SKILLC: Learning Autonomous Skill Internalization in LLM Agents via Contrastive Credit Assignment

Anonymous ACL submission

Abstract

Structured skill prompts improve exploration in long-horizon agentic reinforcement learning (RL). Skill-augmented RL methods retain external skills at inference, while skill-internalization RL methods withdraw them during training to enable autonomous performance. However, existing internalization approaches only use skill-helpfulness contrast for curriculum control, leaving the policy update unchanged and unable to distinguish skill-dependent from autonomous success. We propose SKILLC, a framework based on Contrastive Skill Credit Assignment (CSCA) that converts this contrast into a direct learning signal for internalization. SKILLC samples paired skill-injected and skill-free rollouts for tasks from active skill types within the same policy update, and injects their task-level contrast into optimization via a dual-stream advantage estimator that preserves global ranking while applying a one-sided correction toward skill-free success. A smoothed validation-level signal further drives an adaptive curriculum over attribution strength, rollout allocation, and monotonic active-set pruning. Experiments on ALFWorld and WebShop show that, without runtime skill access, SKILLC surpasses the strongest prior skill-internalization RL baseline by 5.5% and 4.4%, respectively, while remaining competitive with skill-augmented RL methods.

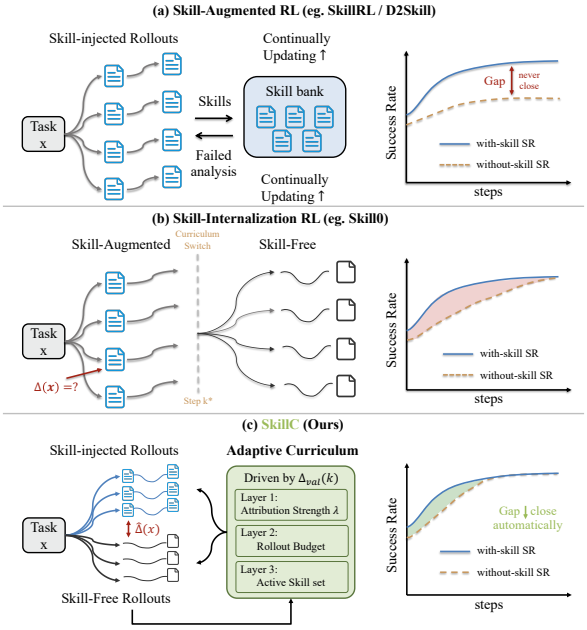

Figure 1: Comparison of paradigms for skill use in Agentic RL. (a) Skill-augmented RL methods keep skills available at inference time and optimize performance with runtime skill support. (b) Skill-internalization RL methods withdraw skills during training and may estimate skill helpfulness for control, but leave task-level credit assignment unchanged. (c) SKILLC turns helpfulness comparison into contrastive credit assignment for autonomous success, while adapting attribution strength, rollout allocation, and the active skill set through a validation-driven curriculum.

1 Introduction

Structured skill prompts (Xu and Yan, 2026; Zhang et al., 2026a; Liu et al., 2026b) have become a practical mechanism for training long-horizon LLM agents. By providing reusable procedural guidance during interaction, they improve exploration and sample efficiency in agentic reinforcement learning (RL) (Li et al., 2026b,c; Su et al., 2026; Cho et al., 2026). Recent work has advanced this paradigm through skill augmentation and co-evolving skill banks (Li et al., 2026c), skill distillation (Wang et al., 2026b; Xia et al., 2026), and skill lifecycle management (Pu et al., 2026; Ouyang et al., 2026; Shen et al., 2026). Yet most of these approaches remain in the skill-augmented regime shown in Figure 1(a): skills are treated as external runtime support, and autonomous performance without retrieval is not the optimization target.

A more ambitious goal is skill internalization, illustrated in Figure 1(b), where skills are used only during training and gradually withdrawn so that the final policy can act autonomously. SKILL0 (Lu

et al., 2026b) is a representative example. Current internalization methods can already estimate whether a skill remains useful and use that information to control withdrawal. What remains unchanged, however, is the policy objective. Within each GRPO update (Shao et al., 2024), all rollouts for a task are still generated under a single prompt condition, so optimization reinforces success under the current regime rather than autonomy after withdrawal. The training loop may know that a skill is still globally useful, yet it cannot determine whether a particular successful trajectory is already independent of that skill.

We term this optimization mismatch *internalization blindness*. To address it, we propose SKILLC, a framework based on Contrastive Skill Credit Assignment (CSCA) that converts this contrast into a direct learning signal for internalization. For tasks from active skill types, SKILLC samples paired skill-injected and skill-free rollouts within the same policy update, injecting their task-level contrast into optimization via a dual-stream advantage estimator that preserves global ranking while applying one-sided correction toward skill-free success. A smoothed validation-level counterpart of the same contrastive quantity further drives a hybrid adaptive curriculum. For multi-category environments like ALFWorld, this curriculum jointly controls attribution strength, rollout allocation, and monotone active-set pruning; for single-skill environments like WebShop, it controls attribution strength and rollout allocation uniformly. Together, these mechanisms yield the contrastive internalization regime illustrated in Figure 1(c).

Our main contributions are as follows:

- We identify trajectory-level credit assignment limitations in fixed-schedule skill internalization where all rollouts share a single prompt condition, preventing the gradient from distinguishing autonomous versus skill-dependent success. We propose SKILLC with paired contrastive rollouts and adaptive curriculum scheduling to address this.

- We implement CSCA in SKILLC through three coordinated mechanisms. Paired skill-injected and skill-free rollouts expose task-level internalization gaps, a dual-stream advantage estimator with per-condition normalization redirects credit toward autonomous success, and a validation-driven curriculum adapts attribution strength and rollout allocation.

- Critical characterization of applicability reveals CSCA's effectiveness is signal-quality-dependent. Strong contrastive signals enable substantial gains (Heat $+10.5\%$, Cool $+11.1\%$, Pick2 $+33.4\%$), while weak signals on ceiling-effect tasks cause regression (Pick -11.5%), and single-skill environments yield only marginal improvements (WebShop $+3.1\%$).

2 Related Work

2.1 Agentic Reinforcement Learning

Reinforcement learning has become a key post-training paradigm for language model agents. GRPO (Shao et al., 2024) introduced critic-free policy optimization using group-relative advantages. Building on this foundation, agentic RL methods address agent-specific challenges through hierarchical RL for long-horizon credit assignment (Li et al., 2026c), self-evolution mechanisms for policy co-adaptation (Fan et al., 2026; Shi et al., 2026), experience reuse for continuous adaptation (Chhikara et al., 2025; Liu et al., 2026a), and prompt-based reasoning like ReAct (Yao et al., 2022b) and Reflexion (Shinn et al., 2023). However, a key challenge remains: how to leverage intermediate guidance during training while ensuring the final policy acquires genuine autonomous competence rather than guidance-dependent behaviors.

