). Recent skill-augmented agent methods show that such memory can improve exploration and sample efficiency (Tu et al., 2026; Shi et al., 2026). When these retrieved skills are further used for training, however, a stronger assumption often enters implicitly: the skill-conditioned behavior is treated as a better source of supervision. ***Can skill-conditioned context be safely treated as a privileged teacher during agent training?***

This assumption is appealing but fragile. Retrieved skills may be irrelevant, stale, overly generic, or locally mismatched to the current state, and the agent may fail to ground even useful skills in the present observation. Consequently, a skill-conditioned prompt can help in one state but mislead the same policy in another. Prior methods improve skill construction, curation, or skill-conditioned self-distillation (Ouyang et al., 2026; Wang et al., 2026; Lu et al., 2026a), but the supervision direction often remains fixed from the skill-conditioned view to the no-skill view. Our diagnostics in Section 2 challenge this fixed-teacher view: the skill-conditioned branch is not consistently better than the no-skill branch, and making skill-conditioned rollouts on-policy does not remove this ambiguity. ***When skill and no-skill views disagree at the same state, which view should teach the other?***

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.

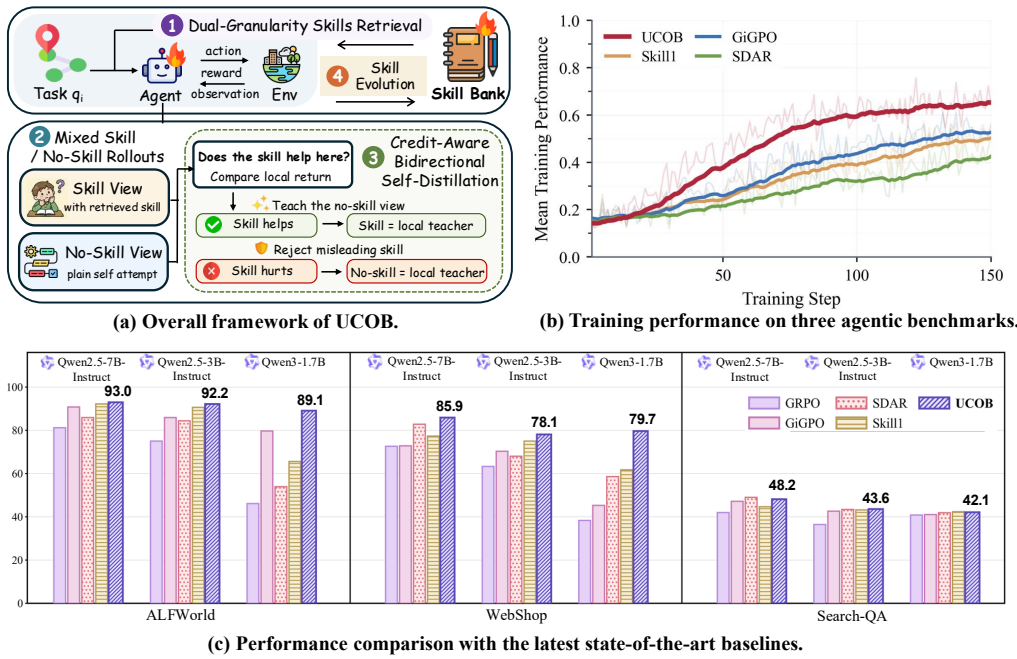


Figure 1: Overview and empirical summary of UCOB.

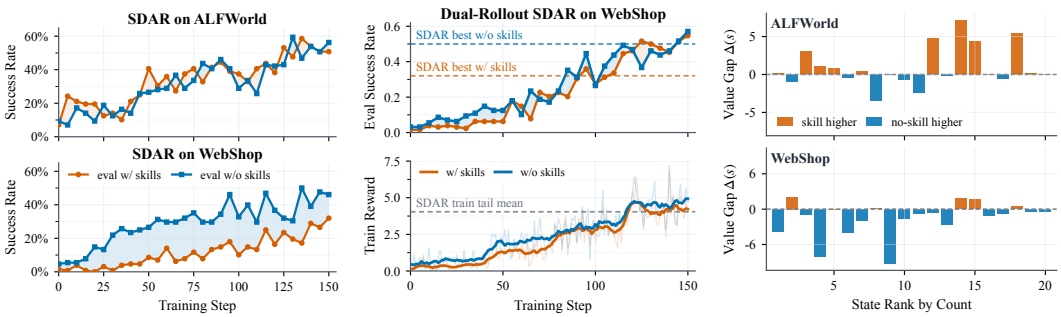
To address this, we propose UCOB (learning to **U**tilize and evolve agentic skills via **C**redit-aware **O**n-policy **B**idirectional self-distillation), which treats the skill-conditioned and no-skill prompts as two on-policy context views of the same model, rather than as a fixed teacher-student pair. The phrase *utilize and evolve* denotes a sequential loop: UCOB first determines whether retrieved skills should be trusted or corrected from local credit evidence, then uses it to improve the skill memory and skill writer. During training, it compares skill/no-skill rollouts within each task-anchor group and lets the higher-return view supervise the other, internalizing useful skill-conditioned behavior while correcting misleading usage. The credited evidence further updates the dual-granularity skill memory, guides utility-aware retrieval, and trains the reflection-based skill writer, closing the loop between policy learning and skill evolution. Figure 1 summarizes the framework and its behavior across training and evaluation.

Experiments on agentic tasks, including ALFWORLD, WEBSHOP, and SEARCH-QA, show that UCOB improves over skill-free RL, skill-augmented baselines, and self-distillation methods across model scales. Ablations verify the contribution of each module and design choice, while further analyses examine teacher routing, memory evolution, continual adaptation across sequential environments, and training cost. The main contributions of this work are as follows:

1. We motivate and propose **Credit-Aware Bidirectional Self-Distillation (CBSD)**, which selects on-policy teacher directions from same-task, same-anchor-state returns and is supported by a local policy-improvement analysis of credited bidirectional updates.
2. We develop **UCOB**, a unified framework that utilizes skills via CBSD and evolves them with dual-granularity skill memory, utility-aware retrieval, and reflection self-training.
3. We validate UCOB across model scales, outperforming SOTA baselines such as SDAR and Skill1 by up to **23.5** and **18.0** absolute success-rate gains on ALFWORLD and WEBSHOP; analyses further confirm its module and design effectiveness.

2 OBSERVATIONS

We examine whether skill-conditioned context can serve as a privileged teacher for multi-turn agent training through three diagnostics: **(i)** whether it reliably outperforms the no-skill view, **(ii)** whether on-policy skill rollouts resolve this issue, and **(iii)** what local evidence should determine distillation



(a) Fixed-teacher SDAR (b) Dual-rollout SDAR (c) State-group value gap

Figure 2: Three diagnostics motivating UCOB (Qwen3-1.7B). (a) Under fixed-teacher SDAR, the skill view does not consistently dominate the no-skill view across environments. (b) Under dual-rollout SDAR, both views are on-policy, yet fixed skill-to-no-skill distillation remains unreliable. (c) Same-anchor-state return gaps have mixed signs; orange/blue denote locally higher skill/no-skill returns. Appendix A provides protocols and the trajectory-step diagnostic.

direction when the views disagree. Figure 2 summarizes the evidence; detailed protocols and the trajectory-step diagnostic are deferred to Appendix A.

Skill-conditioned teachers are unreliable and not self-correcting. We first revisit SDAR (Lu et al., 2026a), an asymmetric setup that samples no-skill rollouts and uses the skill-conditioned prompt as a privileged teacher. This assumes that the skill-conditioned view provides a more reliable target on states induced by the no-skill rollout. Figure 2(a) challenges this assumption on ALFWORLD and WEBSHOP: the with-skill evaluation does not consistently dominate the no-skill evaluation, and on WEBSHOP, it remains weaker throughout training. Thus, ***adding a skill context does not automatically yield a reliable teacher***. Because the privileged view receives no return feedback under its own context distribution, a weak teacher is ***not self-correcting*** and may continue transferring supervision from unverified conditioned behavior.

Rollouts with skills mitigate exposure mismatch but not teacher ambiguity. A natural remedy is to place skills in rollouts so the skill-conditioned view receives environment feedback. We test a dual-rollout fixed-direction variant: half of each batch uses skills and half does not, while the SDAR-style skill-to-no-skill distillation direction remains fixed. Both views receive on-policy RL updates, but only the no-skill branch receives the extra distillation loss. Figure 2(b) shows that this stronger baseline narrows the gap between views, but still does not make the skill-conditioned view consistently better. On WEBSHOP, skill-conditioned rollouts have lower average reward than no-skill rollouts. Thus, ***making the skill-conditioned view on-policy does not make it universally authoritative***. The question is whether skill-induced behavior should be ***internalized when useful or corrected when misleading***.

Teacher direction is locally value-dependent. We compare the two views at the same task-anchor state and, following GiGPO (Feng et al., 2025), estimate average return-to-go within each group. For a rollout record that reaches state s_t, we use the return-to-go $G(s_t) = \sum_{\ell \geq 0} \gamma^\ell r_{t+\ell}$, average it for the skill-conditioned and no-skill views within each task-anchor group, and define $\Delta(\tilde{s}) = \bar{G}_+(\tilde{s}) - \bar{G}_0(\tilde{s})$, where $+$ and 0 denote the two views. A positive gap indicates useful skill context that can be internalized, whereas a negative gap indicates that the skill-conditioned behavior is locally less reliable and should be corrected by the no-skill view. Figure 2(c) ranks frequent state groups and plots the skill-minus-no-skill value gap. The gaps are mixed-sign in both environments, showing that ***teacher quality is not one-sided***; teacher direction should depend on the local value gap within each state group. The trajectory-step diagnostic in Appendix A further shows within-trajectory direction switches: a trajectory-level teacher may provide the right global direction but the wrong local update. Thus, ***trajectory summaries are too coarse for deciding each update***.

From observations to UCOB. These observations turn state grouping into credit-aware cross-context supervision. For each task-anchor state, matched skill/no-skill rollouts form a credited pair:

Table 1: Representative skill-augmented and self-distillation methods. Task-level skills are common and omitted; State-Level Skills denotes additional state skills. The SDAR triangle marks token-level gating without bidirectional teacher routing. Related ablations appear in Sections 4.3–4.4.

Method	Date (YY.M)	State-Level Skills	Credit-Aware Supervision	Skill-Free Inference	Evolving Skill Memory	Evolving Skill Policy
SkillRL (Xia et al., 2026)	26.2	×	×	×	✓	×
D2Skill (Tu et al., 2026)	26.3	✓	×	×	✓	×
RetroAgent (Zhang et al., 2026a)	26.3	×	×	×	✓	✓
Skill-SD (Wang et al., 2026)	26.4	×	×	✓	×	×
Skill1 (Shi et al., 2026)	26.5	×	×	×	✓	✓
SDAR (Lu et al., 2026a)	26.5	×	△	✓	×	×
StepOPSD (Zhang et al., 2026b)	26.5	×	×	✓	×	×
SkillC (Lin et al., 2026)	26.5	×	✓	✓	✓	×
SAPO (Zhang et al., 2026e)	26.6	×	×	×	✓	✓
UCOB (OURS)	**26.6**	✓	✓	✓	✓	✓

the higher-value branch teaches the opposite context, while the lower-value branch identifies an unreliable teacher. Unlike D2Skill (Tu et al., 2026), these pairs derive state skills from ***anchor-state value comparisons*** rather than trajectory-level summaries. Accordingly, UCOB generates and retrieves task- and state-level skills, uses ***local value gaps*** for ***credit-aware bidirectional self-distillation***, and evolves both the answer policy and skill generator. Table 1 positions UCOB against representative skill-augmented and self-distillation methods.

3 METHOD: UCOB

UCOB treats retrieved skills as candidate context rather than privileged labels, using the skill-conditioned and no-skill context views defined in Appendix B. Figure 3 summarizes the four-stage training loop. UCOB forms a closed loop that first utilizes retrieved skills as candidate context, then evolves the skill memory from credited interaction evidence. At its core, UCOB compares skill-conditioned and no-skill views at the same task-anchor state, distills from the higher-return view, and feeds the resulting credit back into policy and skill updates.

