

# EVOLVER: SELF-EVOLVING LLM AGENTS THROUGH AN EXPERIENCE-DRIVEN LIFECYCLE

000  
001  
002  
003  
004  
005 **Anonymous authors**  
006 Paper under double-blind review  
007  
008  
009  
010  
011  
012  
013  
014  
015  
016  
017  
018  
019  
020  
021  
022  
023  
024  
025  
026  
027  
028  
029  
030  
031  
032  
033  
034  
035  
036  
037  
038  
039  
040  
041  
042  
043  
044  
045  
046  
047  
048  
049  
050  
051  
052  
053  
EVOLVER: SELF-EVOLVING LLM AGENTS THROUGH  
AN EXPERIENCE-DRIVEN LIFECYCLE

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

## ABSTRACT

Current Large Language Model (LLM) agents show strong performance in tool use, but lack the crucial capability to systematically learn from their own experiences. While existing frameworks mainly focus on mitigating external knowledge gaps, they fail to address a more fundamental limitation: the inability to iteratively refine problem-solving strategies. In this work, we introduce **EvolveR**, a framework designed to enable agent to self-improve through a complete, closed-loop experience lifecycle. This lifecycle comprises two key stages: (1) **Offline Self-Distillation**, where the agent’s interaction trajectories are synthesized into a structured repository of abstract, reusable strategic principles; (2) **Online Interaction**, where the agent interacts with tasks and actively retrieves distilled principles to guide its decision-making, accumulating a diverse set of behavioral trajectories. This loop employs a policy reinforcement mechanism to iteratively update the agent based on its performance. We demonstrate the effectiveness of EvolveR on complex multi-hop question-answering benchmarks, where it achieves superior performance over strong agentic baselines. Our work presents a comprehensive blueprint for agents that learn not only from external data but also from the consequences of their own actions, paving the way for more autonomous and continuously improving systems.

## 1 INTRODUCTION

Large Language Models (LLMs) have driven the development of autonomous agents capable of solving diverse tasks through advanced reasoning and tool use (Shen et al., 2023; Luo et al., 2025; ang Gao et al., 2025). However, a significant limitation emerges when these agents engage in sequential tasks: each interaction is treated independently. They approach tasks as isolated episodes, suffering from operational amnesia and failing to learn from past successes or avoid prior mistakes(Yao et al., 2023b). This inability to leverage experience fundamentally hinders their development toward greater autonomy and intelligence.

Humans, by contrast, learn through a continuous lifecycle, leveraging both successes and failures to refine strategies over time (Flesch et al., 2018). For example, a student solving math problems reflects on recurring errors and successful approaches to extract general problem-solving strategies. This cycle of interaction, reflection, and abstraction is the cornerstone of developing expertise (Anderson, 1993). Endowing LLM agents with a comparable lifecycle is the key to bridging the gap between episodic problem-solving and sustainable self-improvement. While existing frameworks like Retrieval-Augmented Generation (RAG) effectively address knowledge gaps, they fail to solve a more fundamental limitation: the agent’s inability to systematically learn from the consequences of its own interactions (Yan et al., 2025).

As Figure 1 shows, prior works have attempted to address this limitation, but with critical shortcomings. Researchers store natural language reflections across tasks with a powerful external LLM in an external memory (Zhao et al., 2024; Zhou et al., 2025). While resource-efficient, this approach treats such reflections as a transient hint, leaving the agent’s intrinsic policy unchanged. On the other hand, learning by recalling raw cases retrieves entire past trajectories to directly guide decision-making. However, this reliance on raw cases struggles to generalize and, more importantly, fails to abstract. The agent merely mimics past solutions instead of distilling the reusable strategic principles that made them successful (Chen et al., 2023).

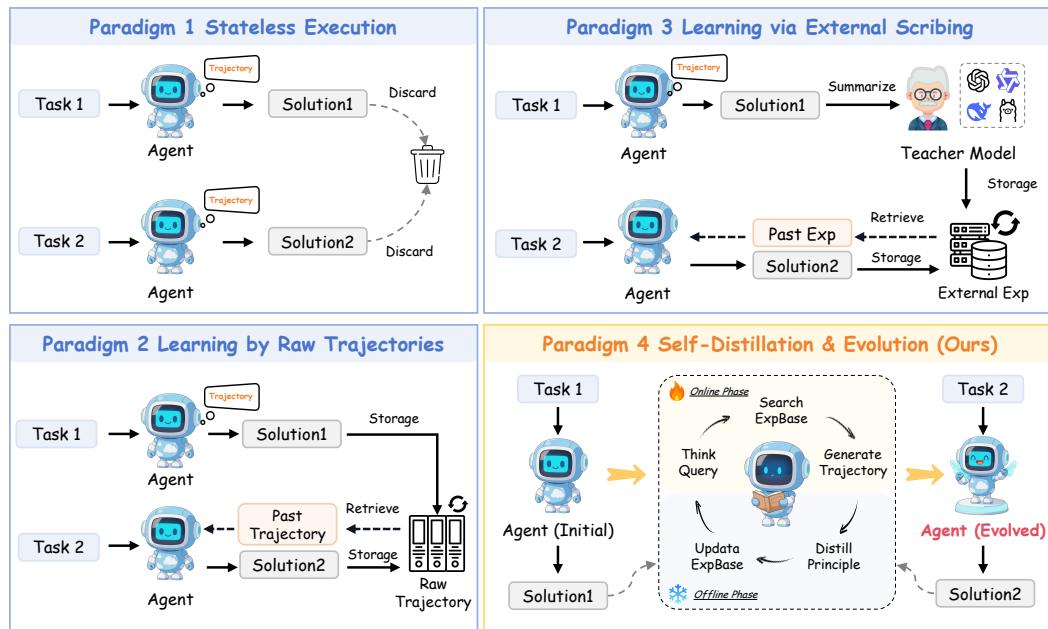


Figure 1: An illustration of four major paradigms for LLM agent learning. (1) **Stateless Execution**: Standard agents discard experiences after each task; (2) **Learning by Raw Trajectories**: Agents retrieve raw, un-distilled past trajectories; (3) **Learning via External Scribing**: Agents rely on an external teacher model to distill insights; (4) **EvolveR (Ours)**: A complete, self-contained lifecycle where the agent autonomously distills its own experiences into principles and evolves its policy.

To overcome these challenges, we introduce EvolveR, a framework that enables agents to self-evolve by utilizing their own experiences. EvolveR implements a full experience lifecycle, in which agents collect trajectories through Online Interaction, distill them into a library of abstract strategic principles during Offline Self-Distillation, and subsequently learn to apply these principles to new tasks. Crucially, EvolveR completes the experience lifecycle with a reinforcement learning mechanism that enables the agent to utilize experience. The agent does not merely mimic its past interactions; it evolves based on what it has learned. EvolveR maintains a dynamic experience base where newly distilled principles are semantically deduplicated and continuously evaluated via a metric score that tracks historical effectiveness.

We demonstrate EvolveR’s effectiveness on complex question-answering benchmarks, where it significantly outperforms strong agentic baselines. Our contributions can be summarized as follows:

- **We propose the Experience-Driven Self-Evolution Paradigm, a novel, closed-loop lifecycle for LLM agents.** In contrast to agents that forget past interactions, EvolveR systematically integrates a complete cycle of *online interaction*, *offline experiences self-distillation* and *policy evolution*. This process enables the agent to continuously transform raw trajectories into a curated repository of strategic principles, establishing a foundation for adaptive agents.
- **We introduce a complete system for dynamic experiences curation.** This system goes far beyond simple experience storage. It features: (1) a **self-distillation** mechanism, where the agent autonomously distills principles from previous interactions; and (2) a full **maintenance pipeline**, including semantic deduplication, integration, and quality control guided by a dynamic metric score.
- **We provide extensive empirical validation of the EvolveR paradigm across multiple model scales.** Our experiments on a diverse suite of complex QA benchmarks demonstrate the effectiveness of our approach. Detailed ablation studies confirm that the synergy of our proposed curation and self-distillation mechanisms is critical to the framework’s success, revealing a key insight: while the self-distillation mechanism is less effective on smaller-scale models, it **surpasses distillation by a stronger, external teacher model** at the 3B scale, validating the importance of cognitive alignment.

108  
109  

## 2 RELATED WORK

110  
111  

### 2.1 CONTINUAL LEARNING AND SELF-EVOLVING AGENTS

112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
Continual learning (CL) aims to enable models to learn sequentially while mitigating catastrophic forgetting (Parisi et al., 2019; Wang et al., 2024). While various replay-based and regularization methods have been proposed, most CL paradigms assume predefined task boundaries and focus on knowledge preservation rather than active acquisition in open-ended environments (Kirkpatrick et al., 2017; Ding et al., 2024; Huai et al., 2025a;b). The pursuit of self-evolving agents moves beyond these limitations by enabling systems to grow autonomously from experience. Frameworks such as Reflexion and Generative Agents explore self-improvement through self-play and reflective reasoning, often storing past trajectories as memory to guide future actions (Shinn et al., 2023; Wei et al., 2022; Yao et al., 2023a; Besta et al., 2024; Yao et al., 2023b). However, these systems either store raw, unstructured data or rely on memory mechanisms that are not designed for the systematic, long-term distillation and refinement of abstract strategic knowledge. Instead of relying on external data streams, our agent autonomously generates and refines its own experiences through an iterative cycle of online interaction and offline reflection.124  
125  
126  

### 2.2 LLM AGENTS AND REINFORCEMENT LEARNING

127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
LLM agents have been widely explored through frameworks such as ReAct, which interleaves reasoning and actions, and Reflecion, which improves task performance via self-reflection (Yao et al., 2023b; Shinn et al., 2023). While these approaches are primarily prompt-based and stateless, they prevent long-term accumulation of strategic knowledge. External memory frameworks like ExpeL address this limitation by reusing past trajectories, but they do not enable systematic self-improvement across tasks (Zhao et al., 2024). While effective, these methods often rely on simple prompting and are inherently stateless, limiting their ability to internalize knowledge across tasks. Recent work has increasingly turned to reinforcement learning (RL) to train agents for long-horizon, multi-turn tasks. However, applying RL is challenging due to sparse rewards and the need for stable training signals. Search-R1 (Jin et al., 2025), O2-Searcher (Mei et al., 2025), and AutoRefine (Shi et al., 2025) all use RL to train LLMs to generate and interact with external search tools. While these works successfully optimize the LLM’s interaction with external factual knowledge, they do not address the broader challenge of an agent’s self-improvement through its own internal experience.139  
140  