2.2 Skill Libraries for Agents

Structured skills have become a core mechanism for extending agent capabilities beyond parametric memory. Early work recognized that raw trajectory retrieval incurs substantial token overhead and noise, motivating abstraction into reusable procedural knowledge (Xia et al., 2026; Lu et al., 2026b). Recent skill-based systems span multiple dimensions: hierarchical skill organization and discovery through reflection (Tu et al., 2026; Wang et al., 2026a), dependency-aware retrieval via graph structures (Li et al., 2026a), and dynamic lifecycle management including retention and retirement (Shen et al., 2026; Pu et al., 2026). For skill selection, approaches range from frozen embedding-based retrieval to learned routing and policy-driven ranking (Xia et al., 2026; Cho et al., 2026). For utilization, RL-based methods condition policies on selected skills with hierarchical reward structures (Tu et al.,

2026; Shi et al., 2026). For distillation, methods span prompt-based extraction and self-reflection to co-evolving skill generators (Xia et al., 2026; Wang et al., 2026b). However, existing skill-augmented RL methods do not jointly optimize all three stages with unified learning signals, and existing skill internalization approaches progressively withdraw skills according to curriculum schedules but leave the policy update itself unchanged during training. Within each gradient step, all rollouts share the same prompt condition, preventing the gradient from observing which behaviors are already autonomous versus skill-dependent. This creates an optimization mismatch we term internalization blindness.

3 Method

We propose SKILLC, addressing a key limitation in skill internalization: within each policy update, standard GRPO jointly normalizes skill-injected and skill-free trajectories, inflating the reward baseline when skill-injected rollouts dominate and suppressing credit for autonomous success. SKILLC resolves this through three mechanisms: (1) sampling paired skill-injected and skill-free rollouts to measure task-level internalization gaps, (2) using per-condition advantage normalization to redirect credit toward autonomous behaviors, and (3) adapting the curriculum via validation-level contrastive signals to balance attribution strength, rollout allocation, and skill retention.

3.1 Problem Formulation

For a task $x \sim \mathcal{X}$, we maintain a fixed skill bank $\mathcal{B} = \{\mathcal{S}_k\}_k$ and an active task-type set $\mathcal{K}^{(t)}_{\text{active}}$. Let $z \in \{0, 1\}$ denote the prompt condition: $z = 1$ for skill-injected rollouts and $z = 0$ for skill-free. For active task types $k(x) \in \mathcal{K}^{(t)}_{\text{active}}$, we sample paired rollouts $\tau^+ \sim \pi_\theta(\cdot \mid x, \mathcal{S}_{k(x)})$ and $\tau^- \sim \pi_\theta(\cdot \mid x)$. For retired types, only τ^- is sampled. Both use the same policy parameters, yielding self-contrastive design.

Task-Level Contrastive Gap. Unlike skill-augmented methods, SKILLC targets without-skill success at inference. The task-level contrastive gap is

$$\Delta(x) = \mathbb{E}[R \mid x, z = 1; \pi_\theta] - \mathbb{E}[R \mid x, z = 0; \pi_\theta], \tag{1}$$

where $z \in \{0, 1\}$ indicates skill injection. Positive $\Delta(x)$ indicates skill dependence; $\Delta(x) \approx 0$ indicates internalization. Standard GRPO cannot observe this signal because all rollouts share one prompt condition. SKILLC exposes it through paired rollouts.

3.2 Paired Contrastive Rollouts

SKILLC enforces paired rollouts for every active task type. At each step t, the curriculum gate value $\text{gate}^{(t)}(k)$ is converted into a skill-injected rollout fraction:

$$\rho^{(t)}_{\text{with}}(k) = \rho_{\min} + (1 - 2\rho_{\min})[2\text{gate}^{(t)}(k) - 1]_+. \tag{2}$$

This mapping reduces skill injection from dominant early-training levels to a small probing budget $\rho_{\min}$ as internalization progresses.

For each active task x, we sample G rollouts and assign $n^{(t)}_+(x) = \text{clip}(\text{round}(G\rho^{(t)}_{\text{with}}(k(x))), 1, G - 1)$ to the skill-injected branch, with the remainder to skill-free. Let $\mu^+(x)$ and $\mu^-(x)$ denote the mean rewards of the two branches. The batch-level contrastive signal is

$$\hat{\Delta}(x) = \text{clip}\Big(\text{EMA}_{\alpha_{\text{batch}}}\big(\mu^+(x) - \mu^-(x)\big), -\delta, \delta\Big), \tag{3}$$

where δ is a clipping bound. Unlike utility-based skill scoring or reward shaping, this contrastive signal is used exclusively for credit assignment and curriculum control; it is never converted into skill bonuses or trajectory rewards. For retired task types, $\hat{\Delta}(x) = 0$ by convention.

3.3 Dual-Stream Advantage Estimation

SKILLC injects this signal via a dual-stream advantage estimator. When skills are beneficial, skill-injected trajectories achieve higher rewards. Standard joint normalization inflates the baseline, suppressing advantages for successful skill-free trajectories precisely when internalization is most critical. To address this, SKILLC decomposes the advantage into two complementary streams:

$$\hat{A}_{\text{CSCA}}(i, x) = \Phi(R_i, \mathcal{R}_{\text{batch}}) + \omega \cdot A^{\text{cond}}_{\text{contra}}(i, x), \tag{4}$$

where Stream 1 preserves global ranking via $\Phi(R_i, \mathcal{R}_{\text{batch}}) = (R_i - \mu_{\text{ep}})/(\sigma_{\text{ep}} + \epsilon)$, and the correction magnitude is

$$C(x) = \lambda_{\text{eff}}(x)[\hat{\Delta}(x)]_+, \tag{5}$$

where $\lambda_{\text{eff}}(x) = \lambda^{(t)}_{\text{dyn}}(k(x))/(1 + \beta_{\text{freq}} n_x)$ scales the signal by internalization progress and task frequency.