3.1 STEP 1: DUAL-GRANULARITY SKILL RETRIEVAL

Utility-aware UCB skill retrieval. UCOB maintains a dual-granularity skill memory, consisting of task-level and state-level skill pools, and expands it online from rollout reflections. We denote these pools by $\mathcal{M}_{\text{task}}$ and $\mathcal{M}_{\text{state}}$, respectively. Task-level skills provide reusable episode-level strategies, whereas state-level skills capture local decision rules grounded in credited same-state contrasts. Within each pool, candidate skills are ranked by retrieval relevance, online utility, and UCB exploration; the precise score and retrieval budgets are given in Appendix C. The selected skills are inserted into the skill-conditioned prompt $P_+(s_t)$, while the no-skill prompt $P_0(s_t)$ contains the same state information without retrieved skills.

3.2 STEP 2: MIXED ROLLOUTS AND ANCHOR-STATE GROUPING

Mixed skill/no-skill rollouts. For each task group u, UCOB samples mixed rollouts from the current policy under the two prompt views. Half of the trajectories act with the skill-conditioned prompt $P_+(s_t)$, and the remaining trajectories act with the no-skill prompt $P_0(s_t)$. We use $+$ and 0 to denote these two rollout branches throughout. Both branches receive online RL updates, while teacher selection is deferred to same-anchor-state return comparisons.

Anchor-state grouping. Following the anchor-state grouping mechanism of GiGPO (Feng et al., 2025), UCOB retroactively groups repeated states inside each task group. For each anchor state $\tilde{s}$ in task group u, we construct

$$\mathcal{C}(u, \tilde{s}) = \{\rho_{i,t} \mid \rho_{i,t} \text{ is from task } u,\ \text{anchor}(s_{i,t}) = \tilde{s}\}. \tag{1}$$

Each record stores its branch, prompts, sampled response, and return-to-go; the full representation is given in Appendix C. The opposite prompt is not used to generate the action, but enables a response

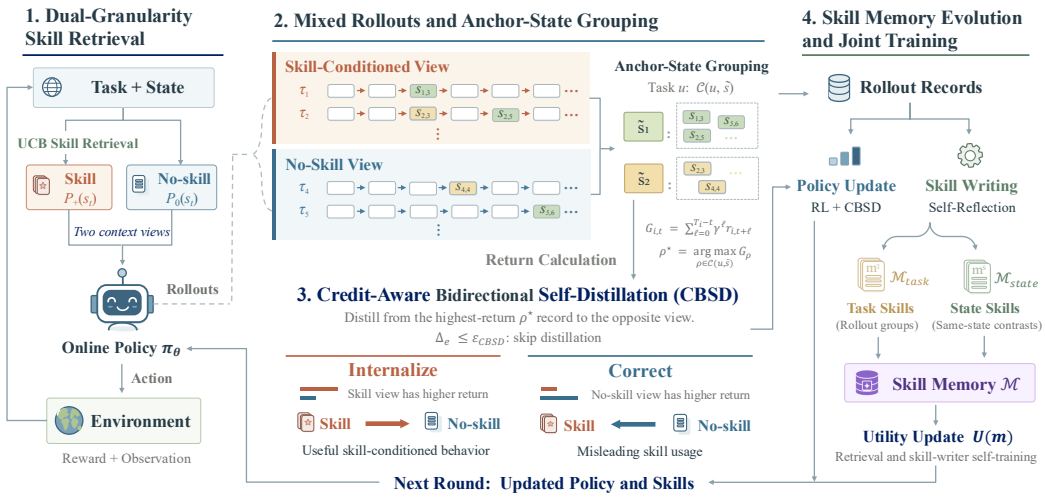


Figure 3: Overview of UCOB: skill retrieval and mixed rollouts support credit-aware bidirectional self-distillation (CBSD), while policy and skill updates close the training loop. CBSD internalizes useful skill-conditioned behavior and corrects misleading skill usage. Figure 9 in Appendix C provides the detailed framework.

sampled under one view to supervise the other. Thus, CBSD operates on same-task, same-anchor-state comparison sets.

3.3 Step 3: Credit-Aware Bidirectional Self-Distillation

CBSD is the core credit-assignment mechanism of UCOB. For each same-task, same-anchor-state group, it uses return evidence to decide which context view should teach the other, then distills the higher-return response distribution to the opposite view. Here *bidirectional* means credit-routed teacher selection: skill to no-skill when retrieved skills help, and no-skill to skill when the skill context is locally harmful.

Credited direction selection. For each anchor-state set $\mathcal{C}(u, \tilde{s})$, UCOB first selects the highest-return record as the credited reference:

$$\rho^\star = \arg\max_{\rho \in \mathcal{C}(u,\tilde{s})} G_\rho, \qquad c^\star = c_{\rho^\star}, \qquad y^\star = y_{\rho^\star}. \tag{2}$$

For each opposite-branch record ρ^{opp}, we form a candidate pair $e = (\rho^\star, \rho^{\text{opp}})$ with return gap $\Delta_e = G_{\rho^\star} - G_{\rho^{\text{opp}}}$ and accept it into $\mathcal{P}_{\text{CBSD}}$ only when $\Delta_e > \epsilon_{\text{CBSD}}$, where $\epsilon_{\text{CBSD}} \ge 0$ is a return-gap margin. For each accepted pair, P_e^{ref} is the acting prompt of $\rho^\star$ and P_e^{tgt} is the corresponding opposite-view prompt. The local winner is the reference and the opposite view is the target: $P_+ \to P_0$ when $c^\star = +$, and $P_0 \to P_+$ when $c^\star = 0$. Thus, teacher direction is selected by same-task, same-anchor-state return evidence rather than by the presence of retrieved skills.

Top-K token-support matching. For accepted pairs, UCOB matches the credited reference distribution on its token-level top-K support, following local-support OPD (Fu et al., 2026). At position j, the reference and target views are evaluated on the same credited prefix $y^\star_{<j}$ under $h^{\text{ref}}_{e,j} = (P_e^{\text{ref}}, y^\star_{<j})$ and $h^{\text{tgt}}_{e,j} = (P_e^{\text{tgt}}, y^\star_{<j})$. The reference support is $S_{e,j} = \text{TopK}^{K_{\text{tok}}}(\pi_\theta(\cdot \mid h^{\text{ref}}_{e,j}))$; on this support, we match the renormalized distributions

$$q_{e,j}(v) = \text{sg}\left[\frac{\pi_\theta(v \mid h^{\text{ref}}_{e,j})}{\sum_{z \in S_{e,j}} \pi_\theta(z \mid h^{\text{ref}}_{e,j})}\right], \qquad p_{e,j}(v) = \frac{\pi_\theta(v \mid h^{\text{tgt}}_{e,j})}{\sum_{z \in S_{e,j}} \pi_\theta(z \mid h^{\text{tgt}}_{e,j})}, \qquad v \in S_{e,j}. \tag{3}$$

Here K_{tok} is the token-support size and $\text{sg}[\cdot]$ denotes stop-gradient. Thus, $D^{\text{top}K}_{e,j} = \text{KL}(q_{e,j} \| p_{e,j})$ transfers the reference view's local distributional preference to the target view.

CBSD objective. Following SDAR (Lu et al., 2026a), a confidence gate suppresses positions where the credited reference is not more reliable than the target:

$$g_{e,j} = \mathrm{sg}\big[\sigma\big(\beta_{\mathrm{gate}}\left[\log \pi_\theta(y_j^\star \mid h_{e,j}^{\mathrm{ref}}) - \log \pi_\theta(y_j^\star \mid h_{e,j}^{\mathrm{tgt}})\right]\big)\big]. \tag{4}$$

Here $\sigma(\cdot)$ is the sigmoid function and β_{gate} controls gate sharpness. The gate is detached during optimization. The resulting credit-aware bidirectional self-distillation objective is

$$\mathcal{L}_{\mathrm{CBSD}} = \frac{1}{Z_{\mathrm{CBSD}}} \sum_{e \in \mathcal{P}_{\mathrm{CBSD}}} \sum_j \chi_{e,j} D_{e,j}^{\mathrm{top}K}, \tag{5}$$

where $\chi_{e,j} = m_{e,j} w_e g_{e,j}$, $w_e = \mathrm{clip}(\Delta_e / \tau_{\mathrm{CBSD}}, 0, w_{\max})$, and $Z_{\mathrm{CBSD}} = \sum_{e,j} m_{e,j}$. Here $m_{e,j}$ masks valid response tokens, τ_{CBSD} and $w_{\max}$ control the scale and cap of gap weighting, and $g_{e,j}$ applies the confidence gate. In practice, $\mathcal{L}_{\mathrm{CBSD}}$ provides auxiliary cross-context supervision alongside the online RL loss. Appendix H provides a local policy-improvement analysis of this credited direction selection.

3.4 Step 4: Skill Memory Evolution and Joint Training

UCOB closes the loop by writing skills, updating their utilities, and training the reflection-based skill writer to improve future skill generation.

Skill writing and utility update. The rollout records used for CBSD also provide evidence for writing and scoring skills. UCOB writes task-level skills by reflecting on rollout groups, and writes state-level skills by contrasting higher- and lower-return records within the same anchor-state set $\mathcal{C}(u, \tilde{s})$. A state-level skill records when the state occurs, which behavior worked better, and which behavior should be avoided; hence it is grounded in a local value contrast rather than a trajectory-level summary.

To score a used skill m, let b_u be the mean no-skill success of task group u, and let δ_m be the success improvement of rollouts using m over b_u. We compute this credit separately for task- and state-level skill pools. UCOB updates the skill utility by an exponential moving average (EMA) with rate β_U:

$$U(m) \leftarrow (1 - \beta_U) U(m) + \beta_U \delta_m. \tag{6}$$

The updated utilities feed back into the UCB retrieval score in Section 3.1, so skills that repeatedly help under on-policy rollouts become easier to retrieve later.

Skill-writer self-training. UCOB further improves the skill writer by replaying reflection prompt-response pairs with pool-normalized utility advantages. Thus, high-utility memories reinforce their reflection responses, whereas low-utility memories are downweighted or suppressed. The full reflection objective is provided in Appendix C.

Overall objective. UCOB minimizes the combined training loss:

$$\mathcal{L} = \mathcal{L}_{\mathrm{RL}} + \lambda_{\mathrm{CBSD}} \mathcal{L}_{\mathrm{CBSD}} + \lambda_{\mathrm{refl}} \mathcal{L}_{\mathrm{refl}} + \beta_{\mathrm{KL}} \mathcal{L}_{\mathrm{KL}}. \tag{7}$$

Here λ_{CBSD}, λ_{refl}, and β_{KL} balance the distillation, reflection, and KL regularization terms. The base losses $\mathcal{L}_{\mathrm{RL}}$ and $\mathcal{L}_{\mathrm{KL}}$ are defined in Appendix B; $\mathcal{L}_{\mathrm{CBSD}}$ is defined above and $\mathcal{L}_{\mathrm{refl}}$ in Appendix C. This objective couples online policy improvement with credited cross-context distillation and utility-guided skill evolution.

4 Experiments

4.1 Experimental Setup

We evaluate UCOB on ALFWorld, WebShop, and Search-QA across Qwen2.5-7B, Qwen2.5-3B, and Qwen3-1.7B backbones, comparing against skill-free RL, skill-augmented, and self-distillation baselines. Appendix D provides the full benchmarks, metrics, models, baselines, and evaluation protocol.

Table 2: Main performance on ALFWORLD, WEBSHOP, and SEARCH-QA. We report success rate (%) for ALFWORLD, Score/Succ (%) for WEBSHOP, and accuracy (%) for SEARCH-QA. For ALFWORLD, *Succ* is the overall success rate rather than an unweighted mean over subtasks, and for SEARCH-QA, *Avg* is the mean accuracy over the seven datasets. For each backbone, **bold** and underline mark the best and second-best values in the highlighted primary columns. Entries marked † are copied from the original papers, entries marked * are copied from SDAR (Lu et al., 2026a).

