Joint Optimization of Reasoning and Dual-Memory for Self-Learning Diagnostic Agent

Bingxuan Li[1], **Simo Du**[2], **Yue Guo**[1]
University of Illinois Urbana-Champaign[1], Albert Einstein College of Medicine[2]
bl61@illinois.edu, yueg@illinois.edu

Abstract

Clinical expertise improves not only by acquiring medical knowledge, but by accumulating experience that yields reusable diagnostic patterns. Recent LLMs-based diagnostic agents have shown promising progress in clinical reasoning for decision support. However, most approaches treat cases independently, limiting experience reuse and continual adaptation. We propose **SEA**, a self-learning diagnostic agent with cognitively inspired dual-memory module. We design a reinforcement training framework tailored to our designed agent for joint optimization of reasoning and memory management. We evaluate **SEA** in two complementary settings. On standard evaluation with MedCaseReasoning dataset, **SEA** achieves 92.46% accuracy, outperforming the strongest baseline by +19.6%, demonstrating the benefit of jointly optimizing reasoning and memory. On the long-horizon with ER-Reason dataset, **SEA** attains the best final accuracy (0.7214) and the largest improvement (+0.35 ΔAcc@100), while baseline methods show limited or unstable gains. Expert evaluation further indicates that the consolidated rules from **SEA** shows strong clinical correctness, usefulness and trust, suggesting that the induced rules in dual-memory module are reliable and practically meaningful. Overall, **SEA** improves both diagnostic reasoning ability and continual learning by effectively transforming experience into reusable knowledge.

1 Introduction

In real-world clinical practice, expert diagnostic decision-making does not arise from the accumulation of declarative medical knowledge alone. Instead, it develops through experience-shaped cognitive structures: Compact mental representations that organize knowledge into usable patterns (Schmidt et al., 1990; Dornan et al., 2019). During clinical encounters, clinicians rely on two complementary resources: (i) formal professional knowledge (e.g., disease definitions, guidelines, and pathophysiology) and (ii) experiential reasoning grounded in prior cases, where remembered contexts and outcomes guide the interpretation of new evidence. The latter form of learning is particularly critical for rare diseases and atypical presentations, where textbook descriptions are often insufficient and diagnostically relevant signals are subtle.

Large Language Models (LLMs) have recently shown strong potential for automating clinical reasoning and diagnosis. One line of work frames diagnosis as an *agentic* workflow, emphasizing explicit intermediate steps to enable grounded and auditable reasoning (Zhao et al., 2025b; Wang et al., 2025; Zang et al., 2025; Peng et al., 2025). Another line investigates *multi-agent* coordination, such as role specialization, debate, or consensus mechanisms, to improve robustness and efficiency (Chen et al., 2024; Zuo et al., 2025; Zhao et al., 2025a; Li et al., 2025; Han et al., 2025; Zheng et al., 2025). Separately, several studies explore long-term learning for agents (Qiu et al., 2025; Wang et al., 2021b;a; Liu et al., 2026). However, most prior work focuses on teaching agents *what* to diagnose (i.e., acquiring medical knowledge) rather than *how* to improve through accumulated experience, as clinicians do. This gap motivates our central question: *Can agents benefit from experience-driven long-term learning, and how should such experience be stored and reused?*

A key bottleneck is long-term memory management. While general-purpose memory mechanisms have shown promise for agentic AI in web and sandbox environments (Peng et al., 2025; Zang et al., 2025; Chen et al., 2024; Zuo et al., 2025), the medical domain poses distinct challenges: clinical cases are information-dense, evidence is noisy or incomplete, and useful learning often requires abstraction (e.g., reusable heuristics) rather than verbatim storage of trajectories. Naïvely retaining every past case quickly becomes infeasible, whereas overly aggressive compression risks discarding rare but diagnostically decisive signals.

To address this issue, we present **SEA**, a self-learning diagnostic agent for long-horizon improvement from experience, built around two coupled designs: First, we introduce a **dual-memory architecture** inspired by human cognition: a *short-term memory* stores a bounded set of recent, concrete cases, while a *long-term memory* stores abstract diagnostic rules distilled from experience. The agent maintains up to K case memories in an append-only manner; once this capacity exceeded, selected cases are evicted and summarized into reusable reasoning rules, which are consolidated into long-term memory. Second, we propose a **reinforcement learning framework that jointly optimize diagnostic reasoning and memory management.** We design structured rollouts that produce traceable reasoning trajectories leading to final diagnosis while explicitly deciding how memory is updated. We then introduce *adaptive rewards* that jointly optimize (i) diagnostic accuracy and (ii) effective memory-management decisions, encouraging continual improvement from experience rather than passive accumulating knowledge.

We evaluate **SEA** in two complementary settings: (1) In the standard evaluation on MedCaseReasoning dataset (Wu et al., 2025), which measures static diagnostic competence without cross-case adaptation, **SEA** achieves substantial gains over all baselines, reaching up to 92.46% accuracy and surpassing the strongest baseline by +19.6%, demonstrating that jointly optimizing reasoning and memory yields significantly stronger clinical decision-making than outcome-only supervision or naive memory integration. (2) In the long-horizon setting on ER-Reason dataset (Mehandru et al., 2025), which simulates streaming deployment with feedback, **SEA** achieves the best final accuracy of 0.7214 and the largest improvement of +0.35 (ΔAcc@100), while standard and RL-trained models show only marginal gains (e.g., +0.03) or even degradation (e.g., –0.05 at early stages). These results highlight that effective adaptation requires explicit mechanisms for experience accumulation and reuse rather than relying on training alone. Additionally, expert evaluation on the consolidated rules from **SEA** yields strong scores in overall trust (4.216), rule usefulness (3.946), and clinical correctness (3.865)on a 5-point Likert scale (5 denotes the highest rating, with an overall average of 3.655, indicating that the induced rules are reliable and practically meaningful. Overall, **SEA** not only improves diagnostic reasoning capability but also enables sustained learning during deployment by transforming experience into generalizable knowledge.

2 Related Work

LLM Agents for Disease Diagnosis. Medical foundation models achieve strong performance in biomedical QA and clinical reasoning, indicating that generalist LLMs can serve as clinical decision backbones. Tu et al. (2024); Singhal et al. (2025) demonstrate near expert-level QA while highlighting safety gaps. Domain-specialized multimodal models extend this capability to heterogeneous medical data (Sellergren et al., 2025; Lin et al., 2025). Concurrently, diagnosis is increasingly framed as an *agentic* process with structured workflows and verifiable steps, including traceable rare-disease systems (Zhao et al., 2025b), multimodal pipelines (Wang et al., 2025), and hierarchical reasoning frameworks (Zang et al., 2025; Peng et al., 2025). Multi-agent approaches further improve robustness and efficiency via role decomposition and collaboration (Chen et al., 2024; Zuo et al., 2025; Zhao et al., 2025a; Li et al., 2025; Han et al., 2025). Beyond diagnosis, agentic systems also support modular biomedical discovery workflows (Ock et al., 2025).

Memory Mechanisms for Agentic AI. A central challenge in clinical agents is maintaining evolving context across multi-step reasoning. Existing systems often rely on structured intermediates (e.g., evidence or logic trees) to track hypotheses (Peng et al., 2025; Zang et al., 2025; Chen et al., 2024; Zuo et al., 2025). More recent work treats memory as an explicit,