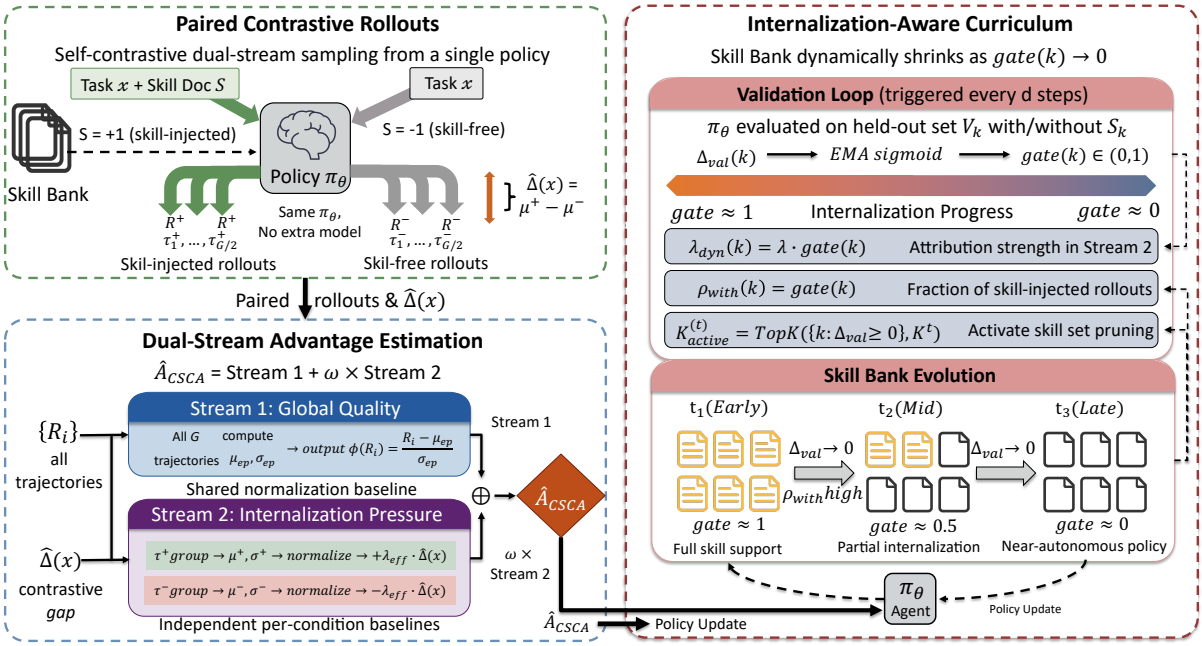

Figure 2: Overview of SKILLC. Paired contrastive rollouts expose residual skill dependence for each task, a dual-stream advantage estimator redirects credit toward autonomous success without mixed-normalization bias, and a internalization-aware curriculum adapts attribution strength, rollout allocation, and active skill set.

Stream 2 performs condition-wise normalization and reallocates credit according to the contrastive signal:

$$A^{\text{cond}}_{\text{contra}}(i,x) = \begin{cases} \dfrac{R_i - \mu^+(x)}{\sigma^+(x)+\epsilon} - C(x), & z_i = 1, \\ \dfrac{R_i - \mu^-(x)}{\sigma^-(x)+\epsilon} + C(x), & z_i = 0. \end{cases} \tag{6}$$

When $\hat{\Delta}(x) > 0$ (skill is still helpful), Stream 2 downweights skill-dependent trajectories (subtracting $C(x)$) while upweighting skill-free successes (adding $C(x)$), encoding the learning objective: "succeed without skill support." This signal is transient by design: as internalization progresses and $\hat{\Delta}(x)$ narrows, $C(x) \to 0$ and Stream 2 degenerates to stratified normalization, a well-established variance-reduction technique. When $\hat{\Delta}(x) \leq 0$, skills are no longer helpful and the correction is clipped to zero; the curriculum handles skill retirement. This architecture avoids mixed-normalization bias while maintaining interpretability and computational efficiency.

3.4 Internalization-Aware Curriculum

SKILLC employs the contrastive signal at two timescales. The advantage estimator uses batch-level $\hat{\Delta}(x)$ for fine-grained per-task credit assignment, while the curriculum uses a validation-level counterpart for coarser-grained withdrawal control. This separation is important because per-batch estimates are noisy and optimized for gradient direction, whereas curriculum decisions require stability.

For each task type k, we periodically (every d training steps) compute a validation contrastive gap and smooth it across checkpoints:

$$\begin{aligned} \Delta^{(t)}_{\text{val}}(k) &= \text{SR}^{(t)}_{\text{with}}(k) - \text{SR}^{(t)}_{\text{without}}(k), \\ \bar{\Delta}^{(t)}_{\text{val}}(k) &= \alpha_{\text{val}} \bar{\Delta}^{(t-1)}_{\text{val}}(k) + (1-\alpha_{\text{val}}) \Delta^{(t)}_{\text{val}}(k), \\ \text{gate}^{(t)}(k) &= \sigma\left(\bar{\Delta}^{(t)}_{\text{val}}(k)/T\right). \end{aligned} \tag{7}$$

The gate controls three mechanisms: (i) attribution strength $\lambda^{(t)}_{\text{dyn}}(k) = \lambda\, \text{gate}^{(t)}(k)$, (ii) rollout fraction $\rho^{(t)}_{\text{with}}(k)$, and (iii) active-set pruning for multi-category environments:

$$\begin{aligned} \mathcal{K}^{(t)}_{\text{keep}} &= \left\{ k \in \mathcal{K}^{(t-1)}_{\text{active}} : \bar{\Delta}^{(t)}_{\text{val}}(k) \geq \tau_{\text{retire}} \right\}, \\ \mathcal{K}^{(t)}_{\text{active}} &= \text{TopK}_{\bar{\Delta}^{(t)}_{\text{val}}}\left(\mathcal{K}^{(t)}_{\text{keep}}, K^{(t)}\right). \end{aligned} \tag{8}$$

For single-skill environments, the active set is constant and only mechanisms (i) and (ii) apply. Once a task is retired, it never re-enters, maintaining monotone structure and avoiding oscillatory behavior near zero.

3.5 Training Objective

Training follows standard GRPO with PPO-style clipping and KL regularization, with the vanilla group-normalized advantage replaced by $\hat{A}_{\text{CSCA}}$:

$$J(\theta) = \mathbb{E}_{x,\{\tau_i\}}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{T_i}\sum_{t=1}^{T_i}\ell_{i,t}(\theta)\right],$$
$$\ell_{i,t}(\theta) = \mathcal{L}_{\text{clip}}\Big(\rho_{i,t}, \hat{A}_{\text{CSCA}}(i,x)\Big) - \beta\,\text{KL}_{i,t}, \tag{9}$$

where $\rho_{i,t} = \pi_\theta(a_{i,t} \mid h_{i,t})/\pi_{\theta_{\text{old}}}(a_{i,t} \mid h_{i,t})$ and $\text{KL}_{i,t} = D_{\text{KL}}(\pi_\theta(\cdot \mid h_{i,t}) \,\|\, \pi_{\text{ref}}(\cdot \mid h_{i,t}))$.

SKILLC modifies credit assignment within a standard policy gradient framework. It jointly optimizes skill-injected and skill-free trajectories, with the contrastive term redirecting credit toward behaviors that succeed autonomously. However, effectiveness depends critically on signal informativeness: strong contrastive signals enable effective credit redistribution and responsive curriculum adaptation, while weak signals on high-performance tasks can cause premature skill retirement and performance degradation.

4 Experiments

4.1 Experimental Setup

Environments. We evaluate SKILLC on two representative agent benchmarks. ALFWorld (Shridhar et al., 2020) is a text-based household environment aligned with the ALFRED embodied benchmark, where agents complete long-horizon tasks through multi-step planning and object interaction across six task categories: Pick, Look, Clean, Heat, Cool, and Pick2. WebShop (Yao et al., 2022a) is a realistic online shopping environment requiring multi-step search, browsing, and purchase decisions under natural-language instructions.

Baselines. We compare SKILLC against representative methods from five categories. Training-free methods include ReAct (Yao et al., 2022b), Reflexion (Shinn et al., 2023), Mem0 (Chhikara et al., 2025), and ExpeL (Zhao et al., 2024). Direct RL methods include PPO (Schulman et al., 2017), RLOO (Ahmadian et al., 2024), GRPO (Shao et al., 2024), and GiGPO (Feng et al., 2026). On-policy self-distillation methods include OPSD (Zhao et al., 2026), Skill-SD (Wang et al., 2026b), RLSD (Yang et al., 2026), and SDAR (Lu et al., 2026a). Skill-augmented RL methods, including EvolveR (Wu et al., 2025), SKILLRL (Xia et al., 2026), RetroAgent (Zhang et al., 2026b), Skill1 (Shi et al., 2026), and D2Skill (Tu et al., 2026), all retain external skill access at inference time. Skill-internalization RL methods, including SKILL0 (Lu et al., 2026b) and our SKILLC, are evaluated *without* external skills at inference time.