	ALFWORLD							WEBSHOP		SEARCH-QA							
Method	**Pick**	**Look**	**Clean**	**Heat**	**Cool**	**Pick2**	**Succ**	**Score**	**Succ**	**NQ**	**Triv**	**Pop**	**Hotp**	**2Wk**	**MuS**	**Bam**	**Avg**
Qwen2.5-7B-Instruct																	
GRPO*	91.2	87.5	96.2	81.0	65.0	57.9	81.2	80.9	72.6	45.1	63.7	44.0	43.6	43.2	16.8	37.6	42.0
GiGPO†	97.7	82.7	98.8	83.7	89.3	79.2	90.8	84.4	72.8	46.4	64.7	46.1	41.6	43.6	18.9	68.9	47.2
SkillRL†	97.9	71.4	90.0	90.0	95.5	87.5	89.9	85.2	72.7	45.9	63.3	45.9	43.2	40.3	20.2	73.8	47.5
D2Skill†	93.8	77.8	94.7	95.0	95.5	72.0	87.8	91.1	80.5	48.7	63.4	44.9	47.5	43.7	21.0	67.7	48.1
SkillC†	88.5	78.6	91.2	94.7	100.0	88.9	90.6	85.6	74.0	-	-	-	-	-	-	-	-
SAPO†	98.7	73.9	98.1	92.6	85.0	89.2	92.2	90.5	78.1	48.4	62.9	46.7	45.0	45.2	18.3	46.4	44.7
RLSD*	100.0	87.5	92.3	58.8	80.0	65.2	82.0	87.4	77.3	46.8	63.0	44.4	45.5	48.9	21.5	73.0	**49.0**
Skill-SD*	93.9	93.8	90.9	100.0	69.2	68.4	85.1	86.1	76.5	47.1	64.5	47.8	44.2	42.1	20.2	69.0	47.8
SDAR*	94.7	75.0	100.0	86.7	68.2	78.9	85.9	89.4	82.8	46.3	63.5	48.2	43.8	48.4	19.6	73.0	**49.0**
Skill1	100.0	83.3	89.5	100.0	88.5	85.0	92.2	86.2	77.3	46.8	46.3	47.8	43.7	39.3	18.2	70.6	44.7
UCOB	93.8	90.9	94.1	95.8	92.9	89.5	**93.0**	91.8	**85.9**	46.4	65.6	46.7	45.7	42.9	19.2	70.6	48.2
Qwen2.5-3B-Instruct																	
GRPO*	91.2	62.5	96.2	61.9	65.0	47.4	75.0	79.8	63.3	39.3	60.6	41.1	37.4	34.6	15.4	26.4	36.4
GiGPO	94.3	72.7	93.8	100.0	64.3	89.5	85.9	83.8	70.3	43.0	60.5	46.9	36.2	35.5	12.1	64.1	42.6
RLSD*	87.9	75.0	90.9	75.0	73.1	68.4	79.7	84.4	66.4	41.5	58.6	42.3	40.4	40.2	16.8	66.9	43.8
Skill-SD*	88.2	50.0	96.2	52.4	65.0	57.9	73.4	75.9	64.0	44.4	60.4	44.0	39.5	40.4	15.4	64.9	44.1
SDAR*	97.1	62.5	100.0	61.9	75.0	84.2	84.4	85.0	68.0	44.8	58.1	44.3	38.6	36.2	15.7	66.1	43.4
StepOPSD†	97.1	66.7	87.0	79.1	78.9	95.0	83.6	-	-	45.0	61.6	46.2	39.5	39.5	14.4	65.3	**44.5**
Skill1	96.7	76.9	96.3	93.8	77.3	95.0	90.6	84.1	75.0	44.2	60.3	44.8	39.2	35.9	13.1	64.9	43.2
UCOB	93.8	93.3	95.8	94.1	91.0	78.9	**92.2**	84.5	**78.1**	45.0	61.4	47.0	37.2	35.8	12.5	66.2	43.6
Qwen3-1.7B																	
GRPO*	71.1	41.7	36.4	40.0	31.8	31.6	46.1	67.3	38.3	40.0	58.9	43.5	35.4	30.3	12.0	65.7	40.8
GiGPO	96.7	69.3	88.9	62.5	59.1	85.0	79.7	70.0	45.3	41.1	59.3	45.1	34.1	32.7	10.5	64.5	41.0
RLSD*	50.0	37.5	61.5	19.0	50.0	21.1	42.2	74.0	50.8	38.6	57.3	43.0	34.5	34.1	11.5	65.3	40.6
Skill-SD*	52.9	37.5	69.2	42.9	60.0	36.8	52.3	81.8	53.9	39.1	57.5	45.4	34.8	34.1	10.7	64.1	40.8
SDAR*	73.5	25.0	76.9	33.3	40.0	36.8	53.9	76.8	58.6	39.7	58.9	45.3	35.9	35.5	12.6	65.3	41.9
StepOPSD†	64.7	44.4	56.5	60.9	42.1	55.0	56.3	-	-	40.5	59.4	44.4	37.1	32.0	11.6	64.1	41.3
Skill1	71.4	54.5	59.4	72.7	60.0	73.7	65.6	83.2	61.7	42.7	59.8	46.0	35.4	34.8	12.0	65.7	**42.3**
UCOB	96.2	85.7	95.5	84.2	81.5	88.8	**89.1**	91.5	**79.7**	44.1	59.6	45.7	36.8	31.9	12.2	64.5	42.1

4.2 MAIN RESULTS

Table 2 shows that UCOB achieves the **best overall success rate across all backbones** on both ALFWORLD (89.1–93.0) and WEBSHOP (78.1–85.9), with top or near-top WEBSHOP scores. The largest gains occur on Qwen3-1.7B: over Skill1, UCOB improves ALFWORLD/WEBSHOP success by $23.5/18.0$ points, and by $35.2/21.1$ points over SDAR. It also exceeds the strongest prior result on larger backbones by 0.8–3.1 points, while remaining within 0.2–0.9 points of the best SEARCH-QA average accuracy. The variation reflects both task bottlenecks and each backbone's ability to calibrate retrieved skills against current evidence. On the multi-turn interactive tasks, smaller models have greater headroom in planning, state tracking, and recovery, so dual-granularity memory and CBSD provide a more visible scaffold; SEARCH-QA relies more on search, evidence reading, and answer synthesis, where reusable state-level skills are less central.

4.3 ABLATION STUDY

Main component ablation. Figure 4 (left) ablates the major components of UCOB under the Qwen3-1.7B backbone. We report success rate and episode length, interpreting shorter episodes as more efficient only at comparable success. The "w/o" variants remove one component at a time: **(i)** *w/o anchor-state grouping* replaces local comparison with trajectory-level comparison; **(ii)** *w/o reflection self-training* removes utility-weighted skill-writer training; **(iii)** *w/o state skills* re-

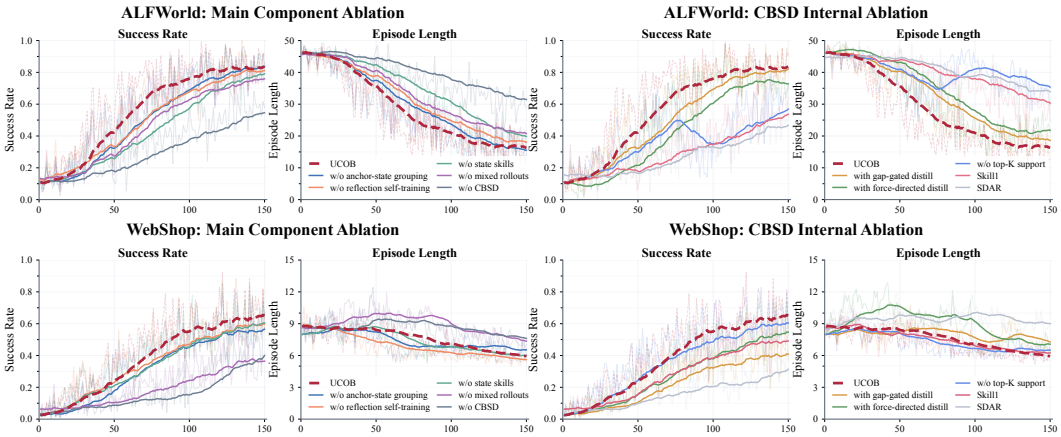


Figure 4: Ablation study under the Qwen3-1.7B backbone on ALFWORLD and WEBSHOP.

tains only task-level skills; **(iv)** *w/o mixed rollouts* removes paired skill/no-skill rollouts; and **(v)** *w/o CBSD* removes bidirectional distillation while retaining memory retrieval and reflection write-back. The first three variants degrade moderately, showing complementary contributions from anchor-state credit, state-level memory, and skill-writer self-training. The larger losses from *w/o mixed rollouts* and *w/o CBSD* show that memory-augmented rollouts alone leave skill/no-skill comparison to sparse RL returns rather than converting local gaps into dense supervision. Thus, for credit-assignment RL, **directly mixing skills into on-policy rollouts is not a substitute for an explicit local distillation signal**.

CBSD internal ablation. Figure 4 (right) isolates the design choices inside CBSD. It compares full CBSD with **(i)** *gap-gated distill*, which retains the gap gate but fixes skill-to-no-skill distillation; **(ii)** *force-directed distill*, which fixes direction without local return comparison; and **(iii)** *w/o top-K support*, which removes the distributional support used to construct the OPD target. The first two variants show that, even with a gap gate, one-way or fixed-direction distillation is less robust than routing supervision from the higher-return view; skill-conditioned prompts are useful but not universally authoritative. *w/o top-K support* retains credit-aware routing but loses support matching, indicating that top-K OPD stabilizes the target rather than replacing local credit assignment. **The ablated variants remain competitive with, and often stronger than, Skill1 and SDAR**, while full CBSD gives the strongest performance stability across environments.

Additional ablation. Table 3 summarizes two additional ablations on key design axes. For CBSD, a smaller λ_{CBSD} weakens distillation, while a larger one can disrupt the balance between RL and bidirectional self-distillation; the default 0.1 performs best on both environments. For skill evolution, the fixed memory initialized from SkillRL (Xia et al., 2026) lags behind evolving memories, showing that static external skills cannot replace continual memory updates. External GPT-5.4 remains below self-trained reflection overall, indicating that UCOB's gains do not rely on a stronger external writer.

Table 3: Additional ablation

Setting	ALFWORLD	WEBSHOP
CBSD coeff. λ_{CBSD}		
0.01	83.6	71.1
0.1 (Ours)	**89.1**	**79.7**
0.5	78.1	68.7
Reflection backend		
Fixed memory	82.0	70.3
External GPT-5.4	88.3	72.7
Self-trained (Ours)	**89.1**	**79.7**

4.4 ANALYSIS

CBSD routing and two-view evaluation. Figure 5(a) tracks the teacher direction selected by CBSD during training. No-skill-to-skill routing occurs more often than skill-to-no-skill routing, indicating that reflection skills are not uniformly reliable local teachers. This matches the local policy-improvement analysis in Appendix H: when the return gap changes sign, a fixed skill-to-no-skill update can be harmful, whereas CBSD follows the

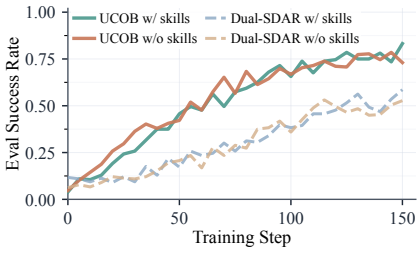


Figure 6: Skill/no-skill evaluation.

