Context-lite Multi-turn Reinforcement Learning for LLM Agents

Anonymous Authors[1]

Abstract

Finetuning large language model (LLM) agents with multi-turn reinforcement learning (RL) is a promising direction. However, applying multi-turn RL to agentic tasks presents unique challenges not typically encountered in reasoning tasks such as solving math problems. These include long interaction histories that hinder relevant context retrieval, sparse rewards that slow down learning, and variable trajectory lengths that reduce training efficiency. To address these challenges, we propose Context-lite Multi-turn RL, a framework that incorporates: (1) customizable agent memory mechanism, allowing the agent to flexibly include different lengths of historical interaction in each turn's prompt based on task requirements, and (2) Dual-discounting GAE, which decouples step-level and token-level credit assignment. Experiments demonstrate that our method surpasses the zero-shot performance of state-of-the-art LLMs across four BabyAI scenarios, while also achieving greater efficiency and effectiveness than variants lacking either the memory mechanism or dual-discounting GAE.

1. Introduction

Reinforcement learning (RL) has been widely applied to reasoning tasks to enhance the deep thinking capabilities of large language models (LLMs) (Guo et al., 2025; Pan et al., 2025), and recent work has extended RL to multi-turn settings with promising results (Zhou et al., 2025; Chen et al., 2024). However, multi-turn tasks differ significantly from typical reasoning tasks, posing challenges for directly applying existing RL methods. First, during inference, as the number of turns increases, LLM agents struggle to extract task-relevant information from overly long histories (Laban et al., 2025). Second, during training, longer trajectories result in sparser rewards, since reward signals are typically provided only at the end, thereby hindering effective learning. At the system level, large variance in trajectory lengths results in inefficient GPU utilization, as shorter trajectories must wait for longer ones to finish.

To address these issues, we propose a context-lite multi-turn RL framework, which has the following advantages: (1) It supports customizable agent memory mechanism, allowing users to design agent memory mechanisms tailored to specific tasks rather than always using the entire trajectory as input, which we show improves training efficiency and convergent performance in agentic tasks such as BabyAI. (2) It adopts dual discounting GAE for finer-grained credit assignment. Specifically, a larger discount factor is applied to tokens within a turn to encourage extended reasoning, while a smaller discount factor is used across turns to discourage unnecessarily long dialogues. (3) It enables batch training with trajectories of varying lengths, significantly improving GPU utilization.

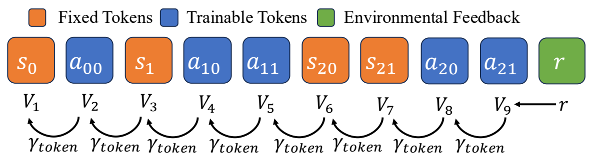
2. Related Works

We compare our method against existing multi-turn RL frameworks for training LLM agents. RAGEN (Wang et al., 2025) supports multi-turn RL but is limited to tasks with short decision horizons (5-10 turns). VeRL (Sheng et al., 2024) enables asynchronous rollouts, improving efficiency when response lengths vary across turns, but does not address challenges posed by a large and variable number of dialogue turns. SkyRL (Cao et al., 2025) supports long-horizon tasks and asynchronous environments, but does not explore efficient memory mechanisms for multi-turn RL. In contrast, our method supports long-horizon, multi-turn tasks, enabling effective credit assignment by using different discount factors (and thus different effective horizons) at the token and step levels.

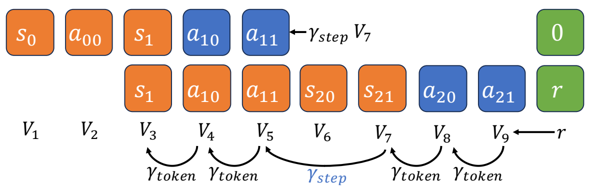
We also notice KIMI K1.5 (Team et al., 2025), a single-turn RL method that handles over-length responses by truncating and storing them in the replay buffer, continuing generation in subsequent training steps. However, this approach is incompatible with PPO-based multi-turn extensions, as PPO is an on-policy algorithm that requires responses to be sampled from the current policy.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Under review by the Workshop on Efficient Systems for Foundation Models at ICML 2025. Do not distribute.



(a) Multi-turn PPO with full memory.



(b) Multi-turn PPO with memory length 1.