## 3 METHOD

141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
In this section, we present **EvolveR**, a novel framework designed to enable agent self-evolution through a complete, closed-loop experience lifecycle. Inspired by the human cycle of work and reflection, our approach is structured around three core, interconnected components, as depicted in Figure 2. First, in the **Offline Experience Self-Distillation** phase, the agent’s policy parameters are frozen, and it systematically distills raw trajectories into a curated base of strategic principles. Second, during the **Online Interaction** phase, the agent applies this distilled wisdom to guide its deliberative reasoning and action, generating new, high-quality interaction data. Finally, the entire cycle is driven by a **Policy Evolution** mechanism, where the trajectories collected online are used to update the agent’s policy parameters via reinforcement learning, thus closing the loop. This iterative process allows the agent to continuously transform its interactions into evolving expertise.151  
152  
153  

### 3.1 PRELIMINARIES: FORMALIZING AGENT INTERACTION

154  
155  
156  
At each state  $t$ , the agent, situated in an unknown state  $s_t$ , selects an action  $a_t \in \mathcal{A}$  based on its policy. Our agent’s action space  $\mathcal{A}$  is designed for complex, knowledge-intensive tasks and comprises three key operations:157  
158  
159  
160  
161  
• **<search\_experience>**: Agent queries its internal experience base  $\mathcal{E}$  to retrieve relevant principles distilled from past trajectories. Environment returns retrieved principles as an observation.  
• **<search\_knowledge>**: Agent queries an external knowledge base (e.g., a search engine) to acquire factual information. Environment returns retrieved information as an observation.  
• **<answer>**: Agent outputs its final answer to the problem and concludes the interaction.

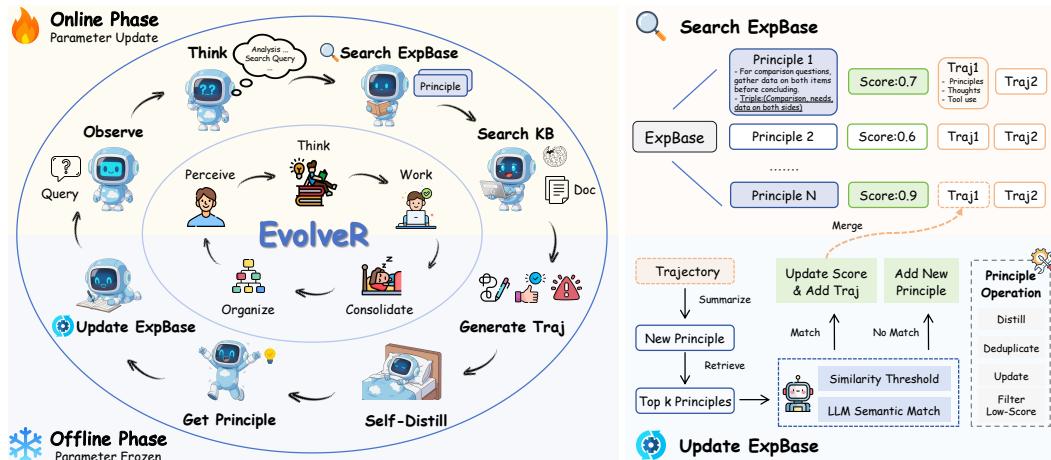


Figure 2: **Overview of the EvolveR framework’s experience lifecycle.** **Left:** The main loop alternates between an *Online Phase*, where the agent interacts with the environment and its policy parameters are updated via RL, and an *Offline Phase*, where the agent’s parameters are frozen and it performs self-distillation and maintains its Experience Base ( $\mathcal{E}$ ). **Top Right:** A detailed view of the *Search ExpBase* action, where the agent retrieves scored principles along with their associated trajectories. **Bottom Right:** The *Update ExpBase* process, which involves summarizing trajectories and applying a suite of curation operations (distill, deduplicate, update, and filter).

### 3.2 THE EVOLVER LIFECYCLE: FROM INTERACTIONS TO PRINCIPLES

#### 3.2.1 OFFLINE EXPERIENCE SELF-DISTILLATION

The core of EvolveR is a self-perpetuating lifecycle designed to transform raw interaction data into a strategic principle. This process is divided into two distinct, alternating phases: an offline self-distillation phase for distilling the principle, and an online interaction phase for applying the principle and gathering new interaction data.

**Principle from Self-Distillation.** The process begins with self-distillation. We leverage the agent’s own policy model  $\pi_\theta$  to analyze its past interaction trajectories. By adopting the persona of an expert through carefully designed prompts, the model reviews each trajectory and, based on its outcome, distills the core strategic insight into a concise natural language statement. This results in either a **guiding principle** from a success or a **cautionary principle** from a failure.

Inspired by structured memory frameworks such as Mem0 (Chhikara et al., 2025) and G-Memory (Zhang et al., 2025), each principle consists of two components: a natural language description paired with several structured knowledge triples, as illustrated in Figure 2. This self-distillation approach enables the agent to autonomously generate reusable knowledge.

**Deduplication and Integration.** To maintain a high-quality experience base ( $\mathcal{E}$ ), we do not add every distilled principle. Instead, each new principle undergoes a rigorous integration process. First, to handle redundancies arising from similar trajectories (e.g., from GRPO sampling), we perform a deduplication step. We use the agent model  $\pi_\theta$  to pair-wise check for semantic equivalence among newly generated principles that originate from the same problem, keeping only one representative from each semantically equivalent cluster.

Second, for each unique principle, we apply a two-stage matching procedure: we first retrieve the most similar existing principles from  $\mathcal{E}$  via embedding similarity, then prompt the agent model to provide a binary semantic equivalence judgment. If a principle is novel, it is added as a new entry in  $\mathcal{E}$ ; otherwise, the new trajectory is merged under the existing principle, enriching it without introducing redundancy.

Let  $p_{\text{cand}}$  be a new candidate principle distilled from trajectory  $\tau_{\text{src}}$ . We update the experience base  $\mathcal{E}$  as follows:

216  
217  
218  
219

$$\mathcal{E} \leftarrow \begin{cases} \mathcal{E} \cup \{p_{\text{cand}}\} & \text{if } \max_{p \in \mathcal{E}} \text{sim}(p_{\text{cand}}, p) < \theta_{\text{sim}} \\ \text{Merge}(\mathcal{E}, p^*, \tau_{\text{src}}) & \text{otherwise} \end{cases} \quad (1)$$

220 where  $\text{sim}(\cdot, \cdot)$  is the cosine similarity between principle,  $\theta_{\text{sim}}$  is a similarity threshold, and  $p^* = \text{argmax}_{p \in \mathcal{E}} \text{sim}(p_{\text{cand}}, p)$ . The Merge operation links  $\tau_{\text{src}}$  to its best match  $p^*$ .

223 This two-level check ensures that  $\mathcal{E}$  grows with novel insights while strengthening existing ones  
224 with new evidence.

225 **Quality Control via Dynamic Scoring.** As the experience base accumulates principles over time,  
226 it becomes essential to evaluate their practical utility and prioritize the most effective strategies. To  
227 this end, each principle tracks its usage and success counts, enabling the computation of an empirical  
228 score that reflects historical performance. We quantify the empirical utility of each principle using a  
229 metric score, which is updated as:

$$230 \quad s(p) = \frac{c_{\text{succ}}(p) + 1}{c_{\text{use}}(p) + 2} \quad (2)$$

232 where  $c_{\text{succ}}(p)$  and  $c_{\text{use}}(p)$  are the success and usage counts for a given principle  $p$ ,  $s(p)$  is the metric  
233 score.

235 This score provides a reliable measure of a principle’s historical effectiveness. To ensure the long-  
236 term health of the experience base, we periodically prune principles whose scores fall below a thresh-  
237 old  $\theta_{\text{prune}}$ . This systematic process of distillation, integration, and quality control ensures that the  
238 agent’s wisdom remains a compact and high-quality repository of its most effective strategies.

### 239 3.2.2 ONLINE INTERACTION

241 Before the online interaction begins, we include a brief cold-start phase to ensure that the model  
242 can produce well-formed interaction formats (e.g., `<think>` and `<search_experience>`). This phase  
243 provides no task knowledge or strategic guidance.

245 With the format stabilized, the online phase serves as the interactive testbed where the agent applies  
246 its distilled principles to solve problems. The agent operates within a deliberative reasoning loop  
247 (e.g., Think-Act-Observe), which enables it to engage in multi-turn, autonomous tool use. However,  
248 the core novelty of EvolveR’s online phase is not the loop itself, but how the principles retrieved  
249 from the experience base ( $\mathcal{E}$ ) fundamentally alter the agent’s behavior within it.

250 **Experience as a Strategic Principle.** Unlike standard agents that must discover reasoning pat-  
251 terns from scratch through trial and error, an EvolveR agent is guided by a strategic wisdom  
252 provided by its own past experiences. At any point in its reasoning loop, the agent can issue a  
253 `<search_experience>` action. The retrieved principles  $\mathcal{P}_k$  do not merely provide factual information;  
254 they offer heuristic guidance that shapes the agent’s subsequent reasoning. For instance, re-  
255 trieving a principle such as “For comparison questions, gather data on both items before concluding,”  
256 can directly influence the agent’s internal monologue (`<think>`) and steer its subsequent potential  
257 `<search_knowledge>` actions. This makes the agent’s exploration more efficient and less prone to  
258 common pitfalls, as it learns to follow the wisdom in its own distilled principles.

259 **Generating High-Quality Trajectories for Future Distillation.** The ultimate purpose of the on-  
260 line phase, within the EvolveR paradigm, extends beyond solving the immediate task. It is respon-  
261 sible for generating high-quality data for the next cycle of offline reflection. Because the agent’s  
262 actions are guided by proven principles, the resulting trajectories,  $\tau_{\text{new}}$ , are not random walks but  
263 are instead rich recordings of structured, experience-guided problem-solving. These trajectories  
264 capture the interplay between distilled principles, internal reasoning, and external tool use (e.g.,  
265 `<search_knowledge>`), and serve as valuable input for the offline phase, enabling EvolveR to refine  
266 existing principles and discover more effective strategies in a virtuous cycle.