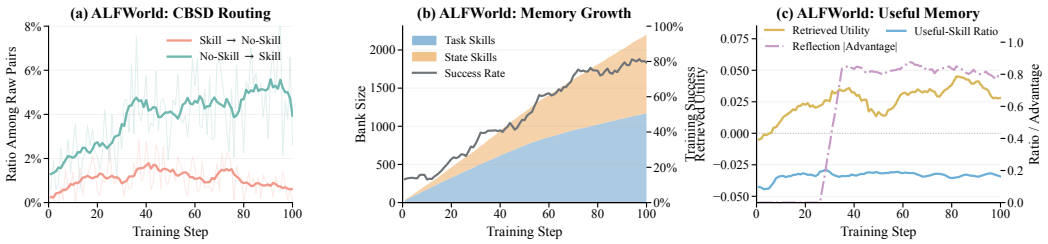


Figure 5: Mechanism and memory-evolution analysis of UCOB on ALFWORLD with QWEN3-1.7B. The WEBSHOP counterpart is in Appendix F.1.

higher-return view. Thus, **the skill-conditioned view is not a fixed privileged teacher**. Figure 6 compares skill/no-skill evaluation success with QWEN3-1.7B, averaged over ALFWORLD and WEBSHOP. Under fixed-direction dual-rollout SDAR, the skill-conditioned view is not consistently dominant; UCOB instead improves both views through bidirectional distillation. **The resulting policy remains effective whether skills are provided at evaluation or not.**

Useful memory evolution. Figure 5(b) shows that the task- and state-level skill pools expand throughout training. As this memory grows, panel (c) tracks the utility of retrieved skills, the fraction whose utility exceeds its initialization, and the reflection-writer advantage magnitude. Retrieved utility remains positive later in training, alongside a non-trivial useful-skill ratio and active reflection signal. **Thus, the growing memory continues to supply useful retrieved skills rather than merely accumulating entries.**

Additional evidence. Appendix F complements this analysis with localized training-cost measurements, continual adaptation across environments, and qualitative cases of local teacher routing. The additional CBSD computation remains localized, while sequential training shows that the evolving memory retains a favorable adaptation–retention balance. The local policy-improvement analysis in Appendix H provides the corresponding formal perspective: credited updates follow the higher-return view, rather than assuming a fixed privileged skill teacher. **Together, these results support UCOB as an efficient and adaptive alternative to fixed-direction skill distillation.**

5 RELATED WORK

Skill learning and local credit assignment. Prior work retrieves, curates, and evolves reusable memories or skills for agent interaction (Shinn et al., 2023; Zhao et al., 2024; Wang et al., 2024; Xia et al., 2026; Tu et al., 2026; Shi et al., 2026). Sparse multi-turn rewards motivate fine-grained credit assignment (Feng et al., 2025; Wei et al., 2025; Zhu et al., 2026; Li et al., 2026a; Liu et al., 2026). **On-policy self-distillation with privileged context.** On-policy self-distillation supplies dense auxiliary RL supervision (Zhao et al., 2026a; Yang et al., 2026; Fu et al., 2026; Li et al., 2026b). Skill-SD and SDAR use retrieved skills as a fixed privileged teacher (Wang et al., 2026; Lu et al., 2026a). UCOB instead uses same-anchor-state returns to select the supervising view, extending credit assignment beyond action tokens to internalize or correct imperfect skill-conditioned behavior while evolving skill memory. Appendix G provides an extended discussion.

6 CONCLUSION

Skill-conditioned teachers are state-dependent, so even on-policy skill rollouts require local credit assignment. We introduced UCOB to utilize and evolve reusable but imperfect skills via credit-aware on-policy bidirectional self-distillation. CBSD selects the higher-return skill/no-skill view within the same task and anchor state as the local teacher, internalizing useful behavior and correcting misleading skill context. With dual-granularity skill memory evolution, utility-aware retrieval, and reflection self-training, UCOB achieves consistent gains on ALFWORLD, WEBSHOP, and SEARCH-QA. Analyses show improved context views, useful evolving memories, robust sequential adaptation, and localized training overhead. Skill utilization is thus credit assignment: beyond retrieval, agents should learn when to trust, correct, and internalize skills during on-policy training.

AI USE STATEMENT

In this work, generative AI tools were used to assist with literature search and organization, research ideation and methodology discussion, formulation and checking of theoretical arguments, manuscript drafting and language editing, and auxiliary development and analysis of experimental code. All AI-assisted text, mathematical claims, citations, code, figures, and analyses were reviewed and verified by the authors against the underlying literature, derivations, implementations, and experimental records. The authors take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Yurun Chen, Biao Yi, Keting Yin, and Shengyu Zhang. SkillHarness: Harnessing Safe Skills for Computer-Use Agents. *arXiv preprint arXiv:2606.20636*, 2026. URL https://arxiv.org/abs/2606.20636.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-Group Policy Optimization for LLM Agent Training. In *Advances in Neural Information Processing Systems*, volume 38, pp. 46375–46408, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/420c9f777c0b4f78d515e53cf74d58b2-Paper-Conference.pdf.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting On-Policy Distillation: Empirical Failure Modes and Simple Fixes. *arXiv preprint arXiv:2603.25562*, 2026. URL https://arxiv.org/abs/2603.25562.

Zhongyu He, Yuanfan Li, Fei Huang, Tianyu Chen, Siyuan Chen, Xingyang Li, Meng Hsuan Yu, Xiangrong Liu, Leyi Wei, Lu Pan, Ke Zeng, and Xunliang Cai. SIRI: Self-Internalizing Reinforcement Learning with Intrinsic Skills for LLM Agent Training. *arXiv preprint arXiv:2606.02355*, 2026. URL https://arxiv.org/abs/2606.02355.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Ö. Arık, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning. In *Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=Rwhi91ideu.

Sham M. Kakade and John Langford. Approximately Optimal Approximate Reinforcement Learning. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pp. 267–274, 2002. URL https://people.eecs.berkeley.edu/~pabbeel/cs287-fa09/readings/KakadeLangford-icml2002.pdf.

Jeonghye Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Rebellious Student: Reversing Teacher Signals for Reasoning Exploration with Self-Distilled RLVR. *arXiv preprint arXiv:2605.10781*, 2026. URL https://arxiv.org/abs/2605.10781.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

Xiaozhe Li, Tianyi Lyu, Yang Li, Yichuan Ma, Peiji Li, Linyang Li, Qipeng Guo, Dahua Lin, and Kai Chen. What and When to Distill: Selective Hindsight Distillation for Multi-Turn Agents. *arXiv preprint arXiv:2605.19447*, 2026a. URL https://arxiv.org/abs/2605.19447.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking On-Policy Distillation of Large Language Models: Phenomenology, Mechanism, and Recipe. *arXiv preprint arXiv:2604.13016*, 2026b. URL https://arxiv.org/abs/2604.13016.

Yu Li, Mingyang Yi, Xiuyu Li, Ju Fan, Fuxin Jiang, Binbin Chen, Peng Li, Jie Song, and Tieying Zhang. Reasoning and Tool-use Compete in Agentic RL:From Quantifying Interference to Disentangled Tuning. *arXiv preprint arXiv:2602.00994*, 2026c. URL https://arxiv.org/abs/2602.00994.

Hongxiang Lin, Zhirui Kuai, Erpeng Xue, and Lei Wang. SKILLC: Learning Autonomous Skill Internalization in LLM Agents via Contrastive Credit Assignment. *arXiv preprint arXiv:2605.27899*, 2026. URL https://arxiv.org/abs/2605.27899.

Zirui Liu, Tingyue Pan, Jie Ouyang, Qi Liu, Xianquan Wang, Jiayu Liu, Qingchuan Li, Jing Sha, Zhenya Huang, Shijin Wang, and Enhong Chen. SocraticPO: Policy Optimization via Interactive Guidance. *arXiv preprint arXiv:2606.09887*, 2026. URL https://arxiv.org/abs/2606.09887.

Zhengxi Lu, Zhiyuan Yao, Zhuowen Han, Zi-Han Wang, Jinyang Wu, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, and Yongliang Shen. Self-Distilled Agentic Reinforcement Learning. *arXiv preprint arXiv:2605.15155*, 2026a. URL https://arxiv.org/abs/2605.15155.

Zhengxi Lu, Zhiyuan Yao, Jinyang Wu, Chengcheng Han, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, and Yongliang Shen. SKILL0: In-Context Agentic Reinforcement Learning for Skill Internalization. *arXiv preprint arXiv:2604.02268*, 2026b. URL https://arxiv.org/abs/2604.02268.

Qirui Mi, Zhijian Ma, Mengyue Yang, Yisen Wang, Haoxuan Li, Haifeng Zhang, and Jun Wang. ProcMEM: Learning Reusable Procedural Memory from Experience via Non-Parametric PPO for LLM Agents. In *International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/65830.

Siru Ouyang, Jun Yan, Yanfei Chen, Rujun Han, Zifeng Wang, Bhavana Dalvi Mishra, Rui Meng, Chun-Liang Li, Yizhu Jiao, Kaiwen Zha, Maohao Shen, Vishy Tirumalashetty, George Lee, Jiawei Han, Tomas Pfister, and Chen-Yu Lee. SkillOS: Learning Skill Curation for Self-Evolving Agents. *arXiv preprint arXiv:2605.06614*, 2026. URL https://arxiv.org/abs/2605.06614.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-Weighted Regression: Simple and Scalable Off-Policy Reinforcement Learning. *arXiv preprint arXiv:1910.00177*, 2019. URL https://arxiv.org/abs/1910.00177.

Qwen Team. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

Andrei A. Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy Distillation. In *International Conference on Learning Representations*, 2016. URL https://www.iclr.cc/archive/www/2016.html.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust Region Policy Optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pp. 1889–1897, 2015. URL https://proceedings.mlr.press/v37/schulman15.html.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-Distillation Enables Continual Learning. In *International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/61434.

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. URL https://arxiv.org/abs/2605.06130.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1b44b878bb782e6954cd888628510e90-Paper-Conference.pdf.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. ALFWorld: Aligning Text and Embodied Environments for Interactive Learning. In *International Conference on Learning Representations*, 2021. URL https://iclr.cc/virtual/2021/poster/2973.

Mingyang Song and Mao Zheng. A Survey of On-Policy Distillation for Large Language Models. *arXiv preprint arXiv:2604.00626*, 2026. URL https://arxiv.org/abs/2604.00626.

Manan Tomar, Lior Shani, Yonathan Efroni, and Mohammad Ghavamzadeh. Mirror Descent Policy Optimization. In *International Conference on Learning Representations*, 2022. URL https://iclr.cc/virtual/2022/poster/6402.

Songjun Tu, Chengdong Xu, Qichao Zhang, Yaocheng Zhang, Xiangyuan Lan, Linjing Li, Dong Li, and Dongbin Zhao. Dynamic Dual-Granularity Skill Bank for Agentic RL. *arXiv preprint arXiv:2603.28716*, 2026. URL https://arxiv.org/abs/2603.28716.

Yash Vishe, Rohan Surana, Xunyi Jiang, Zihan Huang, Xintong Li, Nikki Lijing Kuang, Tong Yu, Ryan A. Rossi, Jingbo Shang, Julian McAuley, and Junda Wu. Skill-R1: Agent Skill Evolution via Reinforcement Learning. *arXiv preprint arXiv:2605.09359*, 2026. URL https://arxiv.org/abs/2605.09359.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An Open-Ended Embodied Agent with Large Language Models. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=ehfRiF0R3a.

Hao Wang, Guozhi Wang, Han Xiao, Yufeng Zhou, Yue Pan, Jichao Wang, Ke Xu, Yafei Wen, Xiaohu Ruan, Xiaoxin Chen, and Honggang Qi. Skill-SD: Skill-Conditioned Self-Distillation for Multi-turn LLM Agents. *arXiv preprint arXiv:2604.10674*, 2026. URL https://arxiv.org/abs/2604.10674.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, Yiping Lu, Kyunghyun Cho, Jiajun Wu, Li Fei-Fei, Lijuan Wang, Yejin Choi, and Manling Li. RAGEN: Understanding Self-Evolution in LLM Agents via Multi-Turn Reinforcement Learning. *arXiv preprint arXiv:2504.20073*, 2025. URL https://arxiv.org/abs/2504.20073.

Quan Wei, Siliang Zeng, Chenliang Li, Zhongruo Wang, William Brown, Oana Frunza, Wei Deng, Anderson Schneider, Yuriy Nevmyvaka, Yang Katie Zhao, Alfredo Garcia, and Mingyi Hong. Reinforcing Multi-Turn Reasoning in LLM Agents via Fine-Grained Reward Structure and Credit Assignment. *arXiv preprint arXiv:2505.11821*, 2025. URL https://arxiv.org/abs/2505.11821.

Peng Xia, Jianwen Chen, Hanyang Wang, Jiaqi Liu, Kaide Zeng, Yu Wang, Siwei Han, Yiyang Zhou, Xujiang Zhao, Haifeng Chen, Zeyu Zheng, Cihang Xie, and Huaxiu Yao. SkillRL: Evolving Agents via Recursive Skill-Augmented Reinforcement Learning. *arXiv preprint arXiv:2602.08234*, 2026. URL https://arxiv.org/abs/2602.08234.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng

Zhou, and Zihan Qiu. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

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*, 2026. URL https://arxiv.org/abs/2604.03128.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. WebShop: Towards Scalable Real-World Web Interaction with Grounded Language Agents. In *Advances in Neural Information Processing Systems*, volume 35, pp. 20744–20757, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/82ad13ec01f9fe44c01cb91814fd7b8c-Paper-Conference.pdf.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing Reasoning and Acting in Language Models. In *International Conference on Learning Representations*, 2023. URL https://iclr.cc/virtual/2023/poster/11003.

Xiaoying Zhang, Zichen Liu, Yipeng Zhang, Xia Hu, and Wenqi Shao. RetroAgent: From Solving to Evolving via Retrospective Dual Intrinsic Feedback. *arXiv preprint arXiv:2603.08561*, 2026a. URL https://arxiv.org/abs/2603.08561.