Implementation Details and Metrics. All experiments use Qwen2.5-7B-Instruct (Yang et al., 2024) as the base model, and the full hyperparameters are in Appendix B. For ALFWorld, we use the six task-category skills from SKILL0: Pick, Look, Clean, Heat, Cool, and Pick2. For WebShop, we construct a single shopping skill covering search, browsing, and purchase strategies specific to the e-commerce domain. The skills used in both environments are obtained using the skill learning prompts proposed in SKILL0(Lu et al., 2026b). We report with-skill success rate (SR_{with}) for skill-augmented methods and without-skill success rate ($\text{SR}_{\text{without}}$) for skill-internalization methods on the test split. For ALFWorld, we additionally track the internalization gap $\Delta_{\text{val}} = \text{SR}_{\text{with}} - \text{SR}_{\text{without}}$ on the validation split to measure progress during training. For WebShop, we report both task score (0–100, measuring attribute relevance) and success rate (%, binary completion). All reported metrics represent the average of 5 independent runs.

4.2 Main Results

Table 1 reports test results on ALFWorld and WebShop. SKILLC achieves the strongest *without-skill* performance among all skill-internalized methods, improving over SKILL0 from 85.9% to 90.6% on ALFWorld and from 70.9% to 74.0% on WebShop. The improvement is attributable to resolving internalization blindness: by making the task-level contrastive gap $\hat{\Delta}(x)$ observable within each policy update, CSCA redirects credit toward skill-free success throughout training, rather than relying on the policy to transfer behaviors passively after curriculum withdrawal. On ALFWorld, SKILLC also matches D2Skill and outperforms SKILLRL, both of which retain external skill access at inference time. D2Skill also samples paired rollouts within the same update, but uses the performance gap as a hindsight bonus for skill-injected trajectories under mixed normalization, reinforcing runtime skill use rather than internalization. Matching its performance without skill access confirms that redirecting the same contrastive signal toward skill-free credit assignment is sufficient.

Method	ALFWorld							WebShop	
	Pick	*Look*	*Clean*	*Heat*	*Cool*	*Pick2*	*Avg.*	*Score*	*Succ.*
Training-free Methods									
Zero-Shot	33.4	21.6	19.3	6.9	2.8	3.2	14.8	26.4	7.8
ReAct (Yao et al., 2022b)	48.5	35.4	34.3	13.2	18.2	17.6	31.2	46.2	19.5
Reflexion (Shinn et al., 2023)	62.0	41.6	44.9	30.9	36.3	23.8	42.7	58.1	28.8
Mem0 (Chhikara et al., 2025)	54.0	55.0	26.9	36.4	20.8	7.7	33.6	23.9	2.0
ExpeL (Zhao et al., 2024)	21.0	67.0	55.0	52.0	71.0	6.0	46.3	30.9	11.2
Direct RL Methods									
PPO (Schulman et al., 2017)	92.3	64.0	92.5	89.5	80.3	68.8	80.4	81.4	68.7
RLOO (Ahmadian et al., 2024)	87.6	78.2	87.3	81.3	71.9	48.9	75.5	80.3	65.7
GRPO (Shao et al., 2024)	90.8	66.1	89.3	74.7	72.5	64.7	77.6	79.3	66.1
GiGPO (Feng et al., 2026)	97.7	82.7	98.8	83.7	89.3	79.2	90.8	84.4	72.8
On-Policy Self-Distillation Methods									
OPSD (Zhao et al., 2026)	50.0	60.0	22.7	21.4	17.6	9.5	32.8	4.5	2.3
GRPO+OPSD (Zhao et al., 2026)	91.4	61.5	100.0	87.5	76.5	52.2	80.4	86.8	76.5
Skill-SD (Wang et al., 2026b)	93.9	93.8	90.9	100.0	69.2	68.4	85.1	86.1	76.5
RLSD (Yang et al., 2026)	100.0	87.5	92.3	58.8	80.0	65.2	82.0	87.4	77.3
SDAR (Lu et al., 2026a)	94.7	75.0	100.0	86.7	68.2	78.9	85.9	89.4	82.8
Skill-Augmented RL Methods									
EvolveR (Wu et al., 2025)	64.9	33.3	46.4	13.3	33.3	33.3	43.8	42.5	17.6
SKILLRL (Xia et al., 2026)	97.9	71.4	90.0	90.0	95.5	87.5	89.9	85.2	72.7
RetroAgent (Zhang et al., 2026b)	97.9	90.9	99.2	92.9	85.3	91.0	94.9	88.9	82.3
Skill1 (Shi et al., 2026)	100.0	98.6	97.3	99.2	96.1	96.0	97.5	89.7	82.9
D2Skill (Tu et al., 2026)	97.1	100.0	75.0	87.5	100.0	78.6	90.6	91.1	80.5
Skill-Internalization RL Methods									
SKILL0 (Lu et al., 2026b)	100.0	84.6	91.1	84.2	88.9	55.5	85.9	83.3	70.9
SKILLC	88.5	78.6	91.2	94.7	100.0	88.9	90.6	85.6	74.0

Table 1: Main results on ALFWorld and WebShop (Success Rate, %). Within each category, cells with a yellow background denote the best results, and a grey background denotes the second-best results.

4.3 Ablation Study

Table 2 evaluates the contribution of SKILLC's main components.

Paired rollouts. Without pairing, each task is sampled under only one prompt condition per update, so the task-level contrastive gap $\hat{\Delta}(x)$ is unavailable. Performance drops to 87.3 %, showing that within-update comparison is the main source of internalization credit.

Dual-stream normalization. Replacing per-condition normalization with a mixed baseline lowers performance to 88.4 %. When with-skill and without-skill trajectories share one normalization reference, high-reward skill-injected rollouts inflate the baseline and suppress credit to successful skill-free trajectories.

Curriculum. Curriculum ablations are consistently worse than the full model, but their effect is smaller than removing pairing or dual-stream credit assignment. Fixing the curriculum gate reduces SR to 88.9 %, removing pruning gives 89.5 %, and fixing ρ_{with}=0.5 gives 89.8 %. The worst variant, high init score, reaches only 87.1 %, showing that an overly optimistic prior delays correction. Overall, the validation signal is most useful when it jointly modulates attribution strength, rollout allocation, and retirement.

Variant	SR	Δ
single-condition updates (no pairing)	87.3	−3.3
w/ mixed norm	88.4	−2.2
fixed curriculum gate (= 0.5)	88.9	−1.7
w/o active-skill pruning	89.5	−1.1
fixed ρ_{with}=0.5	89.8	−0.8
high init score ($\bar{\Delta}^{(0)}_{\text{val}}$=0.3)	87.1	−3.5
SKILLC	**90.6**	**ref.**

Table 2: Ablation on ALFWorld. *SR*: without-skill success rate on the test set. Δ: gap relative to full SKILLC.