### 267 3.3 POLICY EVOLUTION: CLOSING THE LOOP WITH REINFORCEMENT LEARNING

268 To enable the agent to learn from its actions and evolve its policy  $\pi_\theta$ , we employ a reinforcement  
269 learning framework. The learning process is guided by a composite reward function and a policy  
optimization algorithm that leverages the trajectories collected during the online phase.

270 **Reward Function.** We design a composite reward function  $R(\tau)$  for a given trajectory  $\tau$  that  
 271 balances task success with procedural correctness. It is a weighted sum of an outcome reward and a  
 272 format reward:  $R(\tau) = w_o R_{\text{outcome}}(\tau) + w_f R_{\text{format}}(\tau)$ .  
 273

- 274 • **Outcome Reward**  $R_{\text{outcome}}$ , is a sparse, binary reward based on the final answer’s correctness.  
 275 Following prior work, it is determined by an exact match with the ground truth:

$$R_{\text{outcome}}(\tau) = \text{EM}(a_{\text{pred}}, a_{\text{gold}}) \quad (3)$$

277 where  $a_{\text{pred}}$  is the final answer extracted from the trajectory  $\tau$  and  $a_{\text{gold}}$  is the ground truth  
 278 answer.  
 279

- 280 • **Format Reward**  $R_{\text{format}}$ , is a dense shaping reward that evaluates the quality of the rea-  
 281 soning process. Let  $N_{\text{think}}(\tau)$ ,  $N_{\text{exp}}(\tau)$  and  $N_{\text{know}}(\tau)$  denote the counts of valid `<think>`,  
 282 `<search_experience>` and `<search_knowledge>` actions within  $\tau$ .  $R_{\text{format}}$  is composed of a  
 283 think score  $R_{\text{think}}$ , rewarding a balanced number of reasoning steps, and a search score  $R_{\text{search}}$   
 284 promoting search experience and knowledge. The final format reward is calculated as:

$$R_{\text{format}}(\tau) = \mathbb{I}(\tau_{\text{complete}}) \cdot \frac{R_{\text{think}}(\tau) + R_{\text{search}}(\tau)}{2} \quad (4)$$

285 where  $\mathbb{I}(\tau_{\text{complete}})$  is an indicator function that is 1 only if the trajectory contains at least one  
 286 of each required action type (`<think>`, any search, and `<answer>`), and 0 otherwise. This  
 287 ensures that only structurally complete trajectories receive a format reward.  
 288

### 291 Policy Optimization.

292 The policy  $\pi_\theta$  is updated using the collected tra-  
 293 jectories. We utilize Group Relative  
 294 Policy Optimization (GRPO) (Shao  
 295 et al., 2024), which balances the  
 296 optimization stability and efficiency by  
 297 using the average reward of multi-  
 298 ple sampled trajectories as a baseline,  
 299 thus avoiding the need for a learned  
 300 value function. Specifically, for each  
 301 input, we sample a group of  $G$  trajec-  
 302 tories. The policy is then optimized  
 303 by maximizing the following objective function:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{\tau \in \mathcal{D}} \left[ \sum_{t=1}^{|\tau|} \min \left( \rho_t(\theta) \hat{A}_t, \text{clip}(\rho_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t \right) - \beta D_{\text{KL}}[\pi_\theta || \pi_{\text{ref}}] \right] \quad (5)$$

307 where  $\rho_t(\theta) = \frac{\pi_\theta(a_t | h_t)}{\pi_{\text{old}}(a_t | h_t)}$  is the importance sampling ratio,  $\hat{A}_t$  is the advantage estimate, and the  
 308 final term is a KL-divergence penalty to constrain policy updates.  
 309

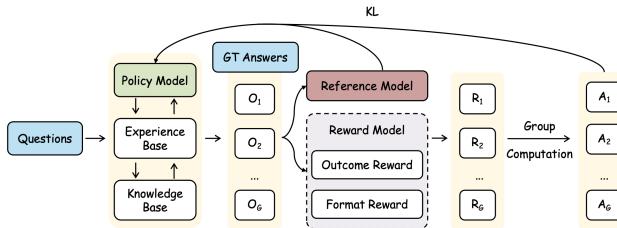
310 Crucially, this optimization process is deeply integrated with our experience lifecycle. As the agent’s  
 311 actions during the online phase are conditioned on the principles  $\mathcal{P}_k$  retrieved from its experience  
 312 base, the trajectories collected in  $\mathcal{D}$  are inherently experience-guided. Consequently, the GRPO  
 313 update does not merely learn a generic reasoning policy. Instead, it explicitly learns a policy of how  
 314 to effectively utilize its own distilled wisdom to generate successful outcomes. The optimization  
 315 process, therefore, reinforces the valuable connections between retrieving high-quality principles  
 316 and producing high-reward trajectories, successfully closing the learning loop.

## 317 4 EXPERIMENTS

### 319 4.1 EXPERIMENTAL IMPLEMENTATION DETAILS

#### 321 4.1.1 TASKS AND DATASETS

323 To comprehensively evaluate the EvolveR paradigm, we assess its performance on seven question-  
 324 answering benchmarks, encompassing both in-domain and out-of-domain datasets. Following prior



325 Figure 3: Policy model update optimization algorithm of  
 326 EvolveR.

324 work (Jin et al., 2025; Mei et al., 2025), the in-domain datasets, whose training splits are used  
 325 to build the experience base, include Natural Questions (NQ) (Kwiatkowski et al., 2019) and the  
 326 multi-hop benchmark HotpotQA (Yang et al., 2018). The out-of-domain datasets, used exclusively  
 327 for evaluating generalization, encompass the general QA benchmarks TriviaQA (Joshi et al., 2017)  
 328 and PopQA (Mallen et al., 2022), as well as the more complex multi-hop challenges 2WikiMulti-  
 329 HopQA (Ho et al., 2020), Musique (Trivedi et al., 2022b), and Bamboogle (Press et al., 2022).

#### 330 4.1.2 BASELINE METHODS

331 Following prior works, we compare against a comprehensive suite of baselines built upon the  
 332 Qwen2.5 foundational models. The baselines represent three primary paradigms. First, prompting-  
 333 based methods, which require no parameter updates, include Direct Inference, Chain-of-Thought  
 334 (CoT) (Wei et al., 2022), Retrieval-Augmented Generation (RAG) (Lewis et al., 2020), and advanced  
 335 variants like IRCoT (Trivedi et al., 2022a) and Searcher-01 (Li et al., 2025). Second, Supervised Fine-  
 336 Tuning (SFT) methods represent approaches that learn from static expert data, including standard  
 337 SFT (Chung et al., 2024) and Rejection Sampling (Ahn et al., 2024). Finally, the most direct com-  
 338 petitors are RL methods, against which we benchmark extensively. This category is primarily com-  
 339 posed of Searcher-R1 (Jin et al., 2025), DeepSeek-R1 (Guo et al., 2025), which are also trained with  
 340 trajectory-level feedback. Specifically, DeepSeek-R1 performs reasoning and answer steps without  
 341 a search engine, whereas Searcher-R1 incorporates an external local or web search engine. Together,  
 342 these baselines provide a challenging evaluation landscape for our proposed paradigm.

#### 343 4.1.3 EVALUATION METRICS

344 To ensure a direct and fair comparison with prior work in our main results, our primary evaluation  
 345 metric is Exact Match (EM), a strict measure that requires the predicted answer to exactly match  
 346 the ground truth after standard normalization. We also report the F1 Score in the analysis of model  
 347 scales’ generalizability, which provides a more comprehensive and robust measure of performance,  
 348 particularly since ground truths may contain multiple valid answers or aliases.

#### 349 4.1.4 IMPLEMENTATION DETAILS

350 Our experiments are conducted on the Qwen2.5 model family. Inspired by DeepSeek-R1 (Guo et al.,  
 351 2025), we introduce a cold-start stage to stabilize early RL training by first fine-tuning the base  
 352 model on a small, curated dataset of CoT interaction trajectories. Following the setup of Searcher-  
 353 R1, we construct this dataset from approximately 700 samples from the NQ and HotpotQA training  
 354 sets. We utilize the LLama.Factory (Zheng et al., 2024) to fine-tune the model with LoRA. For the  
 355 agent evolution phase, we employ GRPO for optimization. At each RL step, we sample a batch  
 356 of 128 prompts, generating  $G = 8$  trajectories for each. The agent is then updated, again using  
 357 Adam, but with a reduced learning rate of  $1 \times 10^{-6}$ , a warm-up step of 20 and a mini-batch size  
 358 of 128. All training is conducted on 8 A100 GPUs, leveraging the Verl framework<sup>1</sup> for efficient  
 359 implementation. We will show more details in Appendix 4.1.

### 360 4.2 MAIN RESULTS

361 The main results of our evaluation are presented in Table 1. Our analysis focuses on the comprehensive  
 362 evaluation conducted on the Qwen2.5-3B and 7B models (we will show more results of  
 363 different model scales in the 5.1). **EvolveR achieves superior average scores 0.382 for 3B and 0.417**  
 364 **for 7B**, outperforming all baselines, including strong RL agents like Searcher-R1. This robust over-  
 365 all performance is not driven by a narrow specialty, but by consistent, top-tier results across a wide  
 366 spectrum of tasks; it secures the best scores on diverse benchmarks, including the in-domain NQ,  
 367 the out-of-domain PopQA, and the adversarial Bamboogle dataset, while remaining highly com-  
 368 petitive on all others. This consistent, high-level performance across diverse benchmarks validates  
 369 that by systematically distilling, managing and utilizing, agents can develop more generalizable and  
 370 powerful problem-solving strategies.

## 371 5 FURTHER ANALYSIS

### 372 5.1 ANALYSIS OF MODEL SCALES GENERALIZABILITY

373  
 374 <sup>1</sup><https://github.com/volcengine/verl>

378

379  
380  
Table 1: Main results on QA benchmarks. The best performance in each column is set in **bold**. Our  
proposed model, EvolveR, is highlighted in gray.