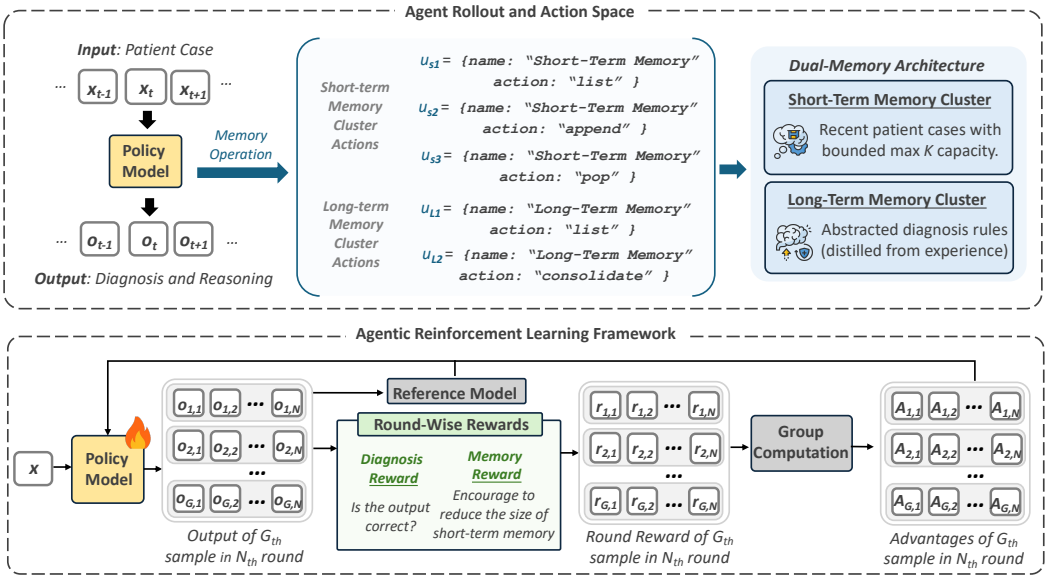

Figure 1: Overview of **SEA**: At each round t, the policy model observes a patient case x_t and may invoke *memory operations* before emitting the final output o_t (diagnosis and reasoning). The agent controls a short-term memory cluster that stores recent patient cases with a bounded capacity K (`list`/`append`/`pop`) and a long-term memory cluster that consolidates experience into abstracted diagnosis rules (`list`/`consolidate`). For each round, we sample a group of candidate outputs, score them with round-wise rewards, compute group-wise advantages, and update the policy.

learnable component that accumulates experience, tracks decisions, and improves future behavior. For example, self-learned knowledge in multi-agent diagnosis acts as persistent memory (Li et al., 2025), while experience-driven agents use memory to store feedback and adapt over time (Yang et al., 2025b). These trends motivate memory designs that record outcomes, retrieve relevant experience, and refine diagnostic strategies beyond static context windows.

Reinforcement Learning for Long Horizon Self-Evolving. Reinforcement learning is increasingly used to enhance long-horizon agent reasoning through iterative feedback. In medicine, RL-based frameworks model diagnosis as sequential hypothesis testing and evidence acquisition (Bani-Harouni et al., 2025), while clinical agents leverage experiential learning to refine decision policies (Lai et al., 2025). More broadly, experience-based RL methods improve both reasoning correctness and process quality over extended trajectories (Zhan et al., 2025; Shao et al., 2025). Together, these works suggest a unified paradigm for self-evolving agents: combine structured reasoning, explicit memory, and RL signals to enable continual adaptation from experience (Yang et al., 2025b).

3 Method

We propose a reinforcement learning framework that jointly optimizes (i) diagnostic reasoning and (ii) memory management for continual on-the-job improvement.

3.1 Agent Design

Dual-memory architecture. A clinician improves by remembering concrete failures/successes and by distilling reusable rules from them. We want the same behavior from the agent: keep a small set of recent outcome-annotated cases for grounding, while gradually building a compact rule set that transfers across future patients. This design also makes

the bottleneck explicit: the agent must decide *what to keep* and *what to abstract*, instead of relying on an external heuristic that may discard the wrong evidence. Motivated by this, we introduce a dual-memory design inspired by human cognition. The agent maintains: (i) a *short-term case memory cluster* $\mathcal{M}_t^{\mathrm{S}}$ that stores a bounded set of recent, concrete cases with outcomes, and (ii) a *long-term rule memory cluster* $\mathcal{M}_t^{\mathrm{L}}$ that stores abstract diagnostic rules distilled from experience. The agent state is $s_t = (\mathcal{M}_t^{\mathrm{S}}, \mathcal{M}_t^{\mathrm{L}})$.

Short-term case memory cluster. Recent cases provide high-fidelity evidence (what was observed, what was predicted, and what feedback was returned), which is crucial for correcting near-miss mistakes and adapting to distribution shifts. But this evidence is expensive to store in-context, so we enforce a hard capacity K to reflect realistic deployment constraints:

$$\mathcal{M}_t^{\mathrm{S}} = \{c_1, \ldots, c_{|\mathcal{M}_t^{\mathrm{S}}|}\}, \qquad |\mathcal{M}_t^{\mathrm{S}}| \leq K, \tag{1}$$

where each record c contains compact information such as $(x, \mathcal{Y}, \hat{y}, f)$ and optionally a brief rationale snippet. We use an append-only interface: after finishing round t, the new case record is appended, and the agent may trigger a *pop-and-summarize* operation when the cluster approaches the capacity limit.

Long-term rule memory cluster. The long-term memory stores a set of reusable diagnostic rules (e.g., symptom–disease associations, discriminative cues, and failure patterns) that are distilled from evicted short-term cases:

$$\mathcal{M}_t^{\mathrm{L}} = \{r_1, \ldots, r_{|\mathcal{M}_t^{\mathrm{L}}|}\}, \tag{2}$$

where each rule r is a concise, natural-language (or structured) statement that can be retrieved and applied in later rounds.

Memory operations as agent actions. Beyond emitting the diagnostic output, the agent also chooses a memory operation u_t from a small action space:

$$u_t \in \{\texttt{list}, \texttt{append}, \texttt{pop}(p), \texttt{consolidate}\}. \tag{3}$$

Intuitively, `list` retrieves the current contents of the episodic case cluster (and the rules in long-term memory) to support case-based reasoning; `append` inserts the newly observed case–outcome pair into the case cluster; $\texttt{pop}(p)$ selects one or more cases to evict when the cluster approaches its capacity; and `consolidate` summarizes the evicted cases into compact diagnostic rules and inserts them into $\mathcal{M}^{\mathrm{L}}$. Importantly, while the case cluster has a hard capacity K, we let the agent *decide when* to pop cases and *how* to distill them into reusable rules.

Structured rollout with Dual-Memory. Separating diagnose and memory management makes the agent's behavior auditable. We can inspect not only what it predicted, but also what it chose to store as experience and what rules it distilled. A round-t rollout therefore consists of a diagnostic output and an explicit memory-update decision:

$$(o_t, u_t) \sim \pi_\theta(\cdot \mid x_t, \mathcal{Y}_t, \mathcal{M}_t^{\mathrm{S}}, \mathcal{M}_t^{\mathrm{L}}),$$

followed by deterministic state transitions that update $\mathcal{M}_{t+1}^{\mathrm{S}}, \mathcal{M}_{t+1}^{\mathrm{L}}$ according to u_t.

3.2 Reward Modeling

We design a reward function that encourages the agent to (i) predict correctly and (ii) manage memory efficiently so that it improves over time. Concretely, we decompose the per-round reward into a *diagnostic reward* and a *memory reward*:

$$r_t \;=\; \lambda_{\text{diag}}\, r_t^{\text{diag}} \;+\; \lambda_{\text{mem}}\, r_t^{\text{mem}},$$

where $\lambda_{\text{diag}}, \lambda_{\text{mem}}$ balance diagnostic performance and memory behavior.

Diagnostic reward. We use feedback-conditioned rewards that provide strong supervision for correct predictions:

$$r_t^{\text{diag}} = \begin{cases} +5, & \text{if } \hat{y}_t \text{ is correct under } f_t, \\ -5, & \text{otherwise.} \end{cases}$$

Here, we set the reward magnitude to ± 5 to balance signal strength and training stability in practice.

Memory-management reward. To prevent uncontrolled growth of short-term memory and incentivize timely consolidation, we penalize large cluster occupancy:

$$r_t^{\text{mem}} \;=\; -\alpha \cdot \frac{|\mathcal{M}_t^{\text{S}}|}{K},$$

with α controlling the strength ($\alpha = 3$ in our implementation). This reward encourages the agent to pop and summarize when appropriate, rather than passively accumulating cases until truncation.

Final per-round reward. Putting everything together, the final reward is designed as

$$r_t \;=\; \lambda_{\text{diag}}\, r_t^{\text{diag}} \;+\; \lambda_{\text{mem}}\, r_t^{\text{mem}}$$

3.3 Round-Wise Rewarding

Long-horizon diagnosis exhibits a cold-start challenge: early in the stream, the agent has limited experience for the target population and its memory is poorly calibrated. In this phase, directly optimizing only correctness can be high-variance; instead, the most useful behavior is to *acquire and consolidate* informative case evidence into persistent memory. As interaction progresses, the agent's memory becomes richer and more stable, and the learning signal should shift toward diagnostic accuracy.