4.4 Internalization Schedule Analysis

Figure 3 traces the internal CSCA signals to confirm that the gap closure observed in Figure 4 is driven by the mechanism rather than by incidental factors. During early training the gate remains

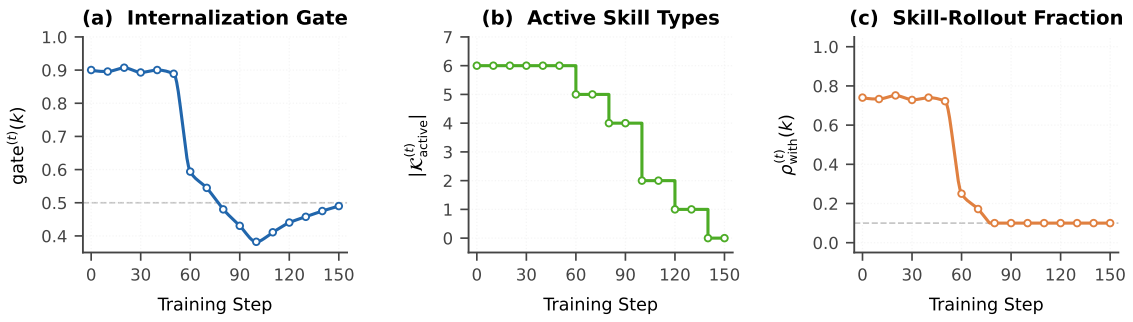

Figure 3: Internalization schedule during CSCA training on ALFWorld. (a) Internalization gate tracking the smoothed validation gap across all task types. (b) Active skill set size showing monotone pruning from six to zero task categories. (c) Skill-rollout fraction decreasing as internalization progresses.

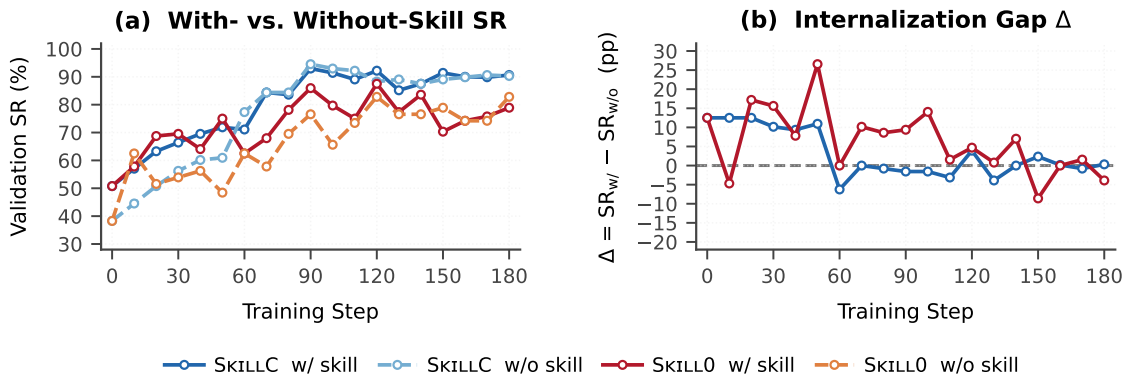

Figure 4: Training dynamics on ALFWorld. (a) With-skill and without-skill validation success rates over training steps. (b) Internalization gap over training steps.

near high values, indicating that skill prompts are still strongly beneficial; at this stage SKILLC follows essentially the same trajectory as SKILL0, which rules out a favorable initialization as the cause of later divergence. As training progresses, $\bar{\Delta}^{(t)}_{\text{val}}$ declines as the policy begins to internalize skills, pulling the gate below 0.5 and triggering a staggered retirement sequence at irregular intervals that reflect per-type heterogeneity in internalization speed. Each retirement immediately reduces the rollout fraction, which is mechanically tied to the gate, so no manual schedule is required. By contrast, SKILL0 withdraws skills on a fixed budget preset before training; without a live contrastive signal its withdrawal timing is decoupled from actual internalization progress, and the gap between the with-skill and without-skill streams persists until the schedule forces retirement. The three-way synchronization between the gate, the active set, and the rollout fraction confirms that SKILLC's internalization schedule is wholly data-driven rather than externally imposed.

4.5 Training Dynamics and Analysis

Figure 4 illustrates the central problem that motivates SKILLC. For SKILL0, the with-skill validation SR consistently exceeds the without-skill SR by a substantial margin throughout training, with no sign of spontaneous convergence. Because SKILL0 always samples all rollouts under the same prompt condition, the GRPO objective never observes the contrastive signal Δ_{val} and therefore cannot steer the policy toward autonomous success. The gap at the curriculum switch point is particularly stark: internalization debt accumulated during the skill-injection phase is exposed all at once when skills are withdrawn. This sustained gap is precisely what we term *internalization blindness*.

Under SKILLC, the internalization gap starts large at the beginning, declines steadily through the early-middle phase, crosses zero during mid-training, and stabilizes slightly below zero in later stages. This monotonic decline reflects the fact that CSCA provides the policy with a continuous training signal for autonomous success through-

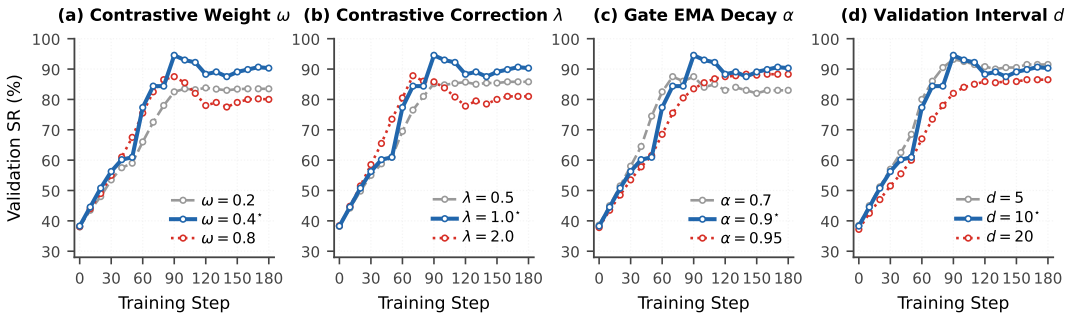

Figure 5: Hyperparameter sensitivity on ALFWorld and each panel varies one CSCA hyperparameter.

out the entire training run, rather than concentrating improvement at a discrete curriculum switch. As shown in Figure 3, the internalization gate, active-set size, and skill-rollout fraction all decrease in lockstep with the gap, confirming that the internalization schedule responds directly to measured contrastive evidence rather than a preset timetable. Once the gap stabilizes below zero, SKILLC reaches its best without-skill validation performance and skill retirement is complete, providing a natural stopping criterion for internalization-oriented training.