Methods	In domain			Out of domain				Avg.
	NQ	HotpotQA	TriviaQA	PopQA	2wiki	Musique	Bamboogle	
<b>Qwen2.5-3B</b>								
Direct Inference	0.106	0.149	0.288	0.108	0.244	0.020	0.024	0.134
CoT	0.023	0.021	0.032	0.005	0.021	0.002	0.000	0.015
IRCoT	0.111	0.164	0.312	0.200	0.171	0.067	0.240	0.181
Search-o1	0.238	0.221	0.472	0.262	0.218	0.054	0.320	0.255
RAG	0.348	0.255	0.544	0.387	0.226	0.047	0.080	0.270
SFT	0.249	0.186	0.292	0.104	0.248	0.044	0.112	0.176
R1-base	0.226	0.201	0.455	0.173	0.268	0.055	0.224	0.229
R1-instruct	0.210	0.208	0.449	0.171	0.275	0.060	0.192	0.224
Rejection Sampling	0.294	0.240	0.488	0.332	0.233	0.059	0.210	0.265
Search-R1-base	0.406	0.284	<b>0.587</b>	0.435	0.273	0.049	0.088	0.303
Search-R1-instruct	0.341	0.324	0.545	0.378	0.319	0.103	0.264	0.325
EvolveR (ours)	<b>0.434</b>	<b>0.373</b>	0.584	0.434	<b>0.381</b>	<b>0.137</b>	<b>0.328</b>	<b>0.382</b>
<b>Qwen2.5-7B</b>								
Direct Inference	0.134	0.183	0.408	0.140	0.250	0.031	0.120	0.181
CoT	0.048	0.092	0.185	0.054	0.111	0.022	0.232	0.106
IRCoT	0.224	0.133	0.478	0.301	0.149	0.072	0.224	0.239
Search-o1	0.151	0.187	0.443	0.131	0.176	0.058	0.296	0.206
RAG	0.349	0.299	0.585	0.392	0.235	0.058	0.208	0.304
SFT	0.318	0.217	0.354	0.121	0.259	0.066	0.112	0.207
R1-base	0.297	0.242	0.539	0.202	0.273	0.083	0.296	0.276
R1-instruct	0.270	0.237	0.537	0.199	0.292	0.072	0.293	0.271
Rejection Sampling	0.360	0.331	0.592	0.380	0.296	0.123	0.355	0.348
Search-R1-instruct	0.393	0.370	0.610	0.397	<b>0.414</b>	0.146	0.368	0.385
EvolveR (ours)	<b>0.462</b>	<b>0.411</b>	<b>0.620</b>	<b>0.473</b>	0.395	<b>0.168</b>	<b>0.392</b>	<b>0.417</b>

405

406

407 To validate that our EvolveR frame-  
408 work is a generalizable paradigm  
409 rather than a method tailored to a spe-  
410 cific model size, we evaluated its per-  
411 formance across a spectrum of open-  
412 source model scales. As presented  
413 in Figure 4, we applied EvolveR to  
414 Qwen2.5 models of 0.5B, 1.5B, **3B**  
415 and **7B parameters**. The results reveal  
416 a clear and consistent positive trend:  
417 as the parameter count of the base  
418 model increases, the performance of  
419 the EvolveR agent improves signif-  
420 icantly on every benchmark. The  
421 average performance rises monoton-  
422 ically from 0.150 on the 0.5B model  
423 to 0.270 on the 1.5B model, **0.382 on the 3B model, and further to 0.417 on the 7B model**. This scaling behavior demonstrates that our experience-driven life-  
424 cycle effectively harnesses the superior reasoning and instruction-following capabilities inherent in  
425 larger foundational models. It confirms that EvolveR acts as a synergistic layer on top of the base  
426 model, and suggests that its performance will continue to improve with future advancements in the  
427 open-source LLM landscape.

428

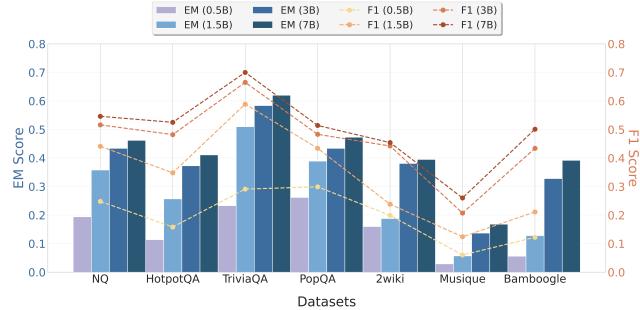
## 5.2 ABLATION STUDIES: DISSECTING THE EVOLVER FRAMEWORK

429

### 5.2.1 VALIDATING THE SELF-DISTILLATION MECHANISM

430

431 A central claim of our work is that an agent can learn effectively through self-distillation. To rigorously investigate this, we compare our standard EvolveR (self-distill) against a strong

432  
Figure 4: Performance of EvolveR across various model  
433 scales.

432  
 433 Table 2: Validating the self-distillation mechanism. We compare our EvolveR, which uses its own  
 434 model for distillation, against a variant that uses a larger, external model (GPT-4o-mini).

435 436 437 Model Variant	438 In domain			439 Out of domain				440 441 442 443 444 445 Avg.
	NQ	446 HotpotQA	447 TriviaQA	448 PopQA	449 2wiki	450 Musique	451 Bamboogle	
<b>452 Qwen2.5-0.5B</b>								
453 EvolveR (self-distill)	0.194	0.114	0.233	0.262	0.160	0.029	0.056	0.150
454 EvolveR (teacher-distill)	0.281 ↑	0.193 ↑	0.402 ↑	0.363 ↑	0.202 ↑	0.033 ↑	0.064 ↑	0.220 ↑
<b>455 Qwen2.5-1.5B</b>								
456 EvolveR (self-distill)	0.358	0.257	0.510	0.389	0.188	0.057	0.128	0.270
457 EvolveR (teacher-distill)	0.352 ↓	0.259 ↑	0.503 ↓	0.395 ↑	0.207 ↑	0.072 ↑	0.240 ↑	0.290 ↑
<b>458 Qwen2.5-3B</b>								
459 EvolveR (self-distill)	0.434	0.373	0.584	0.434	0.381	0.137	0.328	0.382
460 EvolveR (teacher-distill)	0.421 ↓	0.372 ↓	0.583 ↓	0.359 ↓	0.437 ↑	0.127 ↓	0.288 ↓	0.370 ↓

461 alternative, EvolveR (teacher-distill), which uses the powerful GPT-4o-mini as an external  
 462 model for experience distillation.

463 The results, presented in Table 2, reveal a nuanced, scale-dependent relationship. For smaller models  
 464 like the 0.5B variant, the stronger external teacher provides a clear benefit, as the base model’s own  
 465 distillation capabilities are limited. However, as the model scales to 3B, a reversal occurs: our  
 466 EvolveR (self-distill) (0.382 avg.) outperforms the teacher-guided variant (0.370 avg.).  
 467 This is a critical finding, suggesting that as an agent’s own reasoning becomes more sophisticated,  
 468 principles distilled from its own internal policy are ultimately more effective due to better “cognitive  
 469 alignment”. This validates self-distillation as a core, scaling strength of the EvolveR paradigm.

### 470 5.2.2 THE ROLE OF EXPERIENCE RETRIEVAL

471 To quantify the direct benefit of providing the agent with access to its distilled principles at inference  
 472 time. To achieve this, we compare our full EvolveR model against an ablated variant, EvolveR  
 473 w/o exp-retrieve. It is critical to note that both models undergo the identical experience-  
 474 driven RL training process. The sole difference is that the w/o exp-retrieve variant is denied  
 475 access to the experience base during evaluation.

476 The results in Table 3 show a stark performance degradation across all model scales when experience  
 477 retrieval is disabled. For the 3B model, for instance, the average performance drops significantly  
 478 from 0.382 to 0.340. This substantial gap underscores a key finding: an agent trained with our  
 479 EvolveR framework, while powerful on its own, achieves its full potential only when it can access  
 480 and condition on the relevant principles from its past. This demonstrates that experience retrieval is  
 481 a critical and indispensable component of the EvolveR paradigm for optimal performance.

482 Table 3: Investigating the role of experience retrieval at inference time. The w/o exp-retrieve  
 483 variant uses the same model but is not allowed to access the experience base during evaluation.

484 485 Model Variant	486 In domain			487 Out of domain				488 489 490 491 492 493 Avg.
	NQ	494 HotpotQA	495 TriviaQA	496 PopQA	497 2wiki	498 Musique	499 Bamboogle	
<b>499 Qwen2.5-0.5B</b>								
500 EvolveR	0.194	0.114	0.233	0.262	0.160	0.029	0.056	0.150
501 EvolveR w/o exp-retrieve	0.085 ↓	0.065 ↓	0.137 ↓	0.150 ↓	0.082 ↓	0.013 ↓	0.016 ↓	0.078 ↓
<b>502 Qwen2.5-1.5B</b>								
503 EvolveR	0.358	0.257	0.510	0.389	0.188	0.057	0.128	0.270
504 EvolveR w/o exp-retrieve	0.136 ↓	0.112 ↓	0.218 ↓	0.160 ↓	0.136 ↓	0.019 ↓	0.080 ↓	0.123 ↓
<b>505 Qwen2.5-3B</b>								
506 EvolveR	0.434	0.373	0.584	0.434	0.381	0.137	0.328	0.382
507 EvolveR w/o exp-retrieve	0.405 ↓	0.343 ↓	0.569 ↓	0.392 ↓	0.334 ↓	0.100 ↓	0.240 ↓	0.340 ↓

### 508 5.3 EXPLORING THE INFLUENCE OF EXPERIENCE INTERNALIZATION

486 In our proposed framework, all retrieved information (both from the external knowledge base (<information>)  
 487 and our internal experience base (<experience>)) is treated as context, with loss  
 488 masked during the model update phase. A natural question arises from this design: while it is  
 489 sensible to avoid learning the content of transient external documents, could the agent benefit from  
 490 directly absorbing its own distilled wisdom into its parameters?

491 To explore this, we conducted a supplementary experiment on the Qwen2.5-3B model. We created  
 492 a variant, **EvolveR w/ exp-absorb**, where we selectively unmasked the loss for the retrieved  
 493 <experience> tokens, allowing the learning signal to flow through them. Our hypothesis was that  
 494 this might enable the agent to internalise the strategic logic of its principles. The results, presented in  
 495 Table 4, were insightful. The **EvolveR w/ exp-absorb** variant exhibited a slight performance  
 496 degradation compared to our standard approach.