Round-dependent reward shaping. To encourage this staged learning, we schedule the trade-off between diagnostic reward and memory reward across rounds. Let the normalized round index be $i_t = \frac{t}{T} \in [0, 1]$. We linearly interpolate the weights:

$$\lambda_{\text{diag}}(t) = \lambda_{\text{diag}}^{\max} \cdot i_t, \qquad \lambda_{\text{mem}}(t) = \lambda_{\text{mem}}^{\max} \cdot (1 - i_t),$$

so that early rounds emphasize memory formation while later rounds emphasize diagnostic accuracy. The shaped per-round reward is:

$$\tilde{r}_t \;=\; \lambda_{\text{diag}}(t)\, r_t^{\text{diag}} + \lambda_{\text{mem}}(t)\, r_t^{\text{mem}}.$$

Round-wise advantage estimation. Because $\tilde{r}_t$ is non-stationary across t under this schedule, trajectory-level normalization can cause later rounds to dominate updates and make early-round gradients noisy. We therefore compute advantages *separately for each round position*. For each round t, we sample a group of K rollouts and obtain $\{\tilde{r}_t^{(k)}\}_{k=1}^{K}$. We compute the group-relative advantage at round t:

$$\hat{A}_t^{(k)} \;=\; \tilde{r}_t^{(k)} \;-\; \frac{1}{K} \sum_{j=1}^{K} \tilde{r}_t^{(j)}.$$

We then optimize the policy with the standard clipped GRPO objective Shao et al. (2024), using $\hat{A}_t^{(k)}$ in place of trajectory-level advantages.

4 Experiments

We evaluate **SEA** in two complementary settings that isolate (i) *static diagnostic competence*: Standard Evaluation measures accuracy when each case is solved independently, without carrying experience across cases; (ii) *continual on-the-job improvement*: Long-Horizon Evaluation simulates deployment in a streaming environment, where the agent receives feedback after each case and must adapt over time via the memory mechanisms. More details on **SEA** implementation for both experiment settings is introduced in Appendix E.

4.1 Standard Evaluation Setting

This setting assesses *static clinical reasoning ability*: Given a single case and a clinically plausible candidate set, can the model select the correct diagnosis?

4.1.1 Setup

Dataset. We use **MedCaseReasoning** (Wu et al., 2025), which contains challenging clinical cases paired with diagnosis labels. We train on the official training split and report results on the test split, which emphasizes rare and complex diseases. We adopt a closed-set formulation to enable controlled and comparable evaluation across methods. By constraining the output space to a shared candidate set, we eliminate variability introduced by free-form generation and ensure that performance differences reflect reasoning quality rather than differences in answer expression. We fix the entire candidate-generation pipeline across methods: the distractor count N, sampling strategy, and random seeds are held constant. Importantly, we carefully design the distractor generation process to ensure high-quality and challenging candidate sets. Rather than sampling negatives uniformly, we construct clinically plausible and semantically similar distractors to the ground-truth diagnosis, requiring models to perform fine-grained reasoning rather than relying on superficial cues. Thus, for each case, every method sees the same $\mathcal{Y}$ and is evaluated on the same closed-set decision. More details on the candidate-generation is described in Appendix D.

Evaluation Protocol. Each instance provides a patient case x (demographics, symptoms) and a candidate set $\mathcal{Y} = \{y^{(1)}, \ldots, y^{(m)}\}$ containing the ground-truth diagnosis and $N = 199$ clinically plausible distractors. All methods must produce (i) a structured reasoning trace and (ii) a final prediction $\hat{y} \in \mathcal{Y}$, from which accuracy is computed. More details on evaluation setting is attached to Appendix F.

Implementation. We use Qwen3-4b and Qwen3-8B (Yang et al., 2025a) as the base policy model for **SEA** and train with the paradigm introduced in Section 3.2. We additionally choose OLMo3-7B(Olmo et al., 2025), LLaMa-3.1-8B(Grattafiori et al., 2024), and MedGemma-27B(Sellergren et al., 2025) as baseline models, spanning both general-domain and medical-domain settings as well as a range of model sizes. We compare against: **(i)** Zero-shot, **(ii)** Surpervised Finetuning Baselines, **(iii)** RL baselines, and **(iv)** Memory augmented baselines. We fix the memory tool schema and maximum call limits to match **SEA** for fair comparison. More details on implementation is attached in Appendix F.2.

Model	Accuracy (%)	Macro-F1 (%)	Std.
Zeroshot			
Qwen3-4b	43.2	40.0	0.025
Qwen3-8b	45.1	41.9	0.025
Qwen3-14B	50.4	42.9	0.024
OLMo3-7B	47.1	43.9	0.025
LLaMa-3.1-8B	42.1	39.8	0.025
MedGemma-27B	54.7	51.1	0.023
Supervised Fine-Tuning (SFT)			
Qwen3-4b (SFT)	72.3	70.1	0.020
Qwen3-8b (SFT)	72.1	69.0	0.020
Reinforcement Learning (RL)			
Qwen3-4b (RL-DiagnosticRewardOnly)	74.1	69.5	0.025
Qwen3-8b (RL-DiagnosticRewardOnly)	77.3	72.3	0.023
Memory-Augmented			
Qwen3-8b (ReAct+ShortTerm-Memory)	42.5	38.3	0.025
Qwen3-8b (SFT+ShortTerm-Memory)	56.3	50.9	0.023
SEA (Qwen-4b)	84.5 (+9.3%)	81.8 (+13.1%)	0.015
SEA (Qwen-8b)	**92.5 (+19.6%)**	**86.8 (+20.1%)**	0.012

Table 1: Performance comparison across methods under the standard evaluation setting.

4.1.2 Results

Table 1 reports experiment results. We conclude three insights as following.

Insight 1: Joint Reasoning–Memory Optimization Yields Large Gains. **SEA** consistently outperforms all baselines by a large margin. The Qwen-8B variant achieves 92.5% accuracy

and 86.81 macro-F1, surpassing the strongest baseline by +19.6% and +20.1%, respectively. Notably, similar improvements hold for the 4B model, suggesting that the gains are not solely due to scale but arise from the proposed joint optimization of reasoning and memory.

Insight 2: Outcome-Only Reward Is Insufficient for Complex Diagnosis. While reinforcement learning improves over supervised fine-tuning, the gains remain modest. This indicates that optimizing only for final diagnostic correctness provides sparse and underspecified supervision. In contrast, **SEA** introduces structured rewards that guide both intermediate reasoning and memory usage, leading to substantially better performance on challenging clinical cases.

Insight 3: Naive Memory Integration Can Hurt Performance. Memory-augmented baselines show that simply adding memory is not effective. ReAct with short-term memory drops to **42.5%**, even below zeroshot performance, suggesting that unstructured retrieval introduces noise and disrupts reasoning. Even when combined with SFT, memory yields limited improvement, far below RL-based methods.

Overall, **SEA** substantially outperforms all baselines with results suggesting that jointly optimizing reasoning and memory is effective.

4.2 Long-Horizon Evaluation Setting

This setting assesses *on-the-job improvement*: Can the agent convert per-case feedback into reusable experience that improves future diagnoses over a stream?

4.2.1 Setup

Dataset. We use **ER-Reason** (Mehandru et al., 2025), which contains heterogeneous emergency-room cases. We construct $\mathcal{Y}_t$ using the same distractor procedure as in Section 4.1, ensuring comparable decision difficulty across methods.

Evaluation Protocol. We do *not* train on ER-Reason. Instead, we create a single-pass stream $\{x_t\}_{t=1}^{T}$. At each round t, the agent receives a case x_t and a candidate set $\mathcal{Y}_t$, predicts $\hat{y}_t$, then receives feedback f_t (e.g., the ground-truth label) before proceeding to $t+1$. This protocol explicitly evaluates whether the agent improves as more feedback is observed. We design two evaluation metrics: (1) **Final Accuracy**: the average top-1 accuracy computed over the entire interaction stream, reflecting the overall diagnostic performance after all learning and adaptation has taken place. (2) **Improvement over n turns**: $\Delta\text{Acc@}n = \text{Acc}(1{:}n) - \text{Acc}(1{:}n_0)$ with warm-up $n_0 = 10$, which measures the relative performance gain compared to an early-stage performance. More details on evaluation setting is attached to Appendix F.

Implementation We select representative trained models from previous experiments. Model weights are frozen for all methods. We additionally compare against a strong general reasoning model GPT-5.2 (OpenAI, 2025) in the zero-shot and memory-augmented setting. More details on implementation is attached in Appendix F.2.