Figure 1. (a) Prior work applies PPO at the token level, where each action token a_{ti} is generated based on all preceding information: $s_0, a_0, \cdots, s_t, a_{t0:t(i-1)}$. (b) In contrast, our algorithm limits the context length and introduces a dual discounting strategy for PPO training. Specifically, when computing GAEs, we apply $\gamma_{\text{token}}, \lambda_{\text{token}}$ within individual turns and $\gamma_{\text{step}}, \lambda_{\text{step}}$ across turns, as illustrated by the arrows. Although the first turn does not receive a reward, we can still leverage $\gamma_{\text{step}} V_6$ as a training signal.

3. Preliminary

A Markov Decision Process (MDP) can be described as a tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma, \rho_0)$. Here, $\mathcal{S}$ and $\mathcal{A}$ represent the state and action space, respectively; $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition kernel; $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a reward function; $\gamma \in [0, 1)$ is the discount factor. The goal of RL agents is to learn a policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ that maximizes the expected return:

$$\mathbb{E}_{s_0, a_0, s_1, \cdots}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right], \tag{1}$$

where $s_0 \sim \rho_0(\cdot), a_t \sim \pi(\cdot|s_t), s_{t+1} \sim P(\cdot|s_t, a_t)$. LLM agent tasks involve multiple turns. At each turn t, the agent receives a state prompt s_t (task description + feedback from the previous turn) and outputs a series of action tokens a_t (e.g., tool calling or acting). Agents are fine-tuned via multi-turn RL with verifiable, often sparse rewards to improve sequential decision-making.

4. Methodology

In the following section, we address two key challenges in multi-turn RL training for LLM agents: designing efficient agent memory mechanisms and assigning temporal credit across dialogue turns.

4.1. Context-lite Multi-turn RL

Prior work (Wang et al., 2025) typically treats the entire trajectory (i.e., $\tau = [s_0, a_0, \cdots, s_T, a_T]$) as a single training data point, with the full trajectory as the context, a reward signal at the end, and training only the action tokens (as shown in Figure 1(a)). In particular, the policy is defined and trained as $\pi(a_0, \cdots, a_T|\tau)$, where the historical information $s_0, a_0, \cdots, s_t$ is treated as valid input when generating each action a_t.

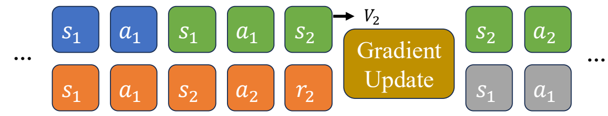


Figure 2. Each trajectory is shown in a different color. For each gradient update, a fixed number of turns are executed. If a trajectory does not terminate within the rollout, the value function of the final state is used as a training signal to guide the LLM agent.

Using such long contexts can lead to inefficient RL training, as it imposes high memory demands and may cause the LLM to lose focus on decision-making at the current time step. In contrast, our framework enables more customizable and granular context usage. Specifically, we treat each turn as an individual training data point, allowing flexible control over how many previous turns are included in the current prompt s_t.

As illustrated in Figure 1(b), we truncate outdated state-action pairs from the trajectory and retain only the most recent *memory length* state-action pairs along with the current state in the context window. The second row in Figure 1(b) demonstrates the case where the memory length is set to one and the data point at each turn t involves (s_{t-1}, a_{t-1}, s_t) as the context and a_t as the action.

Early Trajectory Truncation in PPO Training: When the number of turns in a trajectory exceeds the training batch size, the reward signal may not be immediately available. To address this, our PPO implementation supports early truncation of trajectories, using the value of the final state as a training signal. It is very common for the number of turns in a trajectory to exceed the training batch size. For example, consider a training batch size of 256. To improve inference efficiency, practitioners often increase the number of parallel environments since rollout time becomes the bottleneck in multi-turn RL training. If 16 parallel environments are used, then any task with an episode length exceeding 16 turns may lead to issues in prior frameworks that lack support for early truncation. As shown in Figure 2, this design offers an additional benefit: when trajectory lengths vary significantly, we can truncate trajectories as soon as enough turns have been collected, without waiting for the longest rollout to complete. This improves the overall system throughput.

4.2. Dual Discounting Strategy for Multi-turn RL

In single-turn RL fine-tuning, we typically want to avoid response length shrinkage after training. A common practice is to set the token-level discount factor, γ_{token}, to 1. How-

Table 1. Win Rate (%) across BabyAI tasks for different memory length. The results show that shorter memory lengths (1 to 4) generally lead to higher performance across BabyAI tasks. Values are mean ± standard error.