4.6 Hyperparameter Analysis

Figure 5 summarizes the effect of four key CSCA hyperparameters while fixing the others at their default values (ω=0.4, λ=1.0, α=0.9, d=10). The default curve (marked $\star$) is aligned with the SKILLC without-skill trajectory in Figure 4 and achieves the highest validation performance. Moving away from this point consistently hurts performance in an interpretable way: smaller ω or λ weakens the contrastive signal and leads to flatter convergence around 85 %, whereas larger values improve the early phase but introduce clear late-stage instability, ending near 80 %. For the curriculum controller, a smaller α reacts too aggressively to short-term validation noise and drops from an 87.5 % peak to about 83 %, while a larger α updates too conservatively and tops out around 88.3 %. Similarly, more frequent validation (d=5) stays close to the default but does not surpass it despite higher overhead, whereas less frequent validation (d=20) slows adaptation and caps performance near 86.5 %. Overall, the figure suggests that SKILLC is reasonably stable under nearby hyperparameter changes, with the default configuration providing the best trade-off across all four axes.

	Time / Step (s)			Active Skills		
Method	50	100	150	50	100	150
GRPO (no skill)	941	921	908		—	
SKILL0	1078	1061	1035	6	4	2
SKILLC	1187	1132	1095	6	2	0

Table 3: Computational cost on ALFWorld. Time/Step is median wall-clock seconds per training step. Active Skills counts remaining task types in the curriculum.

4.7 Computational Cost

All timings use 8 A100-80GB GPUs with batch size 8 and group size $G = 8$. GRPO (skill-free) serves as baseline. SKILL0 adds static six-skill context, yielding $+14\%$ overhead at step 20. SKILLC incurs $+26\%$ early overhead due to paired rollouts, but as skills retire (6 to 0 active skills), the rollout budget decreases and overhead stabilizes at $+20\%$ over SKILL0. The increased generation time at later steps reflects improved policy capability rather than algorithmic cost. Total extra compute is approximately $+30\%$ relative to GRPO.

5 Conclusion

We address internalization blindness through paired contrastive rollouts and dual-stream advantage estimation, redirecting task-level performance gaps into internalization credit assignment without distorting the optimization target. By reallocating trajectories under condition-wise normalization, we train skill-free success from the start rather than relying on passive transfer after schedule-driven withdrawal. SKILLC improves over prior work by 4.7% on ALFWorld and 3.1% on WebShop, with full data-driven curriculum control. Future work includes dynamic skill discovery, and generalization to hierarchical skill structures.

Limitations

SKILLC relies on several assumptions that may limit its applicability:

Skill inventory structure. The method assumes fixed, task-aligned skill inventories defined before training. In open-ended settings where skill quality or relevance varies dynamically, or where skill-to-task mappings are uncertain, the curriculum gating mechanism may produce unstable retirement decisions.

Validation-level bottleneck. Reliable internalization progress estimation requires sufficient validation coverage. In environments with extreme class imbalance or rare task types, the smoothed validation signal may become noisy, potentially triggering premature skill retirement.

Computational overhead. Paired sampling incurs $\sim 26\%$ early overhead relative to standard GRPO (Table 3), which may be prohibitive for large-scale training despite the overhead declining as skills retire.

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12248–12267.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*.

Hongcheol Cho, Ryangkyung Kang, and Youngeun Kim. 2026. Skillret: A large-scale benchmark for skill retrieval in llm agents. *arXiv preprint arXiv:2605.05726*.

Zhiyuan Fan, Wenwei Jin, Feng Zhang, Bin Li, Yihong Dong, Yao Hu, and Jiawei Li. 2026. Evolving-rl: End-to-end optimization of experience-driven self-evolving capability within agents. *arXiv preprint arXiv:2605.10663*.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. 2026. Group-in-group policy optimization for llm agent training. *Advances in Neural Information Processing Systems*, 38:46375–46408.

Dawei Li, Zongxia Li, Hongyang Du, Xiyang Wu, Shihang Gui, Yongbei Kuang, and Lichao Sun. 2026a. Graph of skills: Dependency-aware structural retrieval for massive agent skills. *arXiv preprint arXiv:2604.05333*.

Hao Li, Chunjiang Mu, Jianhao Chen, Siyue Ren, Zhiyao Cui, Yiqun Zhang, Lei Bai, and Shuyue Hu. 2026b. Organizing, orchestrating, and benchmarking agent skills at ecosystem scale. *arXiv preprint arXiv:2603.02176*.

Yu Li, Rui Miao, Zhengling Qi, and Tian Lan. 2026c. Arise: Agent reasoning with intrinsic skill evolution in hierarchical reinforcement learning. *arXiv preprint arXiv:2603.16060*.

Jiaqi Liu, Yaofeng Su, Peng Xia, Siwei Han, Zeyu Zheng, Cihang Xie, Mingyu Ding, and Huaxiu Yao. 2026a. Simplemem: Efficient lifelong memory for llm agents. *arXiv preprint arXiv:2601.02553*.

Yujian Liu, Jiabao Ji, Li An, Tommi Jaakkola, Yang Zhang, and Shiyu Chang. 2026b. How well do agentic skills work in the wild: Benchmarking llm skill usage in realistic settings. *arXiv preprint arXiv:2604.04323*.

Zhengxi Lu, Zhiyuan Yao, Zhuowen Han, Zi-Han Wang, Jinyang Wu, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, and 1 others. 2026a. Self-distilled agentic reinforcement learning. *arXiv preprint arXiv:2605.15155*.

Zhengxi Lu, Zhiyuan Yao, Jinyang Wu, Chengcheng Han, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, and Yongliang Shen. 2026b. Skill0: In-context agentic reinforcement learning for skill internalization. *arXiv preprint arXiv:2604.02268*.

Siru Ouyang, Jun Yan, Yanfei Chen, Rujun Han, Zifeng Wang, Bhavana Dalvi Mishra, Rui Meng, Chun-Liang Li, Yizhu Jiao, Kaiwen Zha, and 1 others. 2026. Skillos: Learning skill curation for self-evolving agents. *arXiv preprint arXiv:2605.06614*.

Hongji Pu, Xinyuan Song, and Liang Zhao. 2026. Skillops: Managing llm agent skill libraries as self-maintaining software ecosystems. *arXiv preprint arXiv:2605.13716*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Junhao Shen, Teng Zhang, Xiaoyan Zhao, and Hong Cheng. 2026. Dynamic skill lifecycle management for agentic reinforcement learning. *arXiv preprint arXiv:2605.10923*.

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

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652.

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

Weihang Su, Jianming Long, Qingyao Ai, Yichen Tang, Changyue Wang, Yiteng Tu, and Yiqun Liu. 2026. Skill retrieval augmentation for agentic ai. *arXiv preprint arXiv:2604.24594*.

Songjun Tu, Chengdong Xu, Qichao Zhang, Yaocheng Zhang, Xiangyuan Lan, Linjing Li, and Dongbin Zhao. 2026. Dynamic dual-granularity skill bank for agentic rl. *arXiv preprint arXiv:2603.28716*.

Chenxi Wang, Zhuoyun Yu, Xin Xie, Wuguannan Yao, Runnan Fang, Shuofei Qiao, Kexin Cao, Guozhou Zheng, Xiang Qi, Peng Zhang, and 1 others. 2026a. Skillx: Automatically constructing skill knowledge bases for agents. *arXiv preprint arXiv:2604.04804*.