497 We posit that this degradation arises from two related factors. First, noise from irrelevant principles.  
 498 In our current implementation, the agent retrieves a set of top- $k$  principles at each step, not all of  
 499 which may be perfectly relevant, and directly internalising all retrieved principles without a dynamic  
 500 quality filter risks updating the model with noisy or counter-productive signals. Second, a potential  
 501 mismatch in the optimisation objective. Currently, unmasked experience tokens are treated identi-  
 502 cally to generated reasoning steps and optimised via GRPO’s advantage-based loss, while effective  
 503 internalisation might instead require treating high-quality principles as “ground truth” knowledge  
 504 to be memorised. This suggests that a distinct loss formulation, such as likelihood maximisation  
 505 applied specifically to the <experience> block and separate from the RL loss, might be necessary  
 506 to properly absorb this wisdom without destabilising the policy.

507  
 508 Table 4: Ablation study on the experience internalization mechanism. **EvolveR w/o exp-absorb**  
 509 treats principles as external context by masking gradients during backpropagation.

510 511 Model Variant	512 In domain				513 Out of domain			514 Avg.
	NQ	515 HotpotQA	516 TriviaQA	517 PopQA	2wiki	Musique	Bamboogle	
518 EvolveR	0.434	0.373	0.584	0.434	0.381	0.137	0.328	0.382
519 EvolveR w exp-absorb	0.433 ↓	0.369 ↓	0.583 ↓	0.435 ↑	0.376 ↓	0.124 ↓	0.280 ↓	0.371 ↓

## 520 6 CONCLUSION

521 In this work, we introduced **EvolveR**, a novel paradigm for self-evolving LLM agents centered on  
 522 a complete, closed-loop experience lifecycle. Our extensive experiments demonstrate the effec-  
 523 tiveness of this approach, showing that EvolveR consistently and significantly outperforms a wide  
 524 range of strong baseline methods on a comprehensive suite of QA benchmarks. Furthermore, our  
 525 detailed ablation studies rigorously validate the core tenets of our framework, confirming the sig-  
 526 nificant value of the agent’s self-distilled experiences and demonstrating the high efficacy of the  
 527 self-distillation mechanism itself. While the quality of distilled principles is inherently tied to the  
 528 base model’s capabilities, pointing to promising future work, EvolveR provides a concrete blueprint  
 529 for agents that learn from the consequences of their own experiences, shifting the focus from merely  
 530 accessing knowledge to actively building and evolving expertise.

540 REFERENCES  
541

542 Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. Large language models  
543 for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*, 2024.

544 John R Anderson. Problem solving and learning. *American psychologist*, 48(1):35, 1993.  
545

546 Huan ang Gao, Jiayi Geng, Wenyue Hua, Mengkang Hu, Xinzhe Juan, Hongzhang Liu, Shilong  
547 Liu, Jiahao Qiu, Xuan Qi, Yiran Wu, Hongru Wang, Han Xiao, Yuhang Zhou, Shaokun Zhang,  
548 Jiayi Zhang, Jinyu Xiang, Yixiong Fang, Qiwen Zhao, Dongrui Liu, Qihan Ren, Cheng Qian,  
549 Zhenhailong Wang, Minda Hu, Huazheng Wang, Qingyun Wu, Heng Ji, and Mengdi Wang. A  
550 survey of self-evolving agents: On path to artificial super intelligence, 2025. URL <https://arxiv.org/abs/2507.21046>.  
551

552 Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gian-  
553 inazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczek, et al. Graph of  
554 thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI  
555 conference on artificial intelligence*, volume 38, pp. 17682–17690, 2024.  
556

557 Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding:  
558 Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge dis-  
559 tillation. *arXiv preprint arXiv:2402.03216*, 2024.  
560

561 Liting Chen, Lu Wang, Hang Dong, Yali Du, Jie Yan, Fangkai Yang, Shuang Li, Pu Zhao, Si Qin,  
562 Saravan Rajmohan, et al. Introspective tips: Large language model for in-context decision mak-  
563 ing. *arXiv preprint arXiv:2305.11598*, 2023.  
564

565 Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building  
566 production-ready ai agents with scalable long-term memory, 2025. URL <https://arxiv.org/abs/2504.19413>.  
567

568 Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li,  
569 Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned lan-  
570 guage models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.  
571

572 Xuanwen Ding, Jie Zhou, Liang Dou, Qin Chen, Yuanbin Wu, Chengcui Chen, and Liang He. Boost-  
573 ing large language models with continual learning for aspect-based sentiment analysis, 2024.  
574 URL <https://arxiv.org/abs/2405.05496>.  
575

576 Timo Flesch, Jan Balaguer, Ronald Dekker, Hamed Nili, and Christopher Summerfield. Comparing  
577 continual task learning in minds and machines. *Proceedings of the National Academy of Sciences*,  
578 115(44):E10313–E10322, 2018.  
579

580 Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu,  
581 Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms  
582 via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.  
583

584 Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop  
585 qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*,  
586 2020.  
587

588 Tianyu Huai, Jie Zhou, Yuxuan Cai, Qin Chen, Wen Wu, Xingjiao Wu, Xipeng Qiu, and Liang He.  
589 Task-core memory management and consolidation for long-term continual learning, 2025a. URL  
590 <https://arxiv.org/abs/2505.09952>.  
591

592 Tianyu Huai, Jie Zhou, Xingjiao Wu, Qin Chen, Qingchun Bai, Ze Zhou, and Liang He. Cl-moe: En-  
593 hancing multimodal large language model with dual momentum mixture-of-experts for continual  
594 visual question answering, 2025b. URL <https://arxiv.org/abs/2503.00413>.  
595

596 Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and  
597 Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement  
598 learning, 2025. URL <https://arxiv.org/abs/2503.09516>.  
599

594 Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly  
 595 supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.  
 596

597 James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, An-  
 598 drewi A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis  
 599 Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic  
 600 forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):  
 601 3521–3526, March 2017. ISSN 1091-6490. doi: 10.1073/pnas.1611835114. URL <http://dx.doi.org/10.1073/pnas.1611835114>.  
 602

603 Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris  
 604 Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a  
 605 benchmark for question answering research. *Transactions of the Association for Computational  
 606 Linguistics*, 7:453–466, 2019.

607 Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal,  
 608 Heinrich Kütter, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented gener-  
 609 ation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:  
 610 9459–9474, 2020.

611 Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and  
 612 Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. *arXiv preprint  
 613 arXiv:2501.05366*, 2025.

614 Junyu Luo, Weizhi Zhang, Ye Yuan, Yusheng Zhao, Junwei Yang, Yiyang Gu, Bohan Wu, Binqi  
 615 Chen, Ziyue Qiao, Qingqing Long, Rongcheng Tu, Xiao Luo, Wei Ju, Zhiping Xiao, Yifan Wang,  
 616 Meng Xiao, Chenwu Liu, Jingyang Yuan, Shichang Zhang, Yiqiao Jin, Fan Zhang, Xian Wu,  
 617 Hanqing Zhao, Dacheng Tao, Philip S. Yu, and Ming Zhang. Large language model agent: A sur-  
 618 vey on methodology, applications and challenges, 2025. URL <https://arxiv.org/abs/2503.21460>.  
 619

620 Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi.  
 621 When not to trust language models: Investigating effectiveness of parametric and non-parametric  
 622 memories. *arXiv preprint arXiv:2212.10511*, 2022.

623 Jianbiao Mei, Tao Hu, Daocheng Fu, Licheng Wen, Xuemeng Yang, Rong Wu, Pinlong Cai, Xinyu  
 624 Cai, Xing Gao, Yu Yang, Chengjun Xie, Botian Shi, Yong Liu, and Yu Qiao. O<sup>2</sup>-searcher:  
 625 A searching-based agent model for open-domain open-ended question answering, 2025. URL  
 626 <https://arxiv.org/abs/2505.16582>.  
 627

628 German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual  
 629 lifelong learning with neural networks: A review. *Neural networks*, 113:54–71, 2019.

630 Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring  
 631 and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*,  
 632 2022.

633 Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan  
 634 Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang,  
 635 Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin  
 636 Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li,  
 637 Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang,  
 638 Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.  
 639 URL <https://arxiv.org/abs/2412.15115>.  
 640

641 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,  
 642 Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathe-  
 643 matical reasoning in open language models, 2024. URL <https://arxiv.org/abs/2402.03300>.  
 644

645 Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugging-  
 646 gpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information  
 647 Processing Systems*, 36:38154–38180, 2023.

648 Yaorui Shi, Sihang Li, Chang Wu, Zhiyuan Liu, Junfeng Fang, Hengxing Cai, An Zhang, and Xiang  
 649 Wang. Search and refine during think: Autonomous retrieval-augmented reasoning of llms, 2025.  
 650 URL <https://arxiv.org/abs/2505.11277>.  
 651

652 Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and  
 653 Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning, 2023. URL  
 654 <https://arxiv.org/abs/2303.11366>.  
 655

656 Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving re-  
 657 trieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv*  
 658 preprint *arXiv:2212.10509*, 2022a.  
 659

660 Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop  
 661 questions via single-hop question composition. *Transactions of the Association for Computational*  
*Linguistics*, 10:539–554, 2022b.  
 662

663 Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual  
 664 learning: Theory, method and application, 2024. URL <https://arxiv.org/abs/2302.00487>.  
 665

666 Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny  
 667 Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in*  
*668 neural information processing systems*, 35:24824–24837, 2022.  
 669

670 Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen  
 671 Ma, Hinrich Schütze, Volker Tresp, and Yunpu Ma. Memory-r1: Enhancing large language  
 672 model agents to manage and utilize memories via reinforcement learning. *arXiv preprint*  
*arXiv:2508.19828*, 2025.  
 673

674 Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov,  
 675 and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question  
 676 answering. *arXiv preprint arXiv:1809.09600*, 2018.  
 677

678 Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik  
 679 Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Ad-*  
*680 vances in neural information processing systems*, 36:11809–11822, 2023a.  
 681

682 Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao.  
 683 React: Synergizing reasoning and acting in language models, 2023b. URL <https://arxiv.org/abs/2210.03629>.  
 684

685 Guibin Zhang, Muxin Fu, Guancheng Wan, Miao Yu, Kun Wang, and Shuicheng Yan. G-memory:  
 686 Tracing hierarchical memory for multi-agent systems, 2025. URL <https://arxiv.org/abs/2506.07398>.  
 687

688 Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm  
 689 agents are experiential learners, 2024. URL <https://arxiv.org/abs/2308.10144>.  
 690

691 Yaowei Zheng, Richong Zhang, Junhao Zhang, YeYanhan YeYanhan, and Zheyuan Luo. LlamaFac-  
 692 tory: Unified efficient fine-tuning of 100+ language models. In Yixin Cao, Yang Feng, and  
 693 Deyi Xiong (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Compu-*  
*694 tational Linguistics (Volume 3: System Demonstrations)*, pp. 400–410, Bangkok, Thailand, August  
 695 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-demos.38. URL  
*696 https://aclanthology.org/2024.acl-demos.38*.  
 697

698 Huichi Zhou, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee, Guchun  
 699 Zhang, Kun Shao, Linyi Yang, and Jun Wang. Memento: Fine-tuning llm agents without fine-  
 700 tuning llms, 2025. URL <https://arxiv.org/abs/2508.16153>.  
 701

## A APPENDIX

## A.1 USE OF LARGE LANGUAGE MODELS

The research presented in this paper, including the core ideas, experimental design, and quantitative results, is the original work of the authors. A large language model was used as a writing assistant for tasks such as polishing prose, improving clarity, and correcting grammatical errors in the manuscript. All final content was reviewed and edited by the authors to ensure it accurately reflects our research and contributions.