Memory length	**1**	**2**	**4**	**8**	**16**	**32**	**64**
BabyAI (avg)	31.87±3.68	28.13±3.55	25.00±3.42	17.50±3.00	14.47±2.79	15.72±2.89	18.24±3.06
goto	87.50±5.85	81.25±6.90	50.00±8.84	53.13±8.82	46.88±8.82	37.50±8.56	56.25±8.77
pickup	40.63±8.68	28.13±7.95	25.00±7.65	18.75±6.90	15.63±6.42	28.13±7.95	18.75±6.90
pick_up_seq_go_to	21.88±7.31	21.88±7.31	34.38±8.40	6.25±4.28	6.25±4.28	9.68±5.31	12.90±6.02
open	9.38±5.15	9.38±5.15	15.63±6.42	6.25±4.28	3.13±3.08	3.13±3.08	0.00±0.00

ever, in multi-turn RL fine-tuning, our goal often shifts toward encouraging the agent to complete the task efficiently, minimizing the number of dialogue turns, which can be achieved by using a step-level discount factor $\gamma_{\text{step}} < 1$. Unfortunately, this creates a tension with the need for longer, more coherent reasoning paths, which require more tokens per turn. To address this conflict, we propose a **dual-discounting strategy** for multi-turn RL Generalized Advantage Estimates (GAE) (Schulman et al., 2015) approximation, where we decouple the token-level discount factors $(\gamma_{\text{token}}, \lambda_{\text{token}})$ from the step-level discount factors $(\gamma_{\text{step}}, \lambda_{\text{step}})$, when computing GAE. This approach allows us to independently control reasoning granularity within a step and the overall conversational efficiency across steps. We set $\gamma_{\text{step}} = 0.99, \lambda_{\text{step}} = 0.95, \gamma_{\text{token}} = 1, \lambda_{\text{token}} = 1$ in this work.

With the dual discounting strategy, the GAE formulation is recursively defined as follows:

$$\hat{A}_t = \gamma\lambda\hat{A}_{t+1} + \delta_t^V, \tag{2}$$

where $\gamma\lambda = \gamma_{\text{step}}\lambda_{\text{step}}$ if token t and token $t+1$ are in the different turns and $\gamma\lambda = \gamma_{\text{token}}\lambda_{\text{token}}$ otherwise. δ_t^V, i.e., the TD-residual, is defined as $\delta_t^V = -V(s_t) + r_t + \gamma V(s_{t+1})$, where $V(s_t)$ is the value function. Note that both states and actions consist of multiple tokens. However, the recursive process described in Equation (2) is not applied between state tokens, as states are not generated by the LLM and can be treated as a single chunk of input.

5. Results

In this section, we address the following research questions (RQs): **RQ1**: What is the impact of the memory length on the performance of LLM agents in multi-turn tasks? **RQ2**: Can our proposed multi-turn RL fine-tuning approach improve the performance of LLM agents compared to their zero-shot capabilities? **RQ3**: Does the proposed dual discounting strategy improve value function approximation and lead to improved performance of LLM agents? **RQ4**: How would the memory length impact multi-turn RL fine-tuning for LLM agents? We evaluate our algorithms on four BabyAI (Carta et al., 2023) scenarios, each with a maximum episode length ranging from 64 to 128 steps. In all settings,

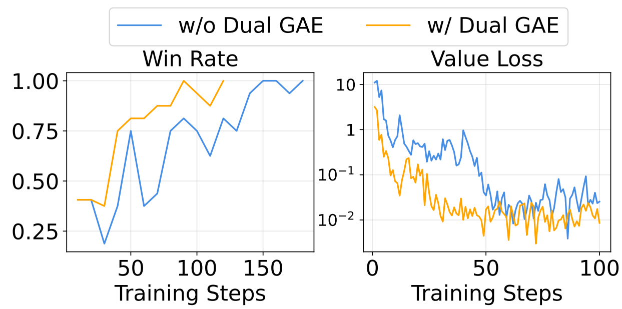


Figure 3. LLM agents trained with dual discounting GAE show faster convergence, and lower PPO value loss on challenging BabyAI *pickup* scenario.

both the policy inputs (observations) and outputs (actions) are represented in text form. The action space is discrete and enumerable.