4.2.2 Results

Table 2 summarizes performance under the long-horizon streaming setting. We highlight three key findings from experiment results.

Insight 1: Memory alone enables adaptation but introduces a trade-off. Augmenting models with dual-memory significantly improves long-horizon adaptation. Both Qwen-8b (SFT + Dual Memory) and GPT-5.2 (Zeroshot + Dual Memory) show substantial gains (+0.21 and +0.17 at 100 rounds, respectively), confirming that external memory provides a mechanism for leveraging past feedback. However, this comes at a cost: Qwen-8b (SFT + Dual Memory) suffers a drop in final accuracy compared to its SFT counterpart (54% vs. 56%), suggesting that naive memory usage can introduce noise or retrieval errors that hurt overall decision quality.

Insight 2: SEA achieves both strong final performance and sustained improvement. SEA outperforms baselines across both final accuracy and improvement metrics. In particular,

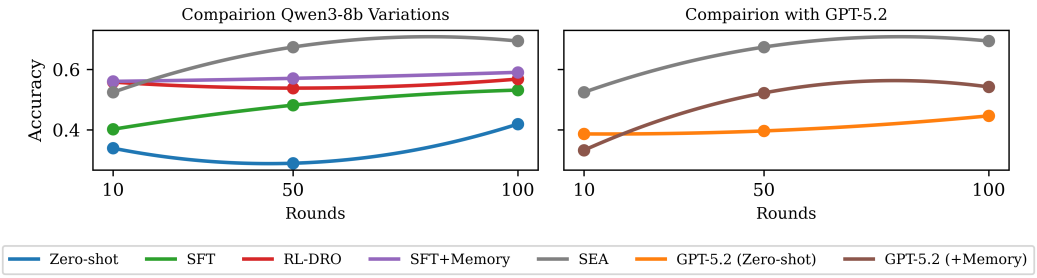

Figure 2: Accuracy trajectories from 10 to 100 rounds for representative methods.

Method	Final Accuracy (%)	ΔAcc@50 (%)	ΔAcc@100 (%)
Zeroshot			
Qwen-4b	41.9	-5.0	+8.0
Qwen-8b	44.6	+1.0	+6.0
GPT-5.2	53.2	+8.0	+13.0
Trained Baselines			
Qwen-8b (SFT)	56.8	-2.0	+1.0
Qwen-8b (RL-DiagnosticRewardOnly)	59.1	+1.0	+3.0
Memory-Augmented			
Qwen-8b (SFT + Dual Memory)	54.2	+19.0	+21.0
GPT-5.2 (Zeroshot + Dual Memory)	69.4	+15.0	+17.0
SEA (Qwen-4b)	68.5	**+23.0**	+32.0
SEA (Qwen-8b)	**72.1**	+20.0	**+35.0**

Table 2: Final accuracy and improvement across different methods.

SEA (Qwen-8b) achieves the best final accuracy (0.72) and the largest long-horizon gain (ΔAcc@100 = +0.35), substantially exceeding both memory-augmented and trained baselines. Notably, SEA avoids the trade-off observed in naive memory systems: it improves continuously over time while also maintaining superior overall accuracy.

Insight 3: Dual-memory is a plug-and-play module for effective memory management. Our dual-memory design serves as an external, plug-and-play module that can be seamlessly integrated with different backbone models to enable structured memory management. Without any additional training, it equips models with the ability to store, retrieve, and consolidate experience over time. Augmenting GPT-5.2 with dual-memory improves final accuracy from 0.53 to 0.69 and increases long-horizon gain from +0.13 to +0.17 at 100 rounds. These results demonstrate that the benefits of our approach generalize across models, highlighting that effective adaptation can be achieved through a modular and reusable memory mechanism without modifying model parameters.

These results demonstrate that the ability to *learn during deployment* is not an emergent property of standard LLMs or conventional training pipelines.

5 Discussion

5.1 Expert Evaluation

The central design of **SEA** is the dual-memory module, which enables the agent to induce reusable diagnostic rules from historical cases. While quantitative results suggest improved reasoning performance, it remains essential to assess whether the induced rules reflect clinically meaningful knowledge rather than superficial heuristics that improve task metrics.

To gain further insight into the nature of these rules, we conduct a human evaluation with a professional physician. For each case, the evaluator is presented with the patient context, the model's reasoning trace, and the induced rule, and is asked to assess them based on clinical validity and practical usefulness. Note that **this evaluation is intended to provide a qualitative perspective on the generated rules and to better understand how SEA leverages experience, rather than to serve as a definitive assessment of clinical quality**. We additionally provided the real trajectory from SEA in Appendix **??**.

Evaluation Dimensions. We evaluate the updated reasoning and induced rules along four dimensions: (1) *Clinical Correctness*, which measures factual accuracy and medical soundness; (2) *Case Relevance*, which assesses whether the reasoning and rules are properly grounded in patient-specific details; (3) *Rule Usefulness*, which captures the extent to which the induced rules are actionable and applicable in real diagnostic scenarios; and (4) *Overall Trust*, which reflects the physician's holistic confidence in the reasoning and rule quality. Each dimension is rated on a 5-point Likert scale, where higher scores indicate better performance. More details are provided in Appendix G.

Results. In the expert evaluation, SEA achieves an overall average score of 3.655 out of 5. The model demonstrates high *overall trust* (4.2) and *rule usefulness* (3.9), indicating that the induced rules are perceived as reliable and practically applicable in clinical settings. *Clinical correctness* also remains high (3.8), suggesting that both the reasoning and induced rules are largely medically sound. Interestingly, *case relevance* receives a relatively lower score (2.6). We emphasize that this behavior is expected and aligns with the design objective of SEA. The goal of rule induction is to abstract generalizable diagnostic principles from individual cases, rather than overfit to case-specific details. As a result, the induced rules intentionally prioritize cross-case applicability over tight coupling to a single instance.

5.2 Ablation Study

Method	Ablated Module	Acc.	Δ
Zero-shot	Training & Memory Modules	0.45	-0.47
DiagnosticReward (w/o memory)	All Memory Modules	0.77	-0.15
DiagnosticReward+ShortTerm-memory	LongTerm Memory&MemoryReward	0.82	-0.10
DiagnosticReward+LongTerm-Memory	ShortTerm Memory&MemoryReward	0.72	-0.20
DiagnosticReward+Dual-Memory	Memory Reward	0.47	-0.45
SEA	None (full model)	**0.92**	—

Table 3: Ablation study on MedCaseReasoning. Δ denotes performance change relative to the full model. Darker color indicates larger performance drop.

To identify which components contribute most to SEA's gains, we perform controlled ablations on MedCaseReasoning. Table 3 reports the results. This result demonstrates that retaining recent case-specific information helps the model better align its reasoning with the current diagnostic context, leading to more accurate predictions. While long-term memory captures generalizable patterns across cases, it is less effective without mechanisms that ensure its relevance to the current instance. Combining short-term and long-term memory without the memory-management reward leads to a sharp performance drop to 0.4741, even below the zero-shot baseline. This indicates that unregulated memory interaction can introduce noise and conflicting signals, ultimately harming the reasoning process. The full model achieves the best performance.

6 Conclusion

We propose SEA, a self-learning diagnostic agent with a cognitively inspired dual-memory module, trained via a reinforcement framework that jointly optimizes reasoning and memory management. Across two settings, SEA consistently outperforms strong baselines: It achieves 92.46% accuracy on MedCaseReasoning with standard evaluation setting (+19.6%)

and attains the best final accuracy (0.7214) with the largest improvement (+0.35 ΔAcc@100) on the ER-Reason dataset with long-horizon setting.

Acknowledgment

This work used Delta GPU at NCSA through allocation [CIS240504] from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296.

References

David Bani-Harouni, Chantal Pellegrini, Ege Özsoy, Matthias Keicher, and Nassir Navab. Language agents for hypothesis-driven clinical decision making with reinforcement learning. *arXiv preprint arXiv:2506.13474*, 2025.

Xuanzhong Chen, Ye Jin, Xiaohao Mao, Lun Wang, Shuyang Zhang, and Ting Chen. Rareagents: Advancing rare disease care through llm-empowered multi-disciplinary team. *arXiv preprint arXiv:2412.12475*, 2024.