Yanfei Zhang, Xu Lin, and Chenglin Wu. StepOPSD: Step-Aware Online Preference Self-Distillation for Agent Reinforcement Learning. *arXiv preprint arXiv:2605.27140*, 2026b. URL https://arxiv.org/abs/2605.27140.

Yaocheng Zhang, Jiajun Chai, Yuqian Fu, Songjun Tu, Xiaohan Wang, Wei Lin, Guojun Yin, Qichao Zhang, Yuanheng Zhu, and Dongbin Zhao. Are Full Rollouts Necessary for On-Policy Distillation? *arXiv preprint arXiv:2605.31490*, 2026c. URL https://arxiv.org/abs/2605.31490.

Yaocheng Zhang, Yuanheng Zhu, Wenyue Chong, Songjun Tu, Qichao Zhang, Jiajun Chai, Xiaohan Wang, Wei Lin, Guojun Yin, and Dongbin Zhao. π-Play: Multi-Agent Self-Play via Privileged Self-Distillation without External Data. *arXiv preprint arXiv:2604.14054*, 2026d. URL https://arxiv.org/abs/2604.14054.

Zeyu Zhang, Quanyu Dai, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. A Survey on the Memory Mechanism of Large Language Model-based Agents. *ACM Transactions on Information Systems*, 43(6):1–47, 2025. URL https://doi.org/10.1145/3748302.

Zhiwei Zhang, Yudi Lin, Nikki Lijing Kuang, Linlin Wu, Xiaomin Li, Songtao Liu, and Fenglong Ma. Co-Evolving Skill Generation and Policy Optimization. *arXiv preprint arXiv:2606.08755*, 2026e. URL https://arxiv.org/abs/2606.08755.

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. URL https://ojs.aaai.org/index.php/AAAI/article/view/29936.

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*, 2026a. URL https://arxiv.org/abs/2601.18734.

Ziqi Zhao, Xinyu Ma, Liu Yang, Yujie Feng, Daiting Shi, Jingzhou He, Xin Xin, Zhaochun Ren, and Xiao-Ming Wu. ROSD: Reflective On-Policy Self-Distillation for Language Model Reasoning across Domains. *arXiv preprint arXiv:2605.28014*, 2026b. URL https://arxiv.org/abs/2605.28014.

Siyuan Zhu, Chao Yu, Rongxin Yang, Zongkai Liu, Jinjun Hu, Qiwen Chen, and Yibo Zhang. GAGPO: Generalized Advantage Grouped Policy Optimization. *arXiv preprint arXiv:2605.13217*, 2026. URL https://arxiv.org/abs/2605.13217.

A Detailed Protocols for the Observation Study

We give the full experimental settings and protocols behind the observation study in Section 2, whose schematic is shown in Figure 8. We use P_+ and P_0 to denote the skill-conditioned and no-skill prompts, respectively. The first two protocols instantiate fixed-direction on-policy self-distillation (OPSD), where the skill-conditioned view supervises the no-skill view through $\mathcal{L}_{\mathrm{OPSD}}$ (Lu et al., 2026a; Zhao et al., 2026a). The third protocol is a diagnostic: it does not fix a teacher, but instead groups matched anchor states and compares return-to-go under the two prompt views. Throughout this appendix, $\mathcal{L}_{\mathrm{RL}}$ denotes the online RL loss used for environment rollouts.

Fixed teacher without skill rollouts. This protocol matches the SDAR setting analyzed above (Lu et al., 2026a). Only the no-skill prompt P_0 is executed in the environment, so rollout records are generated and optimized under $\mathcal{L}_{\mathrm{RL}}$ from the no-skill view. The skill-conditioned prompt P_+ is evaluated only by a teacher forward pass on the same student-generated prefixes, providing the additional $\lambda\,\mathcal{L}_{\mathrm{OPSD}}$ target for P_0. Thus, P_+ is treated as a privileged self-teacher without receiving rollout-based RL updates under its own context. This setting tests whether a skill-conditioned teacher without skill rollouts can be both reliable and self-correcting; the corresponding two-environment curves are reported in Figure 2(a).

Dual rollouts with fixed-direction OPSD. This protocol extends the fixed-direction setting above with dual rollouts. Each batch executes both prompt views: half of the trajectories act under P_+ and half act under P_0, so both views receive $\mathcal{L}_{\mathrm{RL}}$ from environment interaction. However, the distillation direction is still fixed: P_+ supervises P_0, and only the no-skill branch receives the additional $\lambda\,\mathcal{L}_{\mathrm{OPSD}}$ term. This isolates whether making the skill-conditioned teacher on-policy is sufficient. Figure 2(b) shows that dual rollouts reduce exposure mismatch, but do not remove teacher ambiguity because the fixed teacher can still be locally weaker than the no-skill view.

Anchor-state value-gap diagnostic. This protocol removes the fixed teacher direction and serves as the local return-gap diagnostic. Using the anchor-state grouping mechanism of GiGPO (Feng et al., 2025), we group rollout records that share the same task u and anchor state $\tilde{s}$; each record retains its acting branch $c_{i,t} \in \{+, 0\}$ and return-to-go $G_{i,t} = \sum_{\ell \geq 0} \gamma^{\ell} r_{i,t+\ell}$. Within each group, we average return-to-go by branch to obtain $G_+(\tilde{s})$ and $G_0(\tilde{s})$, and use the local value gap $\Delta(\tilde{s}) = G_+(\tilde{s}) - G_0(\tilde{s})$ to diagnose which view is more reliable. This comparison is well-posed because both views are scored under the same task and canonical state, isolating the local effect of skill conditioning from trajectory-level differences. Positive gaps indicate states where the skill-conditioned view is the better local teacher, whereas negative gaps indicate states where the no-skill view is more reliable. Figure 2(c) shows that these gaps are mixed-sign, while the trajectory-step diagnostic in Figure 7 shows that they can switch across steps. Together, these results motivate per-state teacher selection rather than a fixed skill-to-no-skill direction.

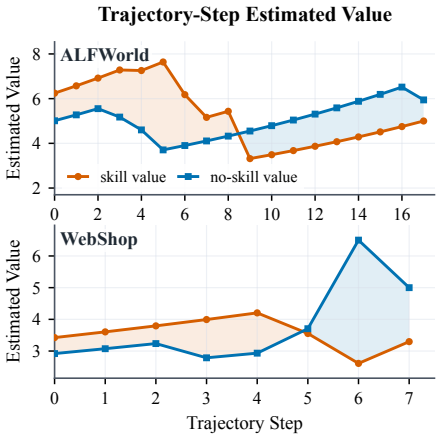
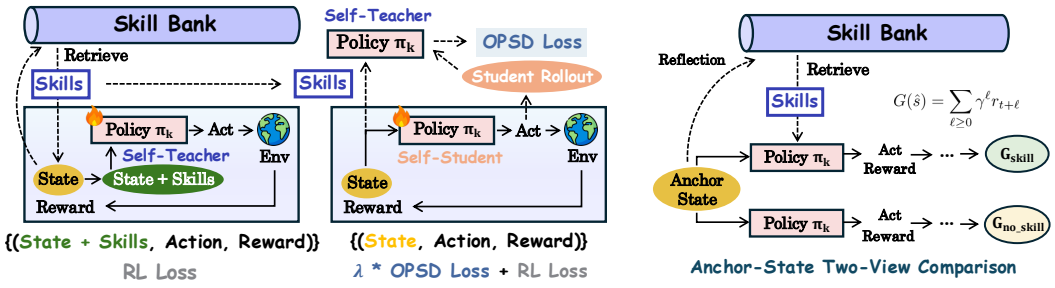


Figure 7: Trajectory-step value diagnostics for two-view rollouts. The locally better view can switch within a trajectory, ruling out a fixed trajectory-level teacher.

B Problem Setup and Preliminaries

Problem setup. We consider reinforcement learning for multi-turn language agents with retrieved skills. Each task instance is indexed by u. At turn t, the agent observes a state s_t containing the task description, current observation, and interaction history. The policy π_θ generates a textual response $y_t = (y_{t,1}, \ldots, y_{t,L_t})$, from which an executable action is parsed and sent to the environment. Let $\gamma \in [0, 1)$ denote the discount factor, and write $\tilde{s} = \mathrm{anchor}(s)$ for the canonical anchor-state abstraction used in local comparisons. The environment returns reward r_t and the next state; we denote the return-to-go by $G_t = \sum_{\ell \geq 0} \gamma^{\ell} r_{t+\ell}$. A skill-conditioned agent retrieves textual skills



(a) Fixed-direction skill/no-skill protocol (b) Anchor-state return-gap diagnostic

Figure 8: Full observation-study protocols. Panel (a) covers the no-skill-only and dual-rollout fixed-direction settings, while panel (b) compares the two context views at the same task-anchor state.

$M(s_t)$ and forms a skill-conditioned prompt $P_+(s_t) = \text{Prompt}(s_t, M(s_t))$, while the no-skill prompt is $P_0(s_t) = \text{Prompt}(s_t, \emptyset)$. These prompts induce two context views of the same policy, $\pi_+(\cdot \mid s_t) = \pi_\theta(\cdot \mid P_+(s_t))$ and $\pi_0(\cdot \mid s_t) = \pi_\theta(\cdot \mid P_0(s_t))$.

Skill-conditioned OPSD. On-policy self-distillation (OPSD) trains on student-sampled rollouts and minimizes a per-token divergence between teacher and student distributions along these on-policy prefixes (Zhao et al., 2026a). For a sampled response y, teacher prompt P_{tea}, and student prompt P_{stu}, we write a generic OPSD loss as

$$\mathcal{L}_{\text{OPSD}} = \frac{1}{Z_{\text{OPSD}}} \sum_j m_j D_{\text{KL}}(\text{sg}[\pi_\theta(\cdot \mid P_{\text{tea}}, y_{<j})] \,, \| \, \pi_\theta(\cdot \mid P_{\text{stu}}, y_{<j})) \,, \tag{8}$$

where m_j masks valid response tokens, $Z_{\text{OPSD}} = \sum_j m_j$ normalizes over valid positions, and $\text{sg}[\cdot]$ denotes stop-gradient on the teacher distribution. Skill-conditioned variants such as Skill-SD and SDAR typically set the skill-conditioned view as the teacher, i.e., $P_{\text{tea}} = P_+$ and $P_{\text{stu}} = P_0$ (Wang et al., 2026; Lu et al., 2026a). This fixed direction provides dense supervision but does not verify whether the skill-conditioned view is locally better.

On-policy agentic RL and credit assignment. Our online RL backbone follows GiGPO (Feng et al., 2025), which extends group-based RL with step-level credit for multi-turn agents. For the rollout group of task u, let $\tau_i = \{(s_{i,t}, y_{i,t}, r_{i,t})\}_{t=1}^{T_i}$ be a rollout from $\pi_{\theta_{\text{old}}}$, $R_i = \sum_t r_{i,t}$ its episode return, and $G_{i,t}$ the return-to-go at turn t. GiGPO combines episode-level credit with anchor-state credit:

$$A_i^E = \frac{R_i - \mu_u^E}{F_u^E}, \quad A_{i,t}^S = \frac{G_{i,t} - \mu_{u,\tilde{s}}^S}{F_{u,\tilde{s}}^S}, \quad A_{i,t} = A_i^E + \omega A_{i,t}^S, \quad \tilde{s} = \text{anchor}(s_{i,t}). \tag{9}$$

Here μ and F are the mean and normalizer within the corresponding episode or anchor-state group, and ω weights step-level credit. Broadcasting $A_{i,t}$ to valid response tokens yields the clipped on-policy loss

$$\mathcal{L}_{\text{RL}} = -\frac{1}{Z_{\text{RL}}} \sum_{i,t,j} m_{i,t,j} \min(\varrho_{i,t,j} A_{i,t}, \text{clip}(\varrho_{i,t,j}, 1 - \epsilon_{\text{clip}}, 1 + \epsilon_{\text{clip}}) A_{i,t}) \,, \tag{10}$$

where $m_{i,t,j}$ is the valid-token mask and $Z_{\text{RL}} = \sum_{i,t,j} m_{i,t,j}$. Let $c_{i,t} \in \{+, 0\}$ denote whether the response was sampled under the skill-conditioned or no-skill prompt. The token-level context is $h_{i,t,j} = (P_{c_{i,t}}(s_{i,t}), y_{i,t,<j})$, and the importance ratio is $\varrho_{i,t,j} = \pi_\theta(y_{i,t,j} \mid h_{i,t,j}) / \pi_{\theta_{\text{old}}}(y_{i,t,j} \mid h_{i,t,j})$. Here ϵ_{clip} is the PPO-style clipping range. We additionally use the reference-policy regularizer $\mathcal{L}_{\text{KL}} = \mathbb{E}[D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})]$.