Hao Wang, Guozhi Wang, Han Xiao, Yufeng Zhou, Yue Pan, Jichao Wang, Ke Xu, Yafei Wen, Xiaohu Ruan, Xiaoxin Chen, and 1 others. 2026b. Skill-sd: Skill-conditioned self-distillation for multi-turn llm agents. *arXiv preprint arXiv:2604.10674*.

Rong Wu, Xiaoman Wang, Jianbiao Mei, Pinlong Cai, Daocheng Fu, Cheng Yang, Licheng Wen, Xuemeng Yang, Yufan Shen, Yuxin Wang, and 1 others. 2025. Evolver: Self-evolving llm agents through an experience-driven lifecycle. *arXiv preprint arXiv:2510.16079*.

Peng Xia, Jianwen Chen, Hanyang Wang, Jiaqi Liu, Kaide Zeng, Yu Wang, Siwei Han, Yiyang Zhou, Xujiang Zhao, Haifeng Chen, and 1 others. 2026. Skillrl: Evolving agents via recursive skill-augmented reinforcement learning. *arXiv preprint arXiv:2602.08234*.

Renjun Xu and Yang Yan. 2026. Agent skills for large language models: Architecture, acquisition, security, and the path forward. *arXiv preprint arXiv:2602.12430*.

An Yang, Baosong Yang, Beichen Zhang, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

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

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

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

Haozhen Zhang, Quanyu Long, Jianzhu Bao, Tao Feng, Weizhi Zhang, Haodong Yue, and Wenya Wang. 2026a. Memskill: Learning and evolving memory skills for self-evolving agents. *arXiv preprint arXiv:2602.02474*.

Xiaoying Zhang, Zichen Liu, Yipeng Zhang, Xia Hu, and Wenqi Shao. 2026b. Retroagent: From solving to evolving via retrospective dual intrinsic feedback. *arXiv preprint arXiv:2603.08561*.

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

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

A Scientific Artifact Documentation

A.1 Model and Data

Base model: Qwen2.5-7B-Instruct (Apache 2.0 License, `https://huggingface.co/Qwen/Qwen2.5-7B-Instruct/blob/main/LICENSE`) Environments: ALFWorld ((Shridhar et al., 2020), academic use) and WebShop ((Yao et al., 2022a) , academic use) Skills: Task-category-specific procedural prompts derived from SKILL0 learning methodology(Lu et al., 2026b).

A.2 Licensing and Use

This research uses Qwen2.5-7B-Instruct under Apache 2.0, permitting both research and commercial use. ALFWorld and WebShop are used solely for academic research purposes consistent with their original licensing. All artifacts are intended for research prototypes and should not be deployed in production without explicit licensing review.

A.3 Computational Cost

Total budget: $\sim$ 60 GPU-hours on 8×A100-80GB for ALFWorld (180 training steps with batch size 8)

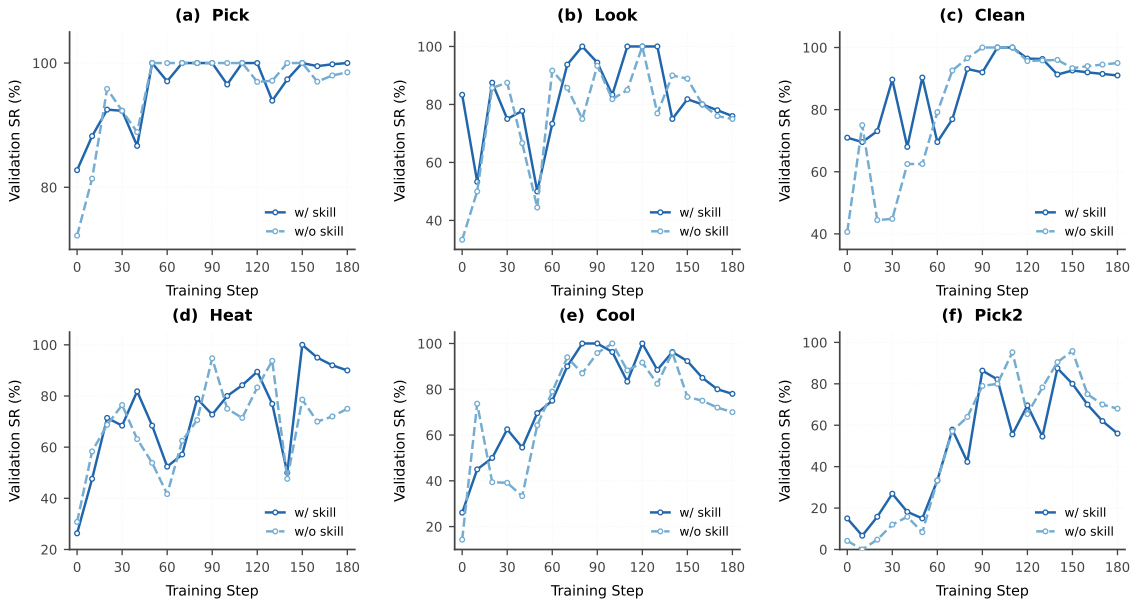

Figure 6: Per-task training dynamics across six ALFWorld task categories. Each subplot shows the with-skill (solid line) and without-skill (dashed line) validation success rates ($\mathrm{SR}_{\mathrm{with}}$ and $\mathrm{SR}_{\mathrm{without}}$) for one task type over 180 training steps.

B Implementation Details

Both ALFWorld and WebShop experiments use the same core CSCA hyperparameters, but differ in environment-specific configuration.

ALFWorld. We train on 8×A100-80 GB GPUs with batch size 8 tasks/step, group size G=8, and learning rate 1×10^{-6}. SKILLC samples four skill-injected and four skill-free paired rollouts per task, runs validation every 10 steps (so d=10), and uses the capacity schedule $K^{(t)} \in \{6, 3, 0\}$. This schedule reflects ALFWorld's six task categories; starting with all six active, then pruning to three midway through training, and finally to zero as skills are fully internalized.

WebShop. We use the same GPU configuration, batch size, and learning rate. SKILLC also samples paired rollouts and validates every 10 steps; however, because WebShop has a single unified shopping skill, there is no active-set pruning and $K^{(t)} \equiv 1$ throughout training. The gate still controls attribution strength and rollout allocation, but these apply uniformly to the single skill category rather than selecting among multiple tasks.

Core CSCA hyperparameters (both environments). Key hyperparameters are λ=1.0 (attribution strength), ω=0.4 (stream weight), δ=2.0 (advantage clipping bound), batch EMA decay α_{batch}=0.9, gate temperature T=0.1, and validation-gate EMA α_{val}=0.7. The minimal probing budget is $\rho_{\min}$=0.1, frequency-penalty weight is β_{freq}=0.01, and the retirement threshold is τ_{retire}=0.0. In early optimization steps, the contrastive gate rises gradually as reliable paired evidence accumulates, reflecting the contrastive learning signal rather than a separate initialization trick.

C Per-Task Training Dynamics Analysis

Figure 6 presents a comprehensive breakdown of the training dynamics across all six ALFWorld task categories over the full 180-step training horizon. All curves represent data collected from a single SkillC training run on the Qwen2.5-7B-Instruct base model.