Tim Dornan, Richard Conn, Helen Monaghan, Grainne Kearney, Hannah Gillespie, and Deirdre Bennett. Experience based learning (exbl): clinical teaching for the twenty-first century. *Medical teacher*, 41(10):1098–1105, 2019.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Xudong Han, Xianglun Gao, Xiaoyi Qu, and Zhenyu Yu. Multi-agent medical decision consensus matrix system: An intelligent collaborative framework for oncology mdt consultations. *arXiv preprint arXiv:2512.14321*, 2025.

Yunghwei Lai, Kaiming Liu, Ziyue Wang, Weizhi Ma, and Yang Liu. Doctor-r1: Mastering clinical inquiry with experiential agentic reinforcement learning. *arXiv preprint arXiv:2510.04284*, 2025.

Wenliang Li, Rui Yan, Xu Zhang, Li Chen, Hongji Zhu, Jing Zhao, Junjun Li, Mengru Li, Wei Cao, Zihang Jiang, et al. Macd: Multi-agent clinical diagnosis with self-learned knowledge for llm. *arXiv preprint arXiv:2509.20067*, 2025.

Tianwei Lin, Wenqiao Zhang, Sijing Li, Yuqian Yuan, Binhe Yu, Haoyuan Li, Wanggui He, Hao Jiang, Mengze Li, Xiaohui Song, et al. Healthgpt: A medical large vision-language model for unifying comprehension and generation via heterogeneous knowledge adaptation. *arXiv preprint arXiv:2502.09838*, 2025.

Qicai Liu, Zhichao Hu, Tao Huang, Yupeng Niu, Xinche Zhang, Shanwu Ma, Chutong Lin, Goh Kim Huat, Hyeokkoo Eric Kwon, Feng Gao, et al. Evomdt: a self-evolving multi-agent system for structured clinical decision-making in multi-cancer. *npj Digital Medicine*, 2026.

Nikita Mehandru, Niloufar Golchini, David Bamman, Travis Zack, Melanie F Molina, and Ahmed Alaa. Er-reason: A benchmark dataset for llm-based clinical reasoning in the emergency room. *arXiv preprint arXiv:2505.22919*, 2025.

Janghoon Ock, Radheesh Sharma Meda, Srivathsan Badrinarayanan, Neha S Aluru, Achuth Chandrasekhar, and Amir Barati Farimani. Large language model agent for modular task execution in drug discovery. *arXiv preprint arXiv:2507.02925*, 2025.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch,

Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2025. URL https://arxiv.org/abs/2512.13961.

OpenAI. Gpt-5 system card. https://cdn.openai.com/gpt-5-system-card.pdf, August 2025. Accessed: 2026-01-03.

Qi Peng, Jialin Cui, Jiayuan Xie, Yi Cai, and Qing Li. Tree-of-reasoning: Towards complex medical diagnosis via multi-agent reasoning with evidence tree. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 1744–1753, 2025.

Pengcheng Qiu, Chaoyi Wu, Junwei Liu, Qiaoyu Zheng, Yusheng Liao, Haowen Wang, Yun Yue, Qianrui Fan, Shuai Zhen, Jian Wang, et al. Evolving diagnostic agents in a virtual clinical environment. *arXiv preprint arXiv:2510.24654*, 2025.

Henk G Schmidt, Geoffrey R Norman, and Henny P Boshuizen. A cognitive perspective on medical expertise: theory and implication [published erratum appears in acad med 1992 apr; 67 (4): 287]. *Academic medicine*, 65(10):611–21, 1990.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025.

Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G Finlayson, David Sontag, et al. Dr tulu: Reinforcement learning with evolving rubrics for deep research. *arXiv preprint arXiv:2511.19399*, 2025.

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

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Mohamed Amin, Le Hou, Kevin Clark, Stephen R Pfohl, Heather Cole-Lewis, et al. Toward expert-level medical question answering with large language models. *Nature Medicine*, 31(3):943–950, 2025.

Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Charles Lau, Ryutaro Tanno, Ira Ktena, et al. Towards generalist biomedical ai. *Nejm Ai*, 1(3):AIoa2300138, 2024.

Zifeng Wang, Rui Wen, Xi Chen, Shilei Cao, Shao-Lun Huang, Buyue Qian, and Yefeng Zheng. Online disease diagnosis with inductive heterogeneous graph convolutional networks. In *Proceedings of the Web Conference 2021*, WWW '21, pp. 3349–3358, New York, NY, USA, 2021a. Association for Computing Machinery. ISBN 9781450383127. doi: 10.1145/3442381.3449795. URL https://doi.org/10.1145/3442381.3449795.

Zifeng Wang, Yifan Yang, Rui Wen, Xi Chen, Shao-Lun Huang, and Yefeng Zheng. Lifelong learning based disease diagnosis on clinical notes. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 213–224. Springer, 2021b.

Ziyue Wang, Junde Wu, Linghan Cai, Chang Han Low, Xihong Yang, Qiaxuan Li, and Yueming Jin. Medagent-pro: Towards evidence-based multi-modal medical diagnosis via reasoning agentic workflow. *arXiv preprint arXiv:2503.18968*, 2025.

Kevin Wu, Eric Wu, Rahul Thapa, Kevin Wei, Angela Zhang, Arvind Suresh, Jacqueline J Tao, Min Woo Sun, Alejandro Lozano, and James Zou. Medcasereasoning: Evaluating and learning diagnostic reasoning from clinical case reports. *arXiv preprint arXiv:2505.11733*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Cheng Yang, Xuemeng Yang, Licheng Wen, Daocheng Fu, Jianbiao Mei, Rong Wu, Pinlong Cai, Yufan Shen, Nianchen Deng, Botian Shi, et al. Learning on the job: An experience-driven self-evolving agent for long-horizon tasks. *arXiv preprint arXiv:2510.08002*, 2025b.

Zelin Zang, Wenyi Gu, Siqi Ma, Dan Yang, Yue Shen, Zhu Zhang, Guohui Fan, Wing-Kuen Ling, and Fuji Yang. A medical multimodal diagnostic framework integrating vision-language models and logic tree reasoning. *arXiv preprint arXiv:2512.21583*, 2025.

Runzhe Zhan, Yafu Li, Zhi Wang, Xiaoye Qu, Dongrui Liu, Jing Shao, Derek F Wong, and Yu Cheng. Exgrpo: Learning to reason from experience. *arXiv preprint arXiv:2510.02245*, 2025.

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

Huiya Zhao, Yinghao Zhu, Zixiang Wang, Yasha Wang, Junyi Gao, and Liantao Ma. Confagents: A conformal-guided multi-agent framework for cost-efficient medical diagnosis. *arXiv preprint arXiv:2508.04915*, 2025a.

Weike Zhao, Chaoyi Wu, Yanjie Fan, Xiaoman Zhang, Pengcheng Qiu, Yuze Sun, Xiao Zhou, Yanfeng Wang, Xin Sun, Ya Zhang, et al. An agentic system for rare disease diagnosis with traceable reasoning. *arXiv preprint arXiv:2506.20430*, 2025b.

Qiaoyu Zheng, Yuze Sun, Chaoyi Wu, Weike Zhao, Pengcheng Qiu, Yongguo Yu, Kun Sun, Yanfeng Wang, Ya Zhang, and Weidi Xie. End-to-end agentic rag system training for traceable diagnostic reasoning. *arXiv preprint arXiv:2508.15746*, 2025.

Kaiwen Zuo, Yirui Jiang, Fan Mo, and Pietro Lio. Kg4diagnosis: A hierarchical multi-agent llm framework with knowledge graph enhancement for medical diagnosis. In *AAAI Bridge Program on AI for Medicine and Healthcare*, pp. 195–204. PMLR, 2025.

Appendix

A Use of LLMs

In this work, LLMs are used strictly for research support rather than as sources of substantive content. Their use falls into: (i) serving as the tested and trained model, and (ii) assisting with language refinement during paper writing. For writing support, we used GPT-5 solely to polish text (improving coherence and grammar) while all ideas, logic, results, and technical contributions originate from the authors. We note that all GPT-5 calls involving the data are made through an Azure-hosted OpenAI deployment, which provides data-handling safeguards for protected and access-controlled information.