## A.2 EXPERIMENTAL IMPLEMENTATION DETAILS

We provide a comprehensive list of hyperparameters and implementation details used in our experiments to ensure full reproducibility.

**General Setup.** Across all experiments, we use models from the Qwen2.5 family (Qwen et al., 2025) with their corresponding tokenizers. The maximum sequence length is set to 8192 tokens for all inputs, and the maximum response sequence length is set to 1024 tokens. The GPT-4o-mini model is used as the teacher model in the corresponding ablation study. We use BGE-M3 (Chen et al., 2024) as our embedding model.

**Cold-start Stage.** This initial SFT stage is designed solely to teach the model the required interaction format (e.g., producing well-structured `<think>` and `<search>` actions). The model is trained for 3 epochs using the Adam optimizer, with an initial learning rate of  $1 \times 10^{-4}$ , a warm-up ratio of 0.1, and a batch size of 16.

**Online Interaction Phase.** For each `<search_knowledge>` action, we retrieve the top- $k_d = 3$  documents from the external knowledge base, following the prior work (Jin et al., 2025). Similarly, for each `<search_experience>` action, we retrieve the top- $k_e = 3$  principles from the experience base  $\mathcal{E}$ .

**Offline Distill Phase.** The self-distill mechanism utilizes the agent’s own policy model  $\pi_\theta$  to distill principles. The temperature is set to 1 during this phase. For the deduplication and integration process, we first use a semantic similarity pre-filter with a threshold of  $\theta_{\text{sim}} = 0.85$  before passing candidates to the LLM-based equivalence check. The periodic principle sweep removes any principle from  $\mathcal{E}$  whose `metric_score` falls below the pruning threshold of  $\theta_{\text{prune}} = 0.3$ .

**Reward Function Details.** As described in the Section 3.3, the Format Reward is an average of a think score and a search score. We detail their specific calculation here. The think score  $R_{\text{think}}$  is determined by a discrete mapping based on the number of `<think>` actions,  $N_{\text{think}}$ : it scales from 0.2 (for  $N_{\text{think}} = 1$ ) to a maximum of 1.0 (for  $N_{\text{think}} = 6$ ), and is capped at 0.5 for excessive reasoning ( $N_{\text{think}} > 8$ ) to encourage conciseness. The search score  $R_{\text{search}}$  is the sum of a diversity score and a quantity bonus. The diversity score is 0.5 if both `<search_experience>` and `<search_knowledge>` are used, 0.2 if only one type is used, and 0 otherwise. A quantity bonus of 0.1 is added for each additional search action beyond the first, up to a maximum bonus of 0.5 (for a total of 6 searches).

**Policy Optimization.** The composite reward function is weighted with  $w_o = 1.0$  for the outcome reward and  $w_f = 0.1$  for the format reward. For the GRPO objective function (Equation 5), the clipping parameter is set to  $\epsilon = 0.2$  and the KL-divergence coefficient is  $\beta = 0.001$ . During the training procedure, we adopt vLLM to accelerate LLM rollouts. The tensor parallel size is set to 1, and the GPU memory utilization ratio is set at 0.6. For rollout sampling, we use a temperature of 1.0 and a top-p value of 0.95.

**Computational Cost Analysis.** We provide detailed computational resources required for training and experience retrieval latency to demonstrate the efficiency of EvolveR.

756 • **Training Cost:** The full training lifecycle for the Qwen2.5-3B model, which includes the cold-  
 757 start SFT phase and the subsequent RL policy evolution via GRPO, requires approximately  
 758 **39.4 hours** on a server equipped with 8 NVIDIA A100 (80GB) GPUs.  
 759

760 • **Experience Retrieval Latency:** A key concern for retrieval-augmented systems is the added  
 761 latency during inference. We implement the Experience Base using Milvus for efficient vector  
 762 similarity search. Our empirical measurements show that even with an experience base  
 763 containing approximately **14,000 principles**, the latency for retrieving the top-3 relevant prin-  
 764 ciples is approximately **0.06 seconds**. That is imperceptible compared to LLM generation time,  
 765 ensuring that EvolveR maintains high throughput during deployment.  
 766

767 **SFT-only Baseline Details.** We used the exact same set of successful trajectories generated during  
 768 the Online Interaction stage. Both `<experience>` and `<information>` tokens were masked during  
 769 training. We utilised the LLaMAFactory Zheng et al. (2024) to fine-tune the Qwen2.5-3B model  
 770 using LoRA (Low-Rank Adaptation). The LoRA rank is set to 8, the learning rate is  $1 \times 10^{-4}$ , and the  
 771 model is trained for 5 epochs with a batch size of 1. The warmup ratio is 0.1, the maximum sequence  
 772 length is 4096 tokens, gradient accumulation steps are set to 8, and the learning rate scheduler is  
 773 `Cosine`.  
 774

### 775 A.3 ADDITIONAL EXPERIMENTAL ANALYSIS

#### 776 A.3.1 NECESSITY OF THE RL (GRPO) STAGE

777 To assess the necessity of the Reinforcement Learning stage in EvolveR, we conduct an ablation  
 778 experiment on the Qwen2.5-3B model. Specifically, we reuse the exact same set of successful  
 779 trajectories collected during online interaction, but train the policy via standard SFT rather than  
 780 GRPO. The full SFT hyperparameter configurations are provided in Appendix A.2.  
 781

782 As shown in Table 5, the RL-based variant substantially outperforms the SFT-only version on the  
 783 3B model, achieving a 7% relative improvement. This result highlights the fundamental limitation  
 784 of SFT: it merely encourages the model to reproduce surface-level action sequences from suc-  
 785 cessful trajectories, without understanding the underlying utility or expected reward of actions such as  
 786 `<search.experience>`. In contrast, our RL approach (GRPO) leverages both successful and failed  
 787 trajectories, enabling the agent to learn *what to do* from positive rollouts and *what to avoid* from  
 788 negative ones, which is essential for developing robust retrieval and reasoning strategies.  
 789

790 Table 5: Comparison between SFT-only and RL/GRPO training in EvolveR on the Qwen2.5-3B  
 791 model.  
 792

793 Model Variant	794 In domain				795 Out of domain			796 Avg.
	797 NQ	798 HotpotQA	799 TriviaQA	800 PopQA	801 2wiki	802 Musique	803 Bamboogle	
804 EvolveR(SFT)	805 0.415	806 0.357	807 0.584	808 0.419	809 0.366	810 0.106	811 0.248	812 0.357
813 EvolveR(RL)	814 <b>0.434</b>	815 <b>0.373</b>	816 <b>0.584</b>	817 <b>0.434</b>	818 <b>0.381</b>	819 <b>0.137</b>	820 <b>0.328</b>	821 <b>0.382</b>

#### 822 A.3.2 HYPERPARAMETER SENSITIVITY ANALYSIS

823 To assess the robustness of EvolveR and the impact of the dynamic experience curation mechanism,  
 824 we conducted a sensitivity analysis on the pruning threshold  $\theta_{\text{prune}}$ . This parameter dictates the  
 825 minimum `metric_score` required for a principle to be retained in the Experience Base after  
 826 each cleaning. We evaluated the performance of the EvolveR-1.5B model across a wide range of  
 827 thresholds:  $\theta_{\text{prune}} \in \{0.1, 0.3, 0.7, 0.9\}$ .  
 828

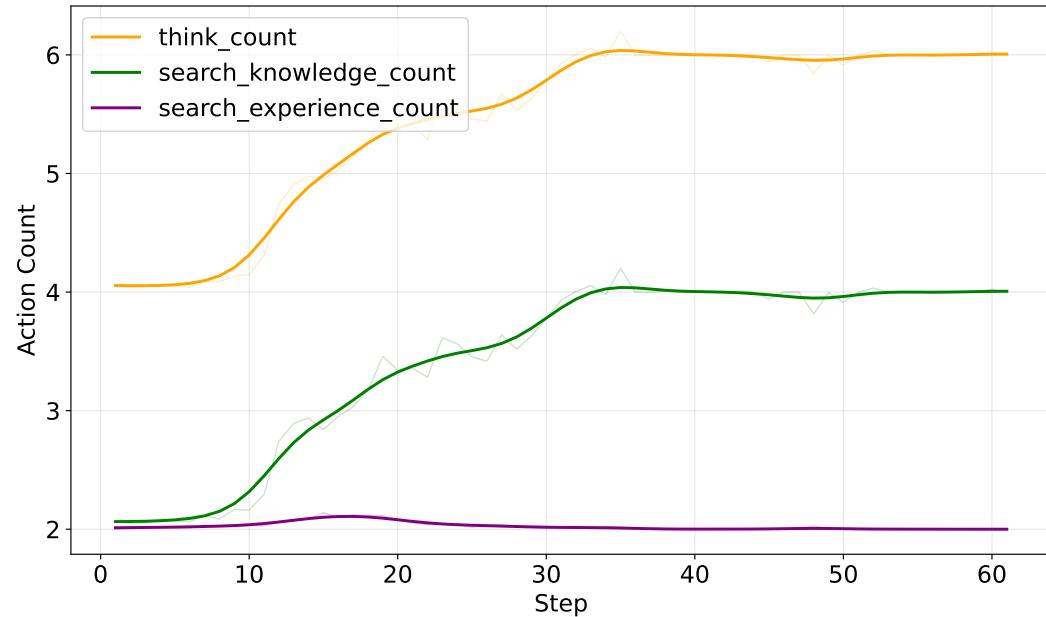
829 The results confirm that performance remains robust across different thresholds. Our default setting  
 830 ( $\theta_{\text{prune}} = 0.3$ ) effectively filters out low-quality principles to prevent the database from growing  
 831 indefinitely.