C Full Training Algorithm

This appendix provides the operational details omitted from the main method and the complete training procedure. Figure 9 expands the compact overview in Figure 3.

Retrieval score and decision records. Each skill item m contains a textual principle, an applicability condition, a retrieval key k_m, a utility score $U(m)$, and a usage count n_m. For each pool $p \in \{\text{task}, \text{state}\}$, UCOB ranks candidate skills with an Upper Confidence Bound (UCB) score:

$$R_p(m; s_t) = \alpha \, \mathrm{sim}(q_p(s_t), k_m) + (1-\alpha) \left(U(m) + \eta \sqrt{\frac{\log(N_p+1)}{n_m+1}} \right), \tag{11}$$

where $q_p(s_t)$ is the task- or state-level retrieval query, $\mathrm{sim}(\cdot,\cdot)$ is the retrieval similarity, α balances relevance against utility and exploration, N_p is the total usage count of pool p, and η controls UCB exploration. The retrieved memory sets are

$$M_p(s_t) = \mathrm{TopK}^{K_p^{\mathrm{mem}}}_{m \in \mathcal{M}_p} R_p(m; s_t), \qquad p \in \{\text{task}, \text{state}\}. \tag{12}$$

Here K_p^{mem} is the skill retrieval budget for pool p. Each decision record in Eq. (1) is

$$\rho_{i,t} = \big(s_{i,t}, c_{i,t}, P_{c_{i,t}}(s_{i,t}), P_{\bar{c}_{i,t}}(s_{i,t}), y_{i,t}, G_{i,t}\big), \tag{13}$$

where $c_{i,t} \in \{+, 0\}$ is the rollout branch, $\bar{c}_{i,t}$ the opposite branch, $P_{c_{i,t}}$ the acting prompt, $P_{\bar{c}_{i,t}}$ the opposite prompt, $y_{i,t}$ the sampled response, and $G_{i,t}$ the return-to-go.

Reflection self-training objective. For a stored skill m, the pool-normalized reflection advantage is

$$A_m^{\mathrm{refl}} = \frac{U(m) - \mu_{\mathrm{pool}}}{\max(\sigma_{\mathrm{pool}}, \epsilon_{\mathrm{refl}})}, \tag{14}$$

where μ_{pool} and σ_{pool} are the mean and standard deviation of utilities in the corresponding task- or state-level pool, and ϵ_{refl} is a numerical floor. Given reflection prompt x_m and response z_m, the clipped reflection loss is

$$\mathcal{L}_{\mathrm{refl}} = \frac{1}{Z_{\mathrm{refl}}} \sum_{m,j} \ell^{\mathrm{refl}}_{m,j} \max\big(-\varrho^{\mathrm{refl}}_{m,j} A^{\mathrm{refl}}_m, -\mathrm{clip}(\varrho^{\mathrm{refl}}_{m,j}, 1-\epsilon_{\mathrm{clip}}, 1+\epsilon_{\mathrm{clip}}) A^{\mathrm{refl}}_m\big), \tag{15}$$

where $\ell^{\mathrm{refl}}_{m,j}$ masks valid reflection-response tokens, $\varrho^{\mathrm{refl}}_{m,j} = \pi_\theta(z_{m,j} \mid x_m, z_{m,<j}) / \pi_{\theta_{\mathrm{old}}}(z_{m,j} \mid x_m, z_{m,<j})$, and $Z_{\mathrm{refl}} = \sum_{m,j} \ell^{\mathrm{refl}}_{m,j}$.

Algorithm 1 summarizes the full UCOB training loop corresponding to this four-stage framework.

D Experimental Setup

Benchmarks and metrics. Following the evaluation protocol of SDAR (Lu et al., 2026a), we evaluate on ALFWorld (Shridhar et al., 2021), WebShop (Yao et al., 2022), and Search-QA (Jin et al., 2025). For ALFWorld, we report success rate on six task types: Pick, Look, Clean, Heat, Cool, and Pick2, together with the overall success rate across evaluation episodes. For WebShop, we report task score and success rate on the 128-task evaluation set. For Search-QA, we report answer accuracy on NQ, TriviaQA, PopQA, HotpotQA, 2Wiki, MuSiQue, and Bamboogle, where the first three are single-hop and the latter four are multi-hop search datasets.

Models and baselines. We use Qwen2.5-7B-Instruct and Qwen2.5-3B-Instruct (Qwen Team, 2024), together with Qwen3-1.7B (Yang et al., 2025), as the main backbones. The compared methods fall into three groups. **No-skill RL baselines** optimize the policy without retrieved skills, including GRPO and GiGPO (Feng et al., 2025). **Skill-augmented agent methods** retrieve, maintain, or evolve reusable skills during training, including SkillRL (Xia et al., 2026), D2Skill (Tu et al., 2026), SkillC (Lin et al., 2026), SAPO (Zhang et al., 2026e), and Skill1 (Shi et al., 2026). **Self-distillation methods** add auxiliary teacher-student supervision, including GRPO+OPSD (Zhao et al., 2026a), RLSD (Yang et al., 2026), Skill-SD (Wang et al., 2026), SDAR (Lu et al., 2026a), and StepOPSD (Zhang et al., 2026b). Together, these baselines cover the main comparison axes of UCOB: skill-free policy optimization, skill memory evolution, and self-distillation with fixed or adaptive supervision. Unless otherwise specified, UCOB uses the same task splits, rollout group size, and evaluation budget as the corresponding baselines. Implementation hyperparameters and dynamic-memory settings are provided in Appendix E.

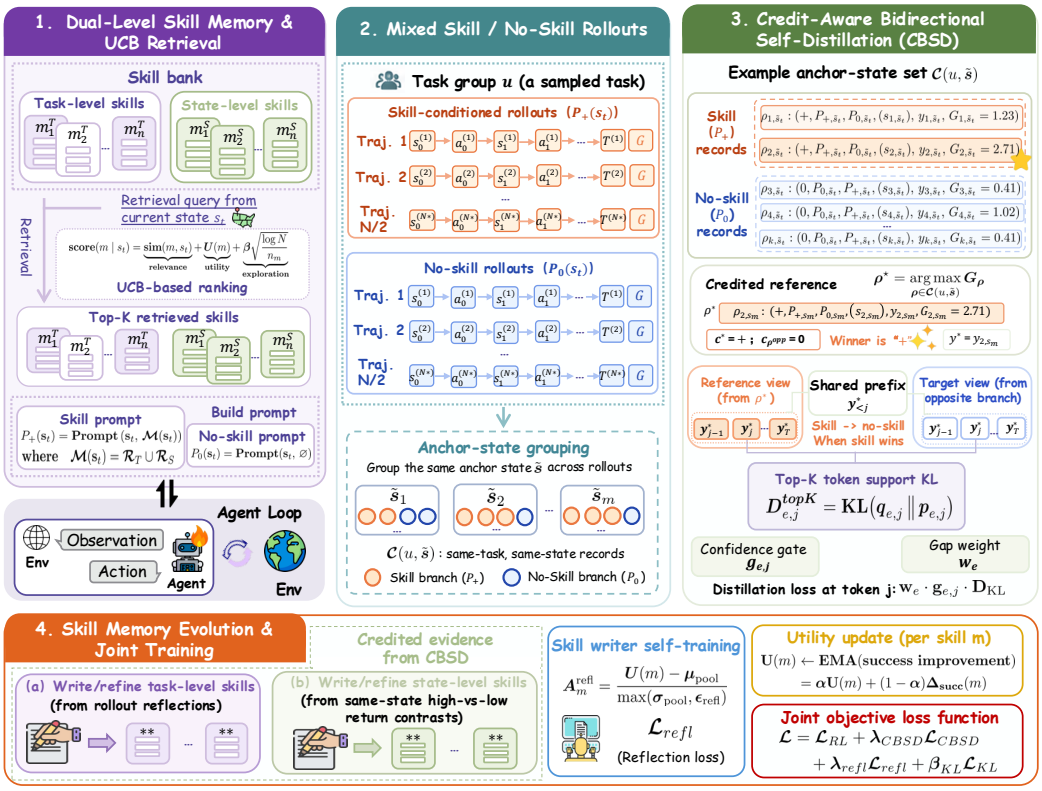


Figure 9: Detailed overview of UCOB as a **four-stage training loop**: (1) retrieve dual-granularity skills to build skill/no-skill views, (2) collect mixed rollouts and group anchor states, (3) route local supervision via credit-aware bidirectional self-distillation, and (4) jointly update the skill memory and policy. The updated memory and policy are reused in the next round.

E Implementation Details

Optimization and rollout. For ALFWorld and WebShop, UCOB uses GiGPO-style group advantage estimation with rollout group size 8, training batch size 16, validation batch size 128, and discount factor $\gamma = 0.95$. We train the actor with learning rate 10^{-6}, PPO mini-batch size 64, per-GPU micro-batch size 8, low-variance KL coefficient 0.01, entropy coefficient 0, and invalid-action penalty coefficient 0.1. The prompt and response length limits are 8192 and 512 tokens, respectively. Rollouts are generated with vLLM using top-$p = 1.0$ and maximum model length 10240; validation uses temperature 0.4 and top-$p = 1.0$. The maximum environment horizon is 50 for ALFWorld and 15 for WebShop. Main runs use 8 GPUs, train for 150 training steps, validate every 5 steps, and keep at most 5 checkpoints.

CBSD. CBSD is enabled on mixed skill/no-skill rollouts and uses the opposite context view as the distillation target selected by same-task, same-anchor-state returns. We use turn-level, response-scope top-K OPD with $K_{\text{tok}} = 32$, $\lambda_{\text{CBSD}} = 0.1$, return-gap margin $\epsilon_{\text{CBSD}} = 0.05$, gap temperature $\tau_{\text{CBSD}} = 0.2$, maximum gap weight $w_{\max} = 2.0$, and at most two accepted teacher-target pairs per anchor-state group. The confidence gate is enabled with coefficient 5.0. CBSD supervision is applied on the action span, with action-token weight 2.0.

Dynamic memory and reflection. The dual-granularity skill memory maintains separate task-level and state-level pools and retrieves from both pools during training and evaluation. For each prompt, we retrieve top-3 task skills and top-3 state skills after top-10 sentence-transformer candidate filtering, using state-aware task/state retrieval queries and UCB scoring with exploration scale 0.1. The relevance threshold is 0.4, the context budget for retrieved skills is 4096 tokens, and skill utilities are updated with the baseline-relative rule using EMA rate $\beta_U = 0.2$ for both pools. Reflec-

Algorithm 1. UCOB: Credit-Aware On-Policy Bidirectional Self-Distillation

Require: policy π_θ, old policy $\pi_{\theta_{\text{old}}}$, reference policy π_{ref}, task distribution $\mathcal{D}$, dual-granularity skill memory $\mathcal{M} = \{\mathcal{M}_{\text{task}}, \mathcal{M}_{\text{state}}\}$, rollout group size N, retrieval budgets K_p^{mem}, CBSD margin ϵ_{CBSD}, loss weights $(\lambda_{\text{CBSD}}, \lambda_{\text{refl}}, \beta_{\text{KL}})$
Ensure: trained policy π_θ, task-level skill pool $\mathcal{M}_{\text{task}}$, and state-level skill pool $\mathcal{M}_{\text{state}}$