Task-specific internalization patterns. The per-task breakdown reveals that internalization speed and baseline performance vary substantially across task categories. High-performance tasks (Pick, Look, Clean) begin training with with-skill SR already near 80 to 90 percent, reflecting their inherent simplicity or close alignment with the base model's capabilities. These tasks show rapid convergence and minimal internalization gap throughout the training run. In contrast, lower-performance tasks (Heat, Cool, Pick2) start below 50 percent with-skill SR and exhibit larger internalization gaps in early training (steps 0 to 50). For these harder tasks, the training from step 50 to 150 shows clear

gap reduction as the policy learns to handle core task logic without skill assistance.

Stability post internalization. Figure 6 demonstrates that once the internalization gap closes (typically by step 100 to 120), both with-skill and without-skill curves stabilize and track closely together. In the later training regime (steps 150 to 180), performance stabilizes at the learned level for most tasks, with minor fluctuations reflecting natural variance in the validation signal. This convergence validates the core hypothesis that continuous contrastive feedback enables the policy to acquire autonomous competence gradually, rather than experiencing sudden performance degradation at a curriculum switch. The fact that without-skill performance remains robust in the later stages (steps 130 to 180) confirms that SkillC has successfully internalized task knowledge rather than merely learning to suppress skill usage.

D Theoretical Justification

This appendix provides theoretical justification for the dual-stream design of CSCA. We focus on the core asymptotic behavior to show that the contrastive correction term automatically decays as the policy internalizes skills, preventing permanent bias.

D.1 Asymptotic Correction Decay

We first formalize the limiting regime in which internalization is complete and the explicit contrastive correction should disappear automatically.

Lemma D.1 (Asymptotic Correction Decay)**.** *Suppose π_θ has fully internalized skill $\mathcal{S}$, i.e., for all tasks x, states s, and actions a,*

$$P(a \mid s, x, \mathcal{S}; \pi_\theta) = P(a \mid s, x; \pi_\theta).$$

Then $\Delta(x) = 0$, and as $\hat{\Delta}(x) \to 0$, the SKILLC *estimator reduces to*

$$\hat{A}_{\text{CSCA}}(i, x) \xrightarrow{\hat{\Delta}\to 0} A_{\text{ep}}(i) + \omega \cdot \frac{R_i - \mu^{\text{cond}}(x)}{\sigma^{\text{cond}}(x) + \epsilon},$$

where $A_{\text{ep}}(i)$ is the global ranking term, and $\mu^{\text{cond}}(x), \sigma^{\text{cond}}(x)$ are the within-condition statistics of the branch to which trajectory i belongs.

Proof. If the policy fully internalizes $\mathcal{S}$, then by definition $P(a \mid s, x, \mathcal{S}; \pi_\theta) = P(a \mid s, x; \pi_\theta)$ for all tasks x, states s, and actions a. This implies that the trajectory distributions τ^+ and τ^- become identical, so $\Delta(x) = \mathbb{E}[R \mid x, z = 1] - \mathbb{E}[R \mid x, z = 0] = 0$. Since $\hat{\Delta}(x)$ is a smoothed estimate via EMA, we have $\hat{\Delta}(x) \to 0$ as internalization progresses. Therefore, $C(x) = \lambda_{\text{eff}}(x)[\hat{\Delta}(x)]_+ \to 0$.

By construction, the condition-wise advantage term is

$$A^{\text{cond}}_{\text{contra}}(i, x) = \frac{R_i - \mu^{\text{cond}}(x)}{\sigma^{\text{cond}}(x) + \epsilon} \pm C(x),$$

where the sign is negative for skill-injected trajectories ($z_i = 1$) and positive for skill-free ($z_i = 0$). As $C(x) \to 0$, the correction terms $\pm C(x)$ vanish identically, leaving only the within-condition normalization:

$$A^{\text{cond}}_{\text{contra}}(i, x) \to \frac{R_i - \mu^{\text{cond}}(x)}{\sigma^{\text{cond}}(x) + \epsilon}.$$

Therefore, the full CSCA advantage approaches

$$\hat{A}_{\text{CSCA}}(i, x) \to A_{\text{ep}}(i) + \omega \cdot \frac{R_i - \mu^{\text{cond}}(x)}{\sigma^{\text{cond}}(x) + \epsilon},$$

which is the sum of a global ranking term and a stratified within-condition term. This completes the proof. □

D.2 Design Implications

The limiting form in Lemma D.1 has three important implications for the CSCA design.

Automatic correction decay. The key insight is that the correction term $C(x) = \lambda_{\text{eff}}(x)[\hat{\Delta}(x)]_+$ automatically decays to zero as the policy internalizes each skill, eliminating the risk of permanent bias. Unlike methods that apply fixed correction terms or external bonuses, CSCA ensures that its contrastive signal is transient and task-dependent, vanishing precisely when no longer needed.

Stratified advantage structure. At convergence, CSCA reduces to a decomposition of global ranking plus per-condition stratification. The stratified term provides variance reduction when the skill-injected and skill-free branches have heterogeneous reward distributions (e.g., $\sigma^+(x) \gg \sigma^{(\text{free})}(x)$ during early skill-injection phases), as it avoids inflating the baseline with high-reward skill-injected trajectories. This stratification is a natural consequence of independent within-condition normalization rather than a manually imposed mechanism.

Consistency with dual stream design. The two stream structure: one preserving global ranking, one applying per condition normalization, is justified by the limiting behavior. Stream 1 ensures that the policy maintains a coherent global quality signal across all trajectories, while Stream 2's correction is temporary and task conditioned. At internalization, Stream 2's contribution naturally vanishes, leaving only stratified advantage estimation, which is a well understood variance reduction technique in policy gradient methods.

E Broader Impact and Potential Risks

Policy robustness and exploration. Contrastive credit assignment may reduce the policy's exploration diversity when skills are temporarily withdrawn. Future work should investigate whether CSCA-trained policies exhibit mode collapse under distributional shift.

Deployment considerations. While SKILLC improves autonomous performance, it does not eliminate the need for skill engineering. Practitioners deploying this method should be aware that internalization success depends critically on skill quality and task-skill alignment, both of which may degrade in noisy real-world environments.

F AI Assistant Use in Research and Writing

This paper was developed with assistance from multiple AI tools in a transparent and limited manner, consistent with ACL ethics guidelines:

- **Code Development**: Claude (Anthropic) was used to assist with implementation of the CSCA algorithm, PyTorch code optimization, and debugging of training pipelines. All core algorithmic logic, experimental design decisions, and hyperparameter tuning were conducted by the researchers. Claude was consulted for standard library usage and code refactoring only.

- **Paper Writing and Editing**: ChatGPT (OpenAI) was used for English language polishing, grammar correction, and stylistic refinement of preliminary drafts. All technical content, experimental interpretations, claims, and conceptual contributions were authored and verified by the researchers against experimental results.

- **Literature Review and Background Research**: Gemini (Google) was consulted for background information gathering on related work in agentic RL, skill-based learning, and contrastive learning methods. All citations, comparisons with prior work, and positioning of contributions were performed independently by the authors.