B Limitations and Future Works

Our study intentionally adopts a controlled closed-set diagnosis setting with immediate feedback, allowing us to isolate and rigorously evaluate the effect of memory-based adaptation independent of parameter updates. This design provides a clean testbed to demonstrate that substantial performance gains can be achieved through experience accumulation and memory updates alone, without modifying model weights. We emphasize that these choices are intentional design decisions to isolate the core research question: *can experience alone, when properly structured and stored, drive meaningful improvement?* By controlling the output space and supervision signal, we create a clean and reproducible environment to rigorously study memory-based adaptation without confounding factors. While real-world clinical environments may involve broader action spaces (e.g., test selection and evidence gathering), and strict safety requirements, we view our current setting as a necessary first step toward such complexity. It establishes a reliable foundation upon which these additional challenges can be systematically incorporated.

C Preliminaries

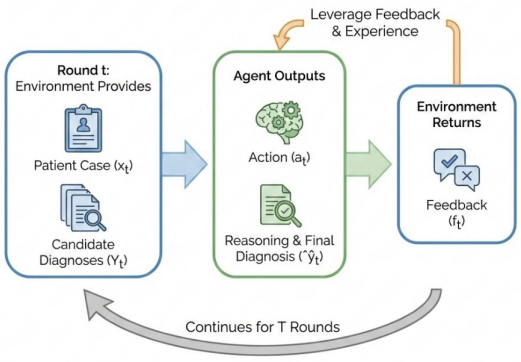

Figure 3: **Task setup:** At each round t, the environment provides a patient case x_t together with a candidate diagnosis set $\mathcal{Y}_t$. The agent outputs an action a_t consisting of a structured reasoning trace and a final prediction $\hat{y}_t \in \mathcal{Y}_t$. The environment then returns feedback f_t (e.g., correct/incorrect or graded), which the agent can leverage as experience for subsequent rounds. This interaction repeats for T rounds, modeling continual on-the-job improvement in a streaming setting.

Long-Horizon Setting Setup. We study long-horizon disease diagnosis as a sequential decision process motivated by *on-the-job improvement* in deployment. In real clinical settings, a diagnostic agent faces a stream of cases from a specific hospital and population, where prevalence, documentation style, and clinical practice can drift over time. A fixed test-time reasoner may be competent on average yet brittle, repeatedly failing on recurring subtypes

or missing subtle local cues. In practice, however, continuously retraining the base model during deployment is often infeasible due to privacy, safety, and operational constraints. This motivates an agent that can *learn from feedback and experience during use*, improving over the stream via a compact internal state (e.g., memory) rather than frequent model retraining.

At round $t \in \{1, \ldots, T\}$, the environment provides a patient case x_t (e.g., demographics, symptoms, and clinical notes) together with a candidate diagnosis set $\mathcal{Y}_t = \{y_t^{(1)}, \ldots, y_t^{(m)}\}$. The agent outputs an action a_t that contains (i) a structured reasoning trace and (ii) a final predicted diagnosis $\hat{y}_t \in \mathcal{Y}_t$ (or $\hat{y}_t \in \mathcal{Y}$ in the open-set setting). After acting, the environment returns feedback f_t, which may include correctness, partial credit, or other evaluative signals. The interaction then continues to the next round.

On-the-job improvement objective. Unlike static test-time inference, our objective is continual improvement during deployment: the agent should become more accurate over the stream by incorporating feedback and experience without retraining on the target test stream. Formally, let π_θ denote the policy parameterized by θ. We aim to learn a policy that maximizes the expected cumulative utility over the stream while enabling performance gains over time. Crucially, the agent maintains an internal state s_t across rounds, which summarizes what it has learned so far (e.g., via memory).

Agentic Reinforcement Learning. Following Zhang et al. (2025), we adopt agentic reinforcement learning to solve the task. We optimize the diagnosis agent as a policy model π_θ that generates a structured output o_t (reasoning trace + final prediction) conditioned on the current case and the agent state: $o_t \sim \pi_\theta(\cdot \mid x_t, \mathcal{Y}_t, s_t)$, where the predicted diagnosis $\hat{y}_t$ is parsed from o_t. Each output is scored by a task-specific reward function $r_t = R(x_t, \mathcal{Y}_t, o_t, f_t)$, which may incorporate decision correctness, partial credit, format constraints, and other criteria.

Group Relative Policy Optimization (GRPO). We train the policy with Group Relative Policy Optimization (GRPO) Shao et al. (2024), a PPO-style method that avoids explicit value-function learning by using *within-group* comparisons among multiple sampled outputs for the same input. At each training step for round t, we draw a group of K candidate structured outputs $\{o_t^{(k)}\}_{k=1}^{K}$ from the current policy $\pi_\theta(\cdot \mid x_t, \mathcal{Y}_t, s_t)$ and score each candidate with our round-wise reward to obtain $\{r_t^{(k)}\}_{k=1}^{K}$. Instead of learning a critic to predict returns, GRPO estimates an advantage signal by *centering* each candidate reward by the group mean, so candidates that outperform their peers receive positive advantages and those that underperform receive negative advantages. We then update π_θ by increasing the likelihood of higher-advantage outputs while applying (i) a clipped policy-ratio constraint to prevent overly large updates and (ii) a KL regularizer that keeps the updated policy close to a reference policy π_{ref} for stability.

D Details on Candidate Label Selection

A core challenge in constructing candidate diagnosis sets is avoiding *trivial* distractors. If negative options are easily distinguishable, models may succeed by exploiting superficial cues (e.g., matching the affected organ system) rather than performing clinically meaningful discrimination. For example, for a lung-related presentation, the gold diagnosis might involve pulmonary inflammation, while distractors may be unrelated conditions of the leg or heart; such mismatches make the task gameable and overestimate diagnostic ability.

To mitigate this issue, we design distractors to be *clinically plausible* and *semantically close* to the ground-truth label. Rather than sampling negatives uniformly at random from a global label pool (e.g., $\sim$ 800 disease names), we adopt a **cluster-then-sample** strategy: given the gold disease name, we retrieve a set of the most relevant diseases from the label list and sample distractors from this neighborhood. Concretely, we use GPT-5.2 OpenAI (2025) to score candidate disease names by clinical relatedness to the gold label, optionally grouping the label space into coarse clusters (e.g., by organ system, etiology, or phenotype similarity). We then draw N distractors from the top-ranked cluster(s), ensuring that negatives share overlapping symptoms, risk factors, or differential-diagnosis structure with the gold

label. This produces candidate sets where success requires fine-grained reasoning (e.g., distinguishing etiologies, time courses, or key discriminative findings) instead of shallow lexical or organ-level matching. Finally, to ensure fair comparison across methods, we fix the entire candidate-generation pipeline: the distractor pool, retrieval/scoring procedure, distractor count $N = 199$, and random seeds are held constant, so every model is evaluated on identical candidate sets.

E SEA Details

E.1 Dual-Memory Module

We implement the dual-memory module as an external tool, enabling plug-and-play integration with different backbone models without modifying their internal parameters. Specifically, the agent interacts with the memory module through structured tool calls, which support memory writing, retrieval, and consolidation during sequential decision-making.

The dual-memory structure consists of two complementary components: a *short-term memory* that stores recent case-specific experiences, and a *long-term memory* that maintains distilled, reusable knowledge (e.g., diagnostic rules). During inference, the agent retrieves relevant context, appends new experiences after feedback, manages memory, and consolidates abstract rules. We incorporate a system prompt explicitly instructs the agent on when and how to invoke memory operations.

```
{
    "type": "function",
    "function": {
        "name": self.name,
        "description": self.description,
        "parameters": {
            "type": "object",
            "properties": {
                "action": {
                    "type": "string",
                    "enum": ["list", "append", "pop", "consolidate"],
                    "description": (
                        "Operation to run. "
                        "`list` returns current short-term cases and long
    -term rules. "
                        "`append` adds a case record to short-term memory
    . "
                        "`pop` evicts cases at given indices from short-
    term memory. "
                        "`consolidate` adds distilled diagnostic rules to
     long-term memory."
                    ),
                },
                "case_record": {
                    "type": "object",
                    "description": (
                        "Case record to append (used with `append`). "
                        "Should contain keys like case_summary, diagnosis
    , feedback, reasoning."
                    ),
                    "properties": {
                        "case_summary": {
                            "type": "string",
                            "description": "Brief summary of the patient
    case.",
                        },
                        "diagnosis": {
                            "type": "string",
```

```
                            "description": "The diagnosis that was made."
    ,
                        },
                        "feedback": {
                            "type": "string",
                            "description": (
                                "Whether the diagnosis was correct or
    incorrect, "
                                "and the ground truth."
                            ),
                        },
                    },
                },
                "indices": {
                    "type": "array",
                    "items": {"type": "integer"},
                    "description": (
                        "Indices of short-term cases to evict (used with
    `pop`)."
                    ),
                },
                "rules": {
                    "type": "array",
                    "items": {"type": "string"},
                    "description": (
                        "Diagnostic rules to add to long-term memory "
                        "(used with `consolidate`). "
                        "Each rule should be a concise, reusable
    statement "
                        "(e.g. symptom-disease associations)."
                    ),
                },
            },
            "required": ["action"],
        },
    },
}
```

E.2 Training Details

Compute and infrastructure. Training is conducted on 8× NVIDIA A100 GPUs. We build our training pipeline on top of the rLLMs infrastructure, which provides distributed rollout sampling, reward computation, and optimizer sharding for efficient policy optimization. The full training consumes about 48 GPU hours for Qwen3-8b base model.