Initialize or load π_θ, π_{ref}, $\mathcal{M}_{\text{task}}$, and $\mathcal{M}_{\text{state}}$
for training step $r = 1, \ldots, T$ **do**
  Sample a batch of task groups $\{u\} \sim \mathcal{D}$
  for each task group u **do**
    Step 1: skill retrieval and view construction.
    for each encountered state s_t **do**
      Retrieve task/state skills $M_{\text{task}}(s_t), M_{\text{state}}(s_t)$ with Eqs. (11-12)
      Build the skill-conditioned view $P_+(s_t)$ by inserting retrieved skills, and the no-skill view $P_0(s_t)$ without retrieved skills
    end for
    Step 2: mixed on-policy rollouts and anchor-state grouping.
    Roll out $N/2$ trajectories under P_+ and $N/2$ trajectories under P_0 with the current policy
    Store each decision record $\rho_{i,t}$ with branch $c_{i,t} \in \{+, 0\}$, acting/opposite prompts, response $y_{i,t}$, and return-to-go $G_{i,t}$
    Group records by same task and canonical anchor state into $\mathcal{C}(u, \tilde{s})$ as in Eq. (1)
  end for
  Step 3 (core): credit-aware bidirectional self-distillation.
  Initialize the accepted pair set $\mathcal{P}_{\text{CBSD}} \leftarrow \emptyset$
  for each anchor-state group $\mathcal{C}(u, \tilde{s})$ **do**
    Select the highest-return local record $\rho^\star$, its branch $c^\star$, and response $y^\star$ by Eq. (2)
    for each opposite-view record $\rho^{\text{opp}} \in \mathcal{C}(u, \tilde{s})$ with $c_{\rho^{\text{opp}}} = \bar{c}^\star$ **do**
      Compute $\Delta_e = G_{\rho^\star} - G_{\rho^{\text{opp}}}$
      if $\Delta_e \le \epsilon_{\text{CBSD}}$ **then**
        Ignore this pair because the local return evidence is weak
      else if $c^\star = +$ **then**
        Internalize useful skill behavior: add a pair e that distills $P_+ \to P_0$ with weight from Δ_e
      else
        Correct misleading skill context: add a pair e that distills $P_0 \to P_+$ with weight from Δ_e
      end if
    end for
  end for
  Compute token-level top-K targets, gap weights, confidence gates, and $\mathcal{L}_{\text{CBSD}}$ as in Eq. (5)
  Step 4: dynamic skill evolution and joint optimization.
  Write/refine task-level skills from rollout-group reflection, and state-level skills from high-vs-low anchor-state contrasts
  Maintain skill memory: deduplicate, update utility by Eq. (6), and prune weak/redundant skills
  Self-train the reflection writer with utility-derived advantages using Eq. (15)
  Update π_θ with the joint objective in Eq. (7)
end for
return $\pi_\theta, \mathcal{M}_{\text{task}}, \mathcal{M}_{\text{state}}$

tion writes one task-level item per rollout group and one state-level item per qualifying anchor-state contrast, with minimum local gap 0.05, maximum reflection response length 2048, temperature 1.0, and top-$p = 1.0$. Reflection self-training starts after a 30-step warmup in the Qwen3-1.7B runs, samples at most 16 task-level and 16 state-level memories per step, keeps memories with absolute utility at least 0.02, computes pool-normalized advantages with clipping range $[-2, 2]$, and uses loss weight $\lambda_{\text{refl}} = 1.0$.

SEARCH-QA runs. For SEARCH-QA, the search-agent scripts use rollout group size 8, training batch size 128, validation batch size 512, prompt and response limits $4096/512$, maximum search horizon 4, and history length 4. The actor learning rate is 10^{-6} with warmup ratio 0.1, PPO mini-batch size 256, per-GPU micro-batch size 16, KL coefficient 0.001, and invalid-action penalty coefficient 0.01. Search rollouts call the local retrieval server and are trained for 150 steps on 8 GPUs, with validation at the final training step.

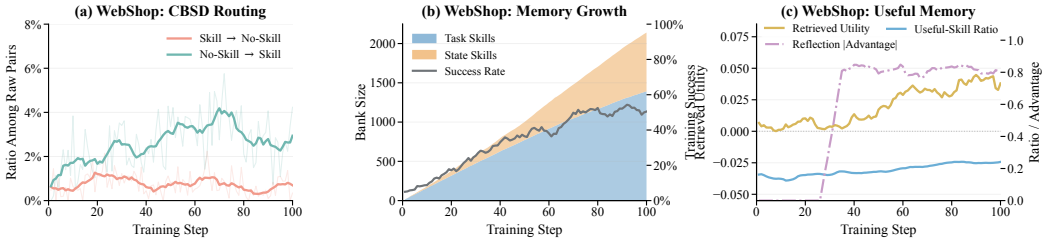


Figure 10: Mechanism and memory-evolution analysis of UCOB on WEBSHOP with QWEN3-1.7B, corresponding to Figure 5.

F ADDITIONAL ANALYSES

F.1 WEBSHOP MECHANISM AND MEMORY EVOLUTION

Figure 10 shows the same qualitative pattern on WEBSHOP: no-skill-to-skill routing remains more frequent, and both skill pools grow over training. Retrieved utility remains positive later in training, while the useful-skill ratio and reflection signal stay active. These results provide a cross-environment counterpart to the ALFWORLD analysis in the main text.

F.2 LOCALIZED TRAINING COST

Figure 11 evaluates the wall-clock cost effectiveness of UCOB. All methods use the same **8 NVIDIA A800 GPUs** and QWEN3-1.7B backbone, and we measure *training* time only, excluding validation and evaluation overhead. Under this protocol, UCOB reaches higher validation success within a comparable wall-clock budget on both ALFWORLD and WEBSHOP. The per-step breakdown further shows that policy rollout remains the dominant cost, while the additional CBSD-related computation is localized mainly to old-policy log-prob computation, reference-logit computation, and the policy update. **UCOB's extra training overhead is modest and localized.**

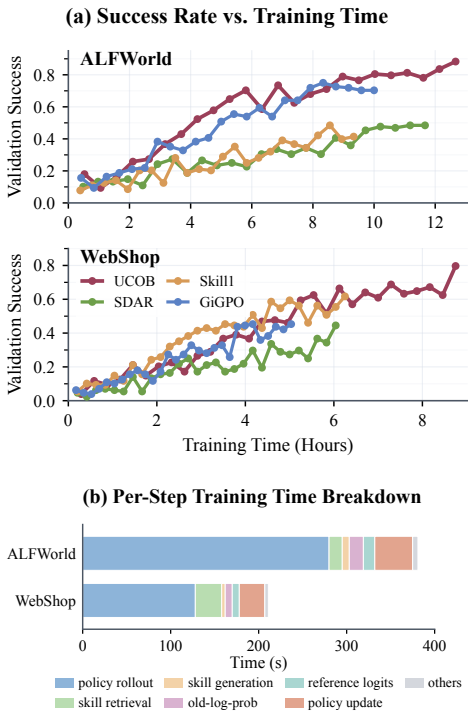


Figure 11: Cost analysis for UCOB.

F.3 CONTINUAL ADAPTATION ACROSS ENVIRONMENTS

Figure 12 tests whether UCOB remains effective beyond isolated single-environment training. We sequentially train each QWEN3-1.7B method on ALFWORLD, WEBSHOP, and SEARCH-QA, with 150 updates per environment and evaluation on all three throughout the stream. All RL methods use the same per-stage update budget. As a supervised control, SFT uses 4,800 examples per environment collected by an RL-trained QWEN3-1.7B, with 32 examples per update to match the 150-step budget. Recent SDFT results show that on-policy self-distillation can reduce forgetting during sequential learning from demonstrations (Shenfeld et al., 2026). Although our setting instead uses reward-driven agentic RL, we observe a related pattern: after the two subsequent stages, SDAR and UCOB retain 67.1% and 82.8% on ALFWORLD, only 2.7 and 4.7 points below their respective stage-end results. UCOB also preserves WEBSHOP performance after SEARCH-QA training (75.8% → 77.3%) and reaches 45.2% on SEARCH-QA. Its final three-environment average

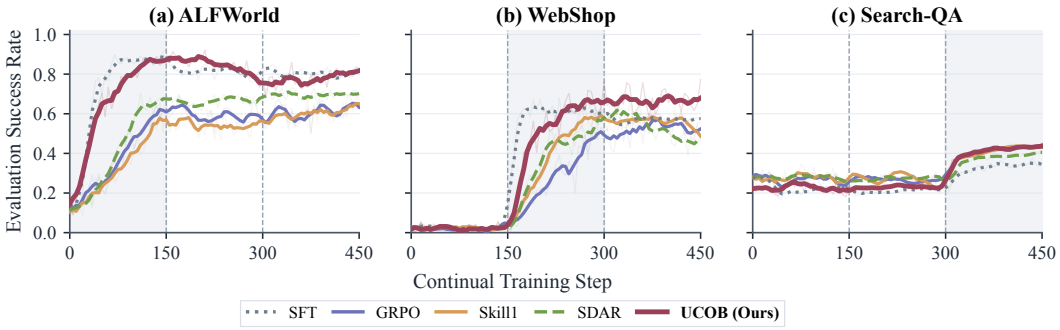


Figure 12: Continual adaptation under the ALFWORLD→WEBSHOP→SEARCH-QA stream with QWEN3-1.7B. Each environment is trained for 150 steps. Dashed lines mark task transitions, and gray shading marks the active training interval in each panel.

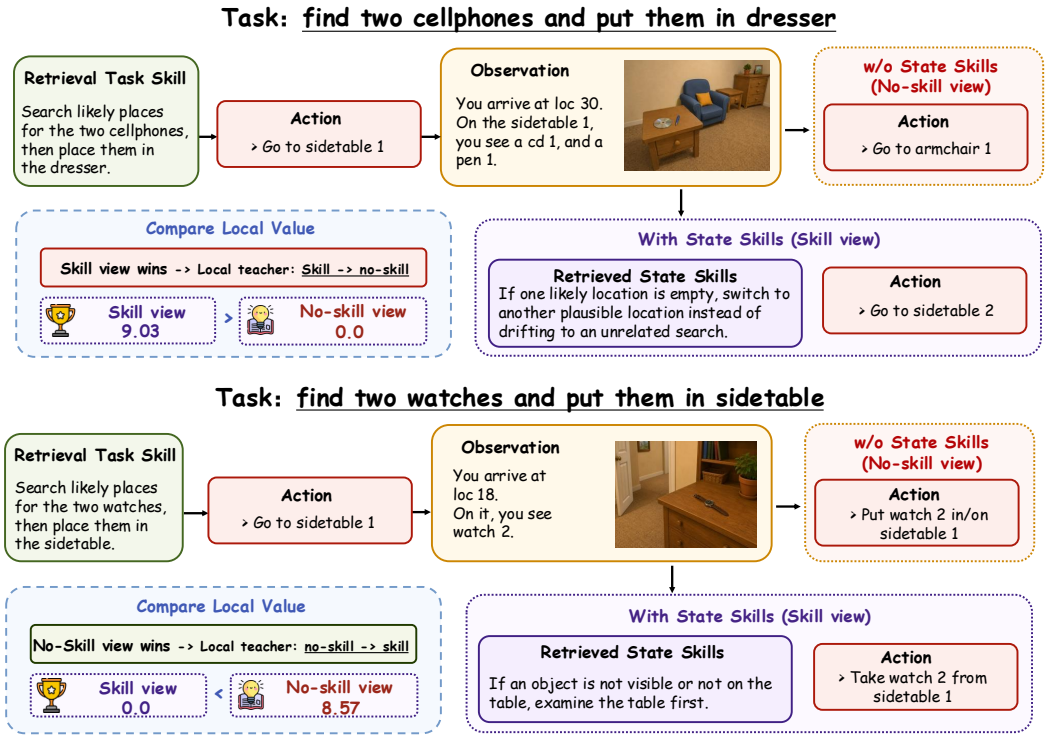


Figure 13: Case study of local teacher selection in UCOB.

is 68.4%, compared with 57.0% for SFT and 51.8% for SDAR. **Thus, on-policy self-distillation provides useful continual-learning behavior, while CBSD and evolving skill memory yield a stronger retention–adaptation balance.**

F.4 CASE STUDY

Figure 13 shows two ALFWORLD cases that make the bidirectional behavior concrete. In the cellphone-to-dresser task, the no-skill view drifts to an unrelated search location, while retrieved state skills guide the skill-conditioned view to another plausible table; UCOB therefore selects the skill view and distills it into the no-skill view. Conversely, in the two-watch-to-sidetable task, the retrieved state skill triggers an unnecessary attempt to take a watch that is already visible on the sidetable, while the no-skill view proceeds to place it; UCOB selects the no-skill view to correct

the misleading skill context. These cases show that skills are neither always trusted nor always discarded: **the teacher direction is decided by local return evidence.**

G Extended Related Work

Language agents with memory and skills. Retrieval-augmented generation conditions language models on external information (Lewis et al., 2020), and agent systems extend this idea with reasoning traces, verbal feedback, long-term memories, and reusable skills (Yao et al., 2023; Shinn et al., 2023; Zhao et al., 2024; Wang et al., 2024; Zhang et al., 2025). Recent agent methods further train agents to build, curate, select, or internalize skill memories, including recursive skill-memory systems, dual-granularity skill pools, learned skill curation, evolving skill repositories, and explicit skill internalization (Xia et al., 2026; Mi et al., 2026; Tu et al., 2026; Ouyang et al., 2026; Zhang et al., 2026a; Shi et al., 2026; Vishe et al., 2026; Lu et al., 2026b; He et al., 2026; Lin et al., 2026; Zhang et al., 2026e; Chen et al., 2026). These works differ in whether skills are treated mainly as retrieval-time context, evolving external memory, or behavior to be internalized by the policy. UCOB addresses a complementary question: when retrieved skills are imperfect, training should decide whether to internalize the skill-conditioned behavior or correct it with evidence from the no-skill view, while evolving the skill memory itself.