810 Table 6: Sensitivity analysis of the pruning threshold  $\theta_{\text{prune}}$  on the Qwen2.5-1.5B model.

$\theta_{\text{prune}}$	In domain			Out of domain				Avg.
	NQ	HotpotQA	TriviaQA	PopQA	2wiki	Musique	Bamboogle	
0.1 (loose)	0.336	0.244	0.496	0.387	<b>0.207</b>	0.052	0.160	0.269
0.3 (Default)	<b>0.358</b>	<b>0.257</b>	<b>0.510</b>	<b>0.389</b>	0.188	0.057	0.128	0.270
0.7 (strict)	0.276	0.253	0.498	0.379	0.196	0.060	<b>0.192</b>	0.265
0.9 (very strict)	0.323	<b>0.257</b>	<b>0.510</b>	0.385	0.204	<b>0.061</b>	<b>0.192</b>	<b>0.276</b>

818  
819 A.3.3 LONGITUDINAL ANALYSIS OF LEARNING DYNAMICS  
820821 To provide a deeper understanding of how the EvolveR agent improves over time, we present a  
822 longitudinal analysis of its behavior during the RL training process. We focus on two key aspects:  
823 the evolution of action frequencies and the improvement in the quality of distilled principles.824  
825 **Evolution of Action Frequencies.** We tracked the average number of `<think>`,  
826 `<search_knowledge>` and `<search_experience>` actions per trajectory across training  
827 steps. As shown in Figure 5, the agent’s behavior exhibits distinct phases of optimization:828  
829 

- **Early Phase (Interval 1):** Thanks to the cold-start SFT, the agent begins with reasonable tool  
830 usage capabilities.
- **Optimization Phase (Intervals 2-3):** As RL training progresses, we observe a clear upward  
831 trend in the frequency of `<think>` and `<search_knowledge>`. This indicates that the  
832 agent is learning to engage in deeper reasoning and more extensive external information gath-  
833 ering to solve complex tasks.
- **Convergence Phase (Interval 4):** Crucially, the action counts do not increase indefinitely.  
834 They converge to a stable, efficient range (approx. 2 experience searches, 4 knowledge  
835 searches, and 6 reasoning steps).

836859 Figure 5: Evolution of average action counts per trajectory during RL training.  
860861  
862 **Intrinsic Improvement in Principle Distillation.** To verify that the agent is indeed improving  
863 its ability to distill high-quality principles (rather than just filtering out bad ones), we analyzed  
the quality of principles grouped by their distilled time. We divided the training process into four

864 intervals and calculated the average `metric_score` of the principles generated in each interval  
 865 that remain in the Experience Base.  
 866

867  
868 Table 7: Quality of principles by creation phase.

869 <b>Creation Phase</b>	870 <b>Avg. Metric Score</b>	871 <b>Avg. Usage Count</b>
872 Interval 1	873 0.462	8.22
874 Interval 2	875 0.464	24.10
876 Interval 3	877 0.487	17.82
878 Interval 4	879 <b>0.500</b>	880 12.36

881 Table 7 presents the evolution of principle quality across different training stages. Principles from  
 882 Interval 1 have undergone the longest duration of dynamic pruning, representing a highly filtered  
 883 subset. Conversely, principles from Interval 4 are relatively new and have been subject to less  
 884 filtering. We observe that the principles generated in the final stage achieve a higher average metric  
 885 score (0.500) compared to those from the earliest stage (0.462). Supported by substantial usage  
 886 counts ( $> 12$ ) that ensure statistical stability, this trend indicates that the agent’s intrinsic ability to  
 887 distill high-quality strategies improves over time.  
 888

889  
890 

#### A.4 PROMPT DETAILS

891  
892 

##### A.4.1 COLD START PROMPT

893 The Prompt in Table A.4.4 is used during the cold-start stage to generate the initial trajectories for  
 894 SFT. This prompt guides a powerful LLM (we used GPT-4o) to act as an expert problem-solver,  
 895 producing a small dataset with the right format trajectories to cold start. It is not used to teach the  
 896 cold-start model knowledge. Once this brief cold start phase is completed, the dependency on the  
 897 external model is removed. All subsequent processes (including reasoning with principles, principle  
 898 distillation, and policy updates) are driven solely by the agent itself.  
 899

900  
901 

##### A.4.2 SYSTEM PROMPT

902 The Prompt in Table A.4.4 is the system prompt used by the EvolveR agent during the online interaction  
 903 phase. It defines the agent’s core identity, its available actions (`<think>`, `<search_knowledge>`,  
 904 `<search_experience>`, `<answer>`), and the overall format for its reasoning process.  
 905

906  
907 

##### A.4.3 DISTILL PRINCIPLE PROMPT

908 The Prompt in Table A.4.4 and Table A.4.4 are used during the offline experience self-distillation  
 909 phase to enable the agent’s self-distillation mechanism. Based on the outcome of a trajectory, one  
 910 of two distinct prompts is used to guide the agent’s own model ( $\pi_\theta$ ) to distill a principle. The first  
 911 Prompt is for successful trajectories, focusing on extracting a guiding principle. The second is for  
 912 failed trajectories, aimed at formulating a cautionary principle.  
 913

914  
915 

##### A.4.4 JUDGE SAME PRINCIPLE PROMPT

916 The Prompt in Table A.4.4 is a crucial component of the deduplication and integration process within  
 917 the offline experience self-distillation. It tasks the agent’s own model ( $\pi_\theta$ ) with acting as a semantic  
 918 judge. Given two principles (a newly distilled candidate and a retrieved existing similar one), the  
 919 model is asked to determine if they are semantically equivalent. The binary “yes/no” output of this  
 920 Prompt is used to decide whether to merge a new experience or create a new principle.  
 921

922  
923 

#### A.5 CASE STUDY OF EVOLVER

924  
925 

##### A.5.1 QUALITATIVE ANALYSIS OF DISTILLED PRINCIPLES

926 To provide concrete insight into the nature of the wisdom distilled by EvolveR, we present a qualita-  
 927 tive comparison between a low-scoring (eliminated) principle and a high-scoring (retained) principle  
 928 from our Experience Base.  
 929

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

Table 8: System prompt for LLM agents

---

Answer the given question.

You must conduct reasoning inside `<think>` and `</think>` first every time you get new information or get new experience principles.

After reasoning, you can search for past experiences by `<search_experience>` query `</search_experience>` to get relevant past experience principles (may be guiding or warning principles) and it will return the top searched results between `<experience>` and `</experience>`.

You can use these principles which you think is helpful to help you answer the question.

If you find you lack some knowledge, you can call a search engine by `<search_knowledge>` query `</search_knowledge>` and it will return the top searched results between `<information>` and `</information>`.

You can search knowledge and experience as many times as you want.

If you find no further external knowledge needed, you can directly provide the answer inside `<answer>` and `</answer>`, without detailed illustrations.

For example, `<answer>` Beijing `</answer>`

---

**User: {question}**

---

As shown in Table 13, the distinction is clear. The high-scoring principle provides a precise heuristic for a specific reasoning scenario (character verification), guiding the model on *what to avoid* and *how to verify*. In contrast, the low-scoring principle offers abstract advice about the reasoning process itself (“avoid redundancy”), which lacks the specificity required to guide the agent’s actual actions. This comparison validates that our dynamic scoring mechanism successfully filters for practical utility, retaining wisdom that is grounded and executable.

#### A.5.2 ROLLOUT CASE

We provide a detail rollout case of EvolveR in Table A.5

#### A.6 LIMITATION AND BROADER IMPACT

We acknowledge several limitations and broader implications of our work. The efficacy of our self-distillation mechanism is inherently bounded by the capabilities of the agent’s own model; a less capable model may struggle to distill high-quality principles, thus limiting its evolutionary ceiling. Further research across a broader range of tasks, such as embodied interaction or creative generation, is necessary to fully delineate the boundaries and applicability of the EvolveR paradigm. While our curation mechanisms mitigate experience base growth, ensuring computational efficiency for truly lifelong learning agents also remains an open challenge. Looking forward, the broader impact of this paradigm is significant. On the one hand, EvolveR represents a crucial step towards more autonomous and personalized agents. The explicit nature of its distilled principles also offers a promising avenue for improving interpretability and steerability. On the other hand, this autonomy raises critical safety considerations. An agent that evolves its own principles could develop undesirable strategies if not guided by a robust, value-aligned reward function, necessitating further research into alignment techniques for such self-evolving systems.

## B ETHICS STATEMENT

Our work on self-evolving agents is a foundational research exploration in controlled, simulated environments. We acknowledge that the deployment of such autonomous learning systems in the real world would raise significant safety and alignment challenges, as an agent could potentially develop undesirable strategies. We believe the inherent interpretability of EvolveR’s distilled, human-readable principles offers a promising direction for mitigating these risks through human oversight, and we advocate for further research in alignment with such evolving systems before any high-stakes application is considered. This work does not involve any personally identifiable information or sensitive data.

972  
973  
974  
975  
976  
977

Table 9: Prompt for cold start.

You are a top-notch intelligent reasoning expert, adept at restoring solution paths from given answers and documents in reverse. Your task is to simulate a full reasoning trajectory for answering the question below, based on the provided documents and answer. You must reason step-by-step as if you do not yet know the final answer, even though it is given for supervision.

In `<think>` blocks, do not reference or confirm the final answer directly. Instead, reason like a human—understand the task, recall prior knowledge, evaluate the need for experience or external information, and gradually infer the answer.