Rollout and truncation. Each training example corresponds to one round t with patient case x_t and candidates $\mathcal{Y}_t$. We cap the maximum number of generation steps and tool calls per round to 10 turns. A rollout terminates when the model emits a final diagnosis action $\hat{y}_t$. The prompt exceed the max prompt length will be truncate.

Training Parameters. We adopt Qwen3-4B and Qwen3-8b as the backbone and optimize it with a low learning rate (1e-6) under a PPO framework, using a relatively loose clipping threshold (0.28) and sequence-level loss aggregation. Training is conducted with moderate batch sizes (32 for training) and long context windows (up to 16k tokens for both prompt and response). Rollouts are generated using vLLM in asynchronous mode, with 8 samples per prompt during training (temperature 0.7) and deterministic evaluation settings (1 sample, temperature 0.6, top-p 0.95) for validation. The system runs on a single node with 8 GPUs, incorporating efficiency optimizations such as gradient checkpointing, controlled GPU memory utilization (0.85), and a token limit of 32k per GPU for PPO. Additionally, stepwise advantage estimation is enabled in per-step mode to provide finer-grained credit assignment during training.

Implementation notes. We keep decoding settings fixed as default across training and evaluation to reduce confounding factors (temperature, max tokens, and stop conditions), and we checkpoint the best model based on validation accuracy.

F Automatic Evaluation Details

F.1 Evaluation Prompts Templates

F.1.1 Standard Evaluation Setting

Standard Evaluation Prompt

Role: You are a clinical reasoning evaluator tasked with diagnosing a patient based strictly on provided evidence.

Task:
Given a **Patient Profile** and a list of **Candidate Diseases** (each with descriptions), determine the **single most likely final diagnosis**.

Core Principles:

- This is a **closed-set diagnostic task**: the answer must be selected from the provided Candidate Diseases list.
- Decisions must rely on **explicit evidence** from the Patient Profile.
- Prioritize **clinical consistency, specificity, and minimal contradictions**.

Strict Rules (must follow):

1. Select **EXACTLY ONE** diagnosis from the Candidate Diseases list.
2. Do **NOT** invent diagnoses or use synonyms not appearing in the list.
3. Output the diagnosis name **EXACTLY as written**.
4. Use **ONLY** information from the Patient Profile:
   - Do not assume missing symptoms, labs, or history.
   - Do not rely on external knowledge beyond the provided descriptions.
5. If multiple candidates are plausible, choose the one with:
   - The **most specific and comprehensive match**.
   - The **fewest contradictions**.
6. Prefer diagnoses that explain **key distinguishing features** (e.g., critical symptoms, lab findings, temporal patterns).
7. Do **NOT** output multiple answers, uncertainty, or extra commentary.

Reasoning Requirements:

- Cite **key evidence** from the Patient Profile (symptoms, history, labs, timeline).
- Justify why the selected diagnosis **fits best**.
- Optionally explain why close alternatives are less suitable.
- Keep reasoning **concise, evidence-grounded, and non-speculative**.

Output Format (strict):
Return **ONLY** a valid JSON object (no markdown, no extra text):

```
{
"reasoning": "Step-by-step evidence-based justification.",
"final_diagnosis": "EXACT disease name from the candidate list"
}
```

Input:
Patient Profile:
`{current_case_prompt}`
Candidate Diseases and Descriptions:
`{current_choices}`

Standard Evaluation Prompt Template (ReAct)

Role: You are a clinical reasoning evaluator. You solve diagnosis tasks via iterative reasoning and evidence grounding.

Task:
Given a **Patient Profile** and **Candidate Diseases**, determine the **single most likely final diagnosis**.

Available Information:
- **Patient Profile**: Current case information (symptoms, history, labs).
- **Candidate Diseases**: Closed-set diagnosis options with descriptions.

ReAct Reasoning Process: You must follow an iterative reasoning format:
- **Thought**: Analyze current evidence and identify key clinical signals.
- **Action**: Examine the patient profile and compare against candidate diseases.
- **Observation**: Extract useful insights and match findings to candidate descriptions.

Iteratively repeat the sequence of Thought, Action, and Observation, where each cycle refines your understanding of the case and progressively grounds the diagnosis in evidence.
Then produce a final reasoning summary and diagnosis.

Strict Rules (must follow):
1. Select **EXACTLY ONE** diagnosis from the Candidate Diseases list.
2. Do **NOT** invent diagnoses or use synonyms not in the list.
3. Output the diagnosis name **EXACTLY as written**.
4. Use **ONLY** information from the Patient Profile.
5. Do **NOT** assume missing facts.
6. Choose the diagnosis with:
    - strongest evidence match,
    - highest specificity,
    - minimal contradictions.

Reasoning Requirements:
- Ground decisions in **explicit patient evidence**.
- Prefer concise, structured, and evidence-based reasoning.

Output Format (strict):
Return **ONLY** a valid JSON object (no markdown, no extra text):

```
{
"reasoning": "Condensed reasoning summary integrating patient evidence",
"final_diagnosis": "EXACT disease name from the candidate list"
}
```

Input:
Patient Profile:
`{current_case_prompt}`
Candidate Diseases and Descriptions:
`{current_choices}`

Standard Evaluation Prompt Template (Memory-Augmented)

Role: You are a clinical reasoning evaluator with access to external memory. You solve diagnosis tasks via iterative reasoning and evidence grounding.

Task:
Given a **Patient Profile**, **Candidate Diseases**, and **Memory**, determine the **single most likely final diagnosis**.

Available Information:

- **Patient Profile**: Current case information (symptoms, history, labs).
- **Candidate Diseases**: Closed-set diagnosis options with descriptions.
- **Memory**: Past cases and history.

ReAct Reasoning Process: You must follow an iterative reasoning format:

- **Thought**: Analyze current evidence and identify key clinical signals.
- **Action**: Decide how to use memory (e.g., retrieve relevant cases or rules).
- **Observation**: Extract useful insights from memory or the patient profile.

Iteratively repeat the sequence of Thought, Action, and Observation, where each cycle refines your understanding of the case and progressively grounds the diagnosis in evidence.
Then produce a final reasoning summary and diagnosis.

Memory Usage Guidelines:

- Use memory as **supporting evidence**, not as a substitute for the Patient Profile.
- Prefer memory entries that match **key attributes** (symptoms, labs, progression).
- **Do NOT override** patient evidence with memory if they conflict.
- Long-term memory (rules) should guide abstraction; short-term memory (cases) should support analogy.

Strict Rules (must follow):

1. Select **EXACTLY ONE** diagnosis from the Candidate Diseases list.
2. Do **NOT** invent diagnoses or use synonyms not in the list.
3. Output the diagnosis name **EXACTLY as written**.
4. Use **ONLY** information from the Patient Profile and provided Memory.
5. Do **NOT** assume missing facts.
6. Choose the diagnosis with:
    - strongest evidence match,
    - highest specificity,
    - minimal contradictions.

Reasoning Requirements:

- Ground decisions in **explicit patient evidence**.
- Incorporate **relevant memory** when helpful.
- Prefer concise, structured, and evidence-based reasoning.

Output Format (strict):
Return **ONLY** a valid JSON object (no markdown, no extra text):

```
{
"reasoning": "Condensed reasoning summary integrating patient evidence",
"final_diagnosis": "EXACT disease name from the candidate list"
}
```

Input:
Patient Profile:
`{current_case_prompt}`
Candidate Diseases and Descriptions:
`{current_choices}`
Memory: `{short_term_memory}`

F.1.2 Long-Horizon Evaluation Setting

Long-Horizon Prompt Template

Role: You are a clinical reasoning agent with access to memory, operating in a sequential evaluation setting.