Agentic RL and credit assignment. PPO and its variants remain common RL objectives for language agent training (Schulman et al., 2017). For multiturn environments, sparse terminal rewards make trajectory-level updates inefficient, motivating agentic RL frameworks with finer-grained credit assignment, interactive guidance, or disentangled optimization (Wang et al., 2025; Feng et al., 2025; Wei et al., 2025; Zhu et al., 2026; Li et al., 2026a; Zhang et al., 2026b; Li et al., 2026c; Liu et al., 2026). UCOB uses credit evidence differently: it compares skill-conditioned and no-skill rollouts within the same task and anchor state, and converts the local return gap into a distillation direction between context views. Thus, credit assignment determines not only which action tokens to reinforce, but also which context view should serve as the local teacher.

On-policy self-distillation with privileged context. Knowledge distillation and policy distillation transfer behavior from a teacher distribution to a student policy (Hinton et al., 2015; Rusu et al., 2016). On-policy self-distillation adapts this idea to student-generated rollouts, providing dense token-level supervision alongside RL and motivating recent work on signal filtering, reversed teachers, and shorter on-policy prefixes (Zhao et al., 2026a; Yang et al., 2026; Song & Zheng, 2026; Fu et al., 2026; Li et al., 2026b; Zhao et al., 2026b; Kim et al., 2026; Zhang et al., 2026c). Privileged self-distillation has also been used for multiagent self-play without external data (Zhang et al., 2026d), while Skill-SD and SDAR instantiate a privileged teacher by conditioning on retrieved skills and distilling from the skill-conditioned prompt into the no-skill prompt (Wang et al., 2026; Lu et al., 2026a). However, this fixed-teacher assumption fails when skill retrieval or utilization is locally harmful. UCOB instead treats the skill-conditioned and no-skill prompts as two on-policy views of the same model and selects the teacher direction from same-anchor-state returns. It complements prior OPD-style filtering by making teacher selection itself credit-aware, internalizing useful skills and correcting misleading skill context within one bidirectional objective.

H Theoretical Perspective

This appendix analyzes CBSD as a local policy-improvement operator rather than a global convergence guarantee. The full UCOB system is nonstationary because the policy, retrieval distribution, and skill memory co-evolve during training. Accordingly, we fix one policy update, one task-anchor group, and the old on-policy continuation, and ask what the CBSD direction means locally. The argument follows the performance-difference and trust-region view of policy improvement (Kakade & Langford, 2002; Schulman et al., 2015), and interprets KL distillation as a small mirror-descent or advantage-weighted supervised step (Tomar et al., 2022; Peng et al., 2019).

Let $x = (u, \tilde{s})$ denote a same-task, same-anchor-state context. For clarity, write $p_c(y \mid x)$ for the response-level distribution induced by branch $c \in \{+, 0\}$ under the old policy, where $+$ denotes the skill-conditioned view and 0 denotes the no-skill view. After response y is produced, the continua-

tion follows the old policy. Define

$$Q^\pi(x,y) = \mathbb{E}[G_t \mid u, \mathrm{anchor}(s_t) = \tilde{s},\, y_t = y]\,, \qquad \mu_c(x) = \mathbb{E}_{y\sim p_c(\cdot|x)} Q^\pi(x,y), \tag{16}$$

and let $A^\pi(x,y) = Q^\pi(x,y) - V^\pi(x)$. The branch-level value gap is $\Delta(x) = \mu_+(x) - \mu_0(x)$. The implemented objective uses sample-level accepted gaps $\Delta_e = G_{\rho^\star} - G_{\rho^{\mathrm{opp}}}$ with weight $w_e = \mathrm{clip}(\Delta_e/\tau_{\mathrm{CBSD}}, 0, w_{\max})$. Let $J(\pi)$ be the expected discounted return from the initial task distribution, $J^\star = \sup_{\pi'} J(\pi')$, and $d_\pi(x) = (1-\gamma)\sum_{t\geq 0}\gamma^t \Pr_\pi(x_t = x)$ be the normalized discounted occupancy of the augmented task-anchor context.

We first isolate the same-anchor credit identity, then use it to prove the main proposition and connect the result to the implemented token-level objective.

Proposition 1 (Same-anchor returns estimate local advantage differences)**.** *For a fixed task-anchor context $x = (u, \tilde{s})$ and fixed old-policy continuation,*

$$\Delta(x) = \mathbb{E}_{y\sim p_+(\cdot|x)} A^\pi(x,y) - \mathbb{E}_{y\sim p_0(\cdot|x)} A^\pi(x,y). \tag{17}$$

Thus, within the same anchor state, the skill-minus-no-skill return gap is a local advantage-difference signal rather than a raw trajectory-level preference.

Proof. By definition, $\mu_c(x) = \mathbb{E}_{y\sim p_c(\cdot|x)} Q^\pi(x,y)$ for each branch c. Since the two branches are compared at the same x, the baseline $V^\pi(x)$ is shared:

$$\mathbb{E}_{p_+} A^\pi(x,y) - \mathbb{E}_{p_0} A^\pi(x,y) = \big(\mathbb{E}_{p_+} Q^\pi(x,y) - V^\pi(x)\big) - \big(\mathbb{E}_{p_0} Q^\pi(x,y) - V^\pi(x)\big) \tag{18}$$

$$= \mu_+(x) - \mu_0(x) = \Delta(x). \tag{19}$$

Therefore, grouping by the same anchor state removes the state-value baseline and makes the return gap a local credit signal for choosing the teacher direction. □

Proposition 2 (First-order local improvement direction of CBSD)**.** *Let π use the lower-value branch p_ℓ at x and remain fixed elsewhere, where $\delta_x = \mu_h(x) - \mu_\ell(x) > 0$. For $\eta \in (0,1]$, let π^η replace only this response distribution by $p_\ell^\eta = (1-\eta)p_\ell + \eta p_h$. The local advantage surrogate then increases by $\eta\delta_x$. Assume that the trust-region implementation has an occupancy-shift remainder $R_\eta \geq -C\eta^2$ for some $C \geq 0$ in the performance-difference decomposition. Then*

$$J(\pi^\eta) - J(\pi) \geq \frac{d_\pi(x)}{1-\gamma}\eta\delta_x - C\eta^2. \tag{20}$$

Proof. Let h and ℓ denote the higher- and lower-value branches, so $\delta_x = \mu_h(x) - \mu_\ell(x) > 0$. Because π uses p_ℓ at x, $V^\pi(x) = \mu_\ell(x)$. For the idealized interpolation $p_\ell^\eta = (1-\eta)p_\ell + \eta p_h$, linearity and Proposition 1 give

$$\mathbb{E}_{p_\ell^\eta} A^\pi(x,y) - \mathbb{E}_{p_\ell} A^\pi(x,y) = \eta\,(\mathbb{E}_{p_h} A^\pi(x,y) - \mathbb{E}_{p_\ell} A^\pi(x,y)) \tag{21}$$

$$= \eta\,(\mu_h(x) - \mu_\ell(x)) = \eta\delta_x, \tag{22}$$

where the second equality uses the shared baseline in Proposition 1. The performance-difference lemma decomposes the full-policy change into the old-occupancy contribution and an occupancy-shift remainder:

$$J(\pi^\eta) - J(\pi) = \frac{d_\pi(x)}{1-\gamma}\eta\delta_x + R_\eta. \tag{23}$$

Only the response distribution at x changes, so the first term is the old-occupancy contribution above; the trust-region assumption bounds the remaining occupancy shift by $R_\eta \geq -C\eta^2$, yielding the stated lower bound. When this lower bound is positive,

$$J^\star - J(\pi^\eta) = J^\star - J(\pi) - (J(\pi^\eta) - J(\pi)) < J^\star - J(\pi), \tag{24}$$

so the value gap to the optimal policy decreases. □

Connection to the implemented objective. The implemented token-level CBSD objective can be viewed as a KL-proximal stochastic approximation to the above interpolation. For an accepted pair e and token position j, the reference distribution $q_{e,j}$ is stop-gradient from the higher-return branch, while $p_{e,j}$ is the target-branch distribution. Treating the reference distribution and pair weights $\chi_{e,j}$ as fixed during differentiation and omitting the normalization constant,

$$-\nabla_\theta(\chi_{e,j}\mathrm{KL}(q_{e,j}\|p_{e,j})) = \chi_{e,j}\,\mathbb{E}_{v\sim q_{e,j}}[\nabla_\theta \log p_{e,j}(v)]\,, \qquad \chi_{e,j} = m_{e,j}w_e g_{e,j} \geq 0. \tag{25}$$

Thus, CBSD applies positive-weight imitation toward the locally higher-return branch, with the weight increasing in the accepted return gap.

Proposition 3 (Bidirectionality avoids negative fixed-teacher updates)**.** *If a fixed skill-to-no-skill rule is used, then for any context with $\Delta(x) < 0$, the induced local surrogate change on the no-skill branch is negative:*

$$\mathbb{E}_{(1-\eta)p_0+\eta p_+}Q^\pi(x,y) - \mathbb{E}_{p_0}Q^\pi(x,y) = \eta\Delta(x) < 0. \tag{26}$$

CBSD reverses the direction in this case and instead obtains a positive local change of $\eta(-\Delta(x))$ on the skill-conditioned branch.

Proof. For a fixed skill-to-no-skill update, the target distribution becomes $(1-\eta)p_0 + \eta p_+$. By linearity of expectation, its local value change relative to p_0 is

$$\eta\,(\mu_+(x) - \mu_0(x)) = \eta\Delta(x). \tag{27}$$

When $\Delta(x) < 0$, this is a negative-advantage update. CBSD selects the no-skill branch as the teacher and updates the skill-conditioned branch toward p_0, giving

$$\eta\,(\mu_0(x) - \mu_+(x)) = \eta(-\Delta(x)) > 0. \tag{28}$$

The same argument shows that when $\Delta(x) > 0$, the skill-conditioned branch is the improving teacher. Therefore, bidirectional teacher selection is necessary to turn mixed-sign local gaps into signed policy-improvement directions. □

Corollary 1 (Conditional contraction toward a local optimal policy)**.** *Let $\pi_x^\star$ be a local optimal response distribution at context x, and assume common support. If the CBSD-selected higher-return branch is also closer to $\pi_x^\star$ than the lower-value branch by margin $\zeta_x > 0$,*

$$D_{\mathrm{KL}}(\pi_x^\star\|p_h) \leq D_{\mathrm{KL}}(\pi_x^\star\|p_\ell) - \zeta_x, \tag{29}$$

then the idealized CBSD update satisfies

$$D_{\mathrm{KL}}(\pi_x^\star\|p_\ell^\eta) \leq D_{\mathrm{KL}}(\pi_x^\star\|p_\ell) - \eta\zeta_x. \tag{30}$$

Proof. The KL divergence $D_{\mathrm{KL}}(\pi_x^\star\|\cdot)$ is convex in its second argument. Thus,

$$\begin{aligned} D_{\mathrm{KL}}(\pi_x^\star\|p_\ell^\eta) &= D_{\mathrm{KL}}(\pi_x^\star\|(1-\eta)p_\ell + \eta p_h) && (31)\\ &\leq (1-\eta)D_{\mathrm{KL}}(\pi_x^\star\|p_\ell) + \eta D_{\mathrm{KL}}(\pi_x^\star\|p_h) && (32)\\ &\leq D_{\mathrm{KL}}(\pi_x^\star\|p_\ell) - \eta\zeta_x. && (33)\end{aligned}$$

This corollary is conditional: return gaps alone do not imply distributional closeness to $\pi_x^\star$. It formalizes the sense in which CBSD can move the target view closer to a local optimal policy when the higher-return branch is also a better local reference. □