The reasoning trajectory must follow the \*\*exact format below\*\*. If the retrieved \*\*experience\*\* alone is sufficient to answer the question\*\*, you may skip the `<search_knowledge>` and `<information>` steps.

**985 Output Format:**

`<think> ... </think>`  
`<search_experience>`

- Retrieve 2–3 relevant abstract experience principles, using structured triple format.
- For each principle, add a short description of its purpose.

`</search_experience>`

`<think>` Explain what you plan to do after retrieving experience. Decide whether you still need to retrieve knowledge. `</think>`

`[IF experience is enough:]`

`<think>`

- List the principles you are applying, include their triple form and description.
- Explain briefly how each principle contributes to your reasoning.
- Continue with reasoning based on these principles and conclude with your final judgment.

`</think>`

`<answer>...</answer>`

`[ELSE:]`

`<search_knowledge>`

- Generate one or more natural language search queries that would help retrieve the provided documents.

`</search_knowledge>`

`<information>`

`{relevant_document}`

`</information>`

`<think>` Reflect on retrieved information. `</think>`

`<think>`

- List the principles you are applying, include their triple form and description.
- Explain how each principle guides the reasoning process using the retrieved information.
- Summarize your reasoning path and justify the answer.

`</think>`

`<Answer>...</Answer>`

**1012 Inputs:**

**1013 Query:** {query}

**1014 Relevant Documents:** {relevant\_document}

**1015 Answer:** {answer}

**1016** Please begin generating the reasoning trajectory.

**1017**

**1018**

**1019**

**1020 C REPRODUCIBILITY STATEMENT**

**1021**

**1022**

**1023**

**1024**

**1025**

To ensure full reproducibility, we will release all code for the EvolveR framework, training scripts and evaluation scripts on GitHub upon publication. Our experiments are conducted on publicly available benchmarks (e.g., NQ, HotpotQA) using open-source Qwen2.5 models. All critical hyperparameters, including those for the SFT cold-start, GRPO optimization and the experience curation

1026

1027 **Table 10: Prompt for summarizing a successful interaction trajectory.**

1028 You are an expert in analyzing interaction logs to distill generalizable wisdom.

1029 Analyze the following successful interaction trajectory. Your goal is to extract a "Guiding Principle" from it.

1031 A "Guiding Principle" has two parts:

1032 1. A concise, one-sentence natural language description. This is the core advice.  
 1033 2. A structured representation of the key steps or logic, as a list of simple (subject, predicate, object)  
 1034 triplets.

1035 **[Trajectory Log]:**1036 `{ {trajectory_log} }`

1037 Final Outcome: SUCCESS

1038

**Your Task:**

1039 Based on the trajectory, generate the Guiding Principle.

1040 First, on a new line, write {DESCRIPTION\_PART\_SEPARATOR}.

1041 Then, write the one-sentence description of the pitfall.

1042 Then, on a new line, write {STRUCTURED\_PART\_SEPARATOR}.

1043 Finally, provide the structured triplets describing the failure pattern in a valid JSON list format.

1044 **[Example]:**1045 `{DESCRIPTION_PART_SEPARATOR}`1046 When a file download fails with a 404 error, do not immediately retry the download; instead, verify  
 1047 the source URL's validity first.1048 `{STRUCTURED_PART_SEPARATOR}`

1049 [

1050     `(file_download, results_in, 404 error),`  
 1051     `(immediate_retry, is, ineffective),`  
 1052     `(correct_action, is, verify URL)`

1053 ]

1054 **[Output]:**

1055

1056 pipeline, are detailed in Appendix A.2. The primary computational requirement is a server with 8  
 1057 A100 GPUs for training the 3B model.

1058

1059

1060

1061

1062

1063

1064

1065

1066

1067

1068

1069

1070

1071

1072

1073

1074

1075

1076

1077

1078

1079

1080

1081

1082

Table 11: Prompt for summarizing a failed interaction trajectory.

1083

You are an expert in analyzing interaction logs to find the root cause of failures.

1084

Analyze the following failed interaction trajectory. Your goal is to extract a "Cautionary Principle" from it.

1085

1086

A "Cautionary Principle" has two parts:

1087

1. A concise, one-sentence description of the key mistake to avoid and under what circumstances.

1088

2. A structured representation of the failure pattern, as a list of simple (subject, predicate, object) triplets.

1089

1090

**[Trajectory Log]:**

1091

`{{trajectory_log}}`

1092

Final Outcome: FAILURE

1093

**Your Task:**

1094

Based on the trajectory, generate the Cautionary Principle.

1095

First, on a new line, write {DESCRIPTION\_PART\_SEPARATOR}.

1096

Then, write the one-sentence description of the pitfall.

1097

Then, on a new line, write {STRUCTURED\_PART\_SEPARATOR}.

1098

Finally, provide the structured triplets describing the failure pattern in a valid JSON list format.

1099

**[Example]:**

1100

`{DESCRIPTION_PART_SEPARATOR}`

1101

When a file download fails with a 404 error, do not immediately retry the download; instead, verify the source URL's validity first.

1102

`{STRUCTURED_PART_SEPARATOR}`

1103

```
[  
    (file download, results_in, 404 error),  
    (immediate_retry, is, ineffective),  
    (correct_action, is, verify URL)  
]
```

1108

**[Output]:**

1109

1110

1111

Table 12: Prompt for Principle Similarity Analysis.

1112

You are a semantic analysis expert. Determine if two principles describe the same core idea, even if they use different words.

1113

Principle A: "{summary}"

1114

Principle B: "{existing-principle-description}"

1115

Do Principle A and Principle B describe the same essential advice or warning?

1116

Please answer with only "Yes" or "No".

1117

1118

1119

1120

1121

Table 13: Qualitative comparison of principles.

1122

1123

Type	Score	Principle Content	Analysis
Low Score (Eliminated)	0.25	"When using multiple principles, ensure there is no redundancy or unnecessary overlapping, leading to confusion about the principle applicability and efficiency."	<b>Vague:</b> This is a generic meta-statement about the process, not a specific strategy for solving the task. It offers no actionable guidance.
High Score (Retained)	0.80	"When determining character relationships in a series, avoid assuming actors based solely on character recognition; confirm with verified role descriptions."	<b>Specific &amp; Actionable:</b> This principle identifies a specific pitfall (hallucinating actors) and provides a concrete correction strategy.

1134

1135

1136

1137

1138

1139

1140

1141

1142

1143

1144

1145

1146

1147

1148

1149

1150

1151

1152

1153

1154

1155

1156

1157

1158

1159

1160

1161

1162

1163

1164

1165

1166

1167

1168

1169

1170

1171

1172

1173

1174

1175

1176

1177

1178

1179

1180

1181

1182

1183

1184

1185

1186

1187

1188

Table 14: A rollout case of EvolveR

**System:** System\_Prompt**Question:** who was the pinkerton detective agency's first female detective?**Ground Truth:** Kate Warne

**EvolveR:** <think>I need to determine who was the first female detective employed by the Pinkerton Detective Agency. This requires identifying the historical context and any notable female figures associated with this agency. Based on prior reasoning experience, I anticipate some potentially useful principles:- Principle 1: (Historical Agency, First Female Employee, Female Detective) - This principle helps identify the first female to hold a specific role within a historical organization. - Principle 2: (Organization, Historical Role, Notable Individual) - This principle assists in linking an organization to a notable individual who played a significant role in its history.</think> <search\_experience>- Principle 1: (Historical Agency, First Female Employee, Female Detective) - This principle helps identify the first female to hold a specific role within a historical organization. - Principle 2: (Organization, Historical Role, Notable Individual) - This principle assists in linking an organization to a notable individual who played a significant role in its history.</search\_experience>

**Env:** <experience>[Principle 0], type: guiding, metric score: 0.67, description: To identify the first woman to hold a specific significant position within an organization or entity, connect the historical narrative context and mention which event or timeline may signify the position in question...</experience>

**EvolveR:** <think>Based on the experience principles, I plan to apply the principles related to identifying the first female employee in a historical organization and linking this to notable figures. I will also consider using historical documents or specific timelines to confirm the answer.</think> <search\_knowledge>- Query: who was the first female detective employed by the Pinkerton Detective Agency?</search\_knowledge>

**Env:** <information>Doc 1>Title: "Kate Warne")... Doc 2>Title: "Private investigator") ... Doc 3>Title: "Kate Warne")...</information>

**EvolveR:** <think>Based on the retrieved documents, I need to determine who was the first female detective employed by the Pinkerton Detective Agency. I will apply the principles to connect the historical context and notable figures mentioned. - Principle 1: (Historical Agency, First Female Employee, Female Detective) - This principle helps identify the first female detective within the Pinkerton Detective Agency. - Principle 2: (Organization, Historical Role, Notable Individual) - This principle links the Pinkerton Detective Agency to Kate Warne as the first female detective.</think> <search\_knowledge>- Query: who was the first female detective employed by the Pinkerton Detective Agency?</search\_knowledge>

**Env:** <information>Doc 1>Title: "Kate Warne") ... Doc 2>Title: "Private investigator") ... Doc 3>Title: "Kate Warne")...</information>

**EvolveR:** <think>Based on the retrieved documents, Kate Warne is identified as the first female detective employed by the Pinkerton Detective Agency. This aligns with the historical context and the principles applied. - Principle 1: (Historical Agency, First Female Employee, Female Detective) - This principle confirms Kate Warne as the first female detective within the Pinkerton Detective Agency. - Principle 2: (Organization, Historical Role, Notable Individual) - This principle links Kate Warne to her role as the first female detective in the agency's history.</think> <search\_knowledge>- Query: who was the first female detective employed by the Pinkerton Detective Agency?</search\_knowledge>

**Env:** <information>Doc 1>Title: "Kate Warne") ... Doc 2>Title: "Private investigator") ... Doc 3>Title: "Kate Warne")...</information>

**EvolveR:** <think>Based on the retrieved documents, Kate Warne is confirmed as the first female detective employed by the Pinkerton Detective Agency. The information aligns with both historical documentation and the application of principles regarding notable figures in organizational history. - Principle 1: (Historical Agency, First Female Employee, Female Detective) - This principle confirms Kate Warne as the first female detective within the Pinkerton Detective Agency. - Principle 2: (Organization, Historical Role, Notable Individual) - This principle links Kate Warne to her role as the first female detective in the agency's history.</think>

<answer>Kate Warne</answer>