Task: At round t, given the current **Patient Profile**, **Candidate Diseases**, and **Memory**, determine the **single most likely diagnosis**.

Setting:

- Cases arrive sequentially; each case is observed once.
- After each prediction, you receive **feedback** (e.g., correct diagnosis).
- You may update and use **memory** to improve future decisions.

Memory Usage Guidelines:

- Use memory as **supporting evidence**, not a replacement for patient data.
- Do **NOT override** patient evidence with memory if they conflict.

Strict Rules:

1. Select **EXACTLY ONE** diagnosis from the Candidate Diseases.
2. Output the diagnosis name **EXACTLY as written**.
3. Base decisions on Patient Profile, optionally supported by Memory.
4. Do **NOT** assume missing information.

Reasoning Requirements:

- Ground reasoning in explicit patient evidence.
- Incorporate relevant memory when beneficial.
- Clearly connect evidence (and memory, if used) to the diagnosis.

Output Format (strict):

```
{
"reasoning": "Concise reasoning integrating patient
evidence and relevant memory.",
"final_diagnosis": "EXACT disease name"
}
```

Memory:
$\{\mathcal{M}\}$
Input (Round t):
Patient Profile:
$\{x_t\}$
Candidate Diseases:
$\{\mathcal{Y}_t\}$

Long-Horizon Prompt Template (ReAct)

Role: You are a clinical reasoning agent with access to memory, operating in a sequential evaluation setting. You solve each diagnosis task through iterative reasoning grounded in patient evidence and relevant prior experience.

Task: At round t, given the current **Patient Profile**, **Candidate Diseases**, and **Memory**, determine the **single most likely diagnosis**.

Setting:
- Cases arrive sequentially; each case is observed once.
- After each prediction, you receive **feedback** (e.g., correct diagnosis).
- You may update and use **memory** to improve future decisions.

ReAct Reasoning Process: You must reason in an iterative format:
- **Thought**: Analyze the current case and identify key clinical signals, uncertainties, or distinguishing findings.
- **Action**: Decide whether to inspect the patient profile further, compare candidate diseases, or consult relevant memory.
- **Observation**: Record the useful evidence obtained from the patient profile or memory, and refine the diagnostic hypothesis.

Iteratively repeat the sequence of **Thought**, **Action**, and **Observation** as needed, where each cycle refines your understanding of the case and progressively grounds the diagnosis in evidence.

Memory Usage Guidelines:
- Use memory as **supporting evidence**, not a replacement for patient data.
- Do **NOT** override patient evidence with memory if they conflict.

Strict Rules:
1. Select **EXACTLY ONE** diagnosis from the Candidate Diseases.
2. Output the diagnosis name **EXACTLY as written**.
3. Base decisions on the Patient Profile, optionally supported by Memory.
4. Do **NOT** assume missing information.
5. Do **NOT** invent diagnoses or use synonyms not appearing in the candidate list.

Reasoning Requirements:
- Ground reasoning in explicit patient evidence.
- Incorporate relevant memory only when it provides useful support.
- Clearly connect the selected diagnosis to the strongest supporting evidence.
- Prefer concise, structured, and non-speculative reasoning.

Output Format (strict):

```
{
"reasoning": "Concise reasoning integrating patient evidence
and relevant memory.",
"final_diagnosis": "EXACT disease name"
}
```

Memory:
$\{\mathcal{M}\}$

Input (Round t):
Patient Profile:
$\{x_t\}$

Candidate Diseases:
$\{\mathcal{Y}_t\}$

F.2 Baseline Implementation Details

F.2.1 Standard Evaluation Setting

All baselines share the same input format, candidate set $\mathcal{Y}$, and output requirements (reasoning trace + final prediction), ensuring a controlled comparison.

Zero-shot models directly perform inference using a unified prompting template without any parameter updates. **Supervised Fine-Tuning (SFT)** baselines are trained on the MedCaseReasoning training split using next-token prediction over reasoning traces and final answers, with the same data preprocessing and candidate sets as **SEA**. **RL baselines** are optimized with *diagnostic reward only* from raw checkpoint, which provides scalar feedback based solely on final prediction correctness, without intermediate supervision or memory-related rewards. The training configuration (optimizer, batch size, rollout strategy) is kept identical to **SEA** to isolate the effect of reward design.

For **memory-augmented baselines**, we equip models with the same memory tool interface but remove structured memory management. Specifically, *ReAct+ShortTerm-Memory* uses tool calls to retrieve and append past cases but does not perform consolidation into long-term rules. *SFT+ShortTerm-Memory* follows the same setup but leveraging the SFT-trained model.

F.2.2 Long-Horizon Evaluation Setting

In the streaming setting, all methods operate under a single-pass protocol with frozen model parameters. **Zero-shot** and **trained baselines (SFT, RL)** perform inference independently at each step without retaining past information. **Memory-augmented baselines** extend these models with the dual-memory tool at inference time, allowing them to store case records after feedback and retrieve past experiences for future decisions. However, these baselines do not optimize memory usage.

For **GPT-5.2**, we evaluate both zero-shot and memory-augmented variants under the same protocol. The memory-augmented version uses the identical tool schema and system prompt as **SEA**, but without any additional training.

F.2.3 Evaluation protocols

All inputs are fixed before the evaluation starts. For each test case, we prepared 10 cases sampled from the training split as memory context, analogous to few-shot examples. No previously evaluated test instance is written into memory or made available to subsequent test instances. Thus, each test case is evaluated independently with respect to the test set. SEA is able to consolidate the rule from providing short-term memory and store in the long-term memory, but the content in the long-term memory is not carried into the next test case. SEA and other memory-augmented baselines received the same short-term memory. In the ER-Reason long-horizon setting, cases are processed sequentially and feedback is provided after each prediction, allowing memory updates across rounds

G Expert Study Details

We conduct invite professional physician to assess the quality of the updated reasoning and induced rules along four dimensions. Annotator is asked to rate each dimension on a 5-point Likert scale, with higher scores indicating better performance.

Evaluation Criteria.

- **Clinical Correctness.** *Does the updated reasoning and rules remain medically accurate and free of factual errors?*
  - 5: No factual errors; medically sound throughout
  - 4: Very minor factual imprecision with negligible impact

Table 4: Comparison of standard and long-horizon evaluation settings.

Evaluation aspect	Standard evaluation	Long-horizon evaluation
Dataset	MedCaseReasoning	ER-Reason
Training on the target dataset	Yes, using its training split	No
Test-case organization	Cases are evaluated independently	Cases arrive sequentially as a stream
Feedback during evaluation	No	Yes, after each round
Memory update during evaluation	No feedback-based self-learning	Yes, previous experience is incorporated
Primary purpose	Evaluate static diagnostic competence	Evaluate OOD sequential adaptation and experience reuse

  - 3: Noticeable factual issue that could alter interpretation
  - 2: Major factual error that undermines clinical validity
  - 1: Severely incorrect or unsafe factual content
- **Case Relevance.** *Does the reasoning appropriately incorporate case-specific details?*
  - 5: Fully grounded in key case details and diagnostic context
  - 4: Mostly grounded with minor omissions or weak links
  - 3: Partially grounded; several important details underused
  - 2: Weakly grounded; reasoning is largely generic or misaligned
  - 1: Not grounded in the case; off-target reasoning
- **Rule Usefulness.** *Is the induced rule practically useful and likely to be applied?*
  - 5: Strongly like and would readily apply this rule
  - 4: Generally like and would likely use in most cases
  - 3: Neutral; might use in some situations
  - 2: Dislike; unlikely to use except as a last resort
  - 1: Strongly dislike and would not use this rule
- **Overall Trust.** *How reliable is the updated reasoning and rule overall?*
  - 5: Very high trust; can be adopted with confidence
  - 4: High trust; acceptable with minor caveats
  - 3: Moderate trust; requires targeted review
  - 2: Low trust; substantial concerns remain
  - 1: Very low trust; should not be adopted

Annotation Guidelines. Annotator is instructed to consider both the updated reasoning trace and the induced rules jointly. Clinical correctness should be judged strictly based on medical validity, while case relevance emphasizes grounding in the specific patient context. Rule usefulness captures practical applicability, and overall trust reflects a holistic judgment combining all aspects.

Annotator Profile. Annotator is an internal medicine resident with a strong interdisciplinary background in clinical medicine, biostatistics, and oncology research. Recipient of multiple academic distinctions, including national and institutional scholarships.