5.1. Impact of Memory Length on Zero-Shot Capabilities of LLM Agents (RQ1)

We evaluate the zero-shot performance of Qwen-2.5-3B-Instruct in the BabyAI environment using different memory lengths. We define the memory length as the number of previous turns included in the policy's context window. Unlike prior work that defaults to including the entire trajectory history (with memory length fixed at 64 for BabyAI tasks), our implementation enables flexible memory configurations tailored to specific tasks.

As shown in Table 1, this flexibility yields substantial gains: with proper memory length, performance improves by over **2×** compared to the baseline. Interestingly, we observe that simpler tasks, such as *goto* and *pickup*, perform best with memory length 1, while more complex tasks like *open* and *pick_up_seq_go_to* benefit most from memory length 4.

This simple memory mechanism already demonstrates significant potential, highlighting that memory design is a critical yet underexplored component of LLM agents.

5.2. Benchmarking Context-Lite Multi-Turn RL (RQ2)

In this subsection, we evaluate the performance of our proposed method across four distinct BabyAI scenarios and compare it against the zero-shot performance of GPT-4o Mini, LLaMA-3.2-3B-Instruct (Grattafiori et al., 2024), and

Table 2. Comparison of model performance across four BabyAI scenarios. We report the average win rate over 96 trajectories for each method. Our fine-tuned 3B model outperforms the zero-shot performance of larger and comparable-sized models, including GPT-4o Mini, Llama-3.2-3B-Instruct (Grattafiori et al., 2024), and Qwen2.5-3B-Instruct (Yang et al., 2025).

Task	Qwen2.5-3B-Instruct	Llama-3.2-3B-Instruct	GPT-4o mini	Ours
BabyAI(avg)	39.60	32.03	69.53	**86.96**
goto	87.50	56.25	81.25	**100.00**
pickup	40.63	31.25	53.13	**96.88**
pick_up_seq_go_to	21.88	34.38	68.75	**70.83**
open	9.38	6.25	75.00	**78.13**

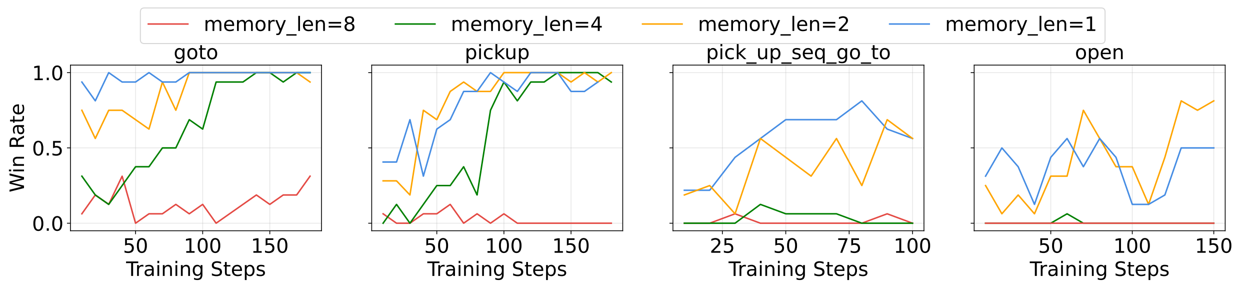


Figure 4. Effect of memory length on multi-turn RL fine-tuning. Shorter memory lengths (1–2) lead to higher performance, likely due to improved zero-shot behavior and denser reward signals. Incorporating limited context supports consistent reasoning while avoiding the inefficiencies introduced by longer histories.

Qwen-2.5-3B-Instruct (Yang et al., 2025). As shown in Table 2, our method fine-tunes a 3B model, Qwen-2.5-3B-Instruct, that ultimately outperforms the larger GPT-4o Mini by 20% on average across all four scenarios.

5.3. Multi-turn RL Benefits from Dual Discounting GAE (RQ3)

We conduct an ablation study on the proposed dual discounting GAE and demonstrate its effectiveness in improving value function estimation and enhancing the sample efficiency of multi-turn RL training for LLM agents. As shown in Figure 3, agents trained with dual discounting outperform those using the baseline configuration ($\gamma_{\text{step}} = \lambda_{\text{step}} = \gamma_{\text{token}} = \lambda_{\text{token}} = 1$) in the BabyAI *pickup* task, exhibiting higher sample efficiency. This improvement is primarily due to the step-level discounting mechanism, which enables more effective temporal credit assignment. Furthermore, dual discounting GAE leads to lower value prediction errors, reflecting a more stable and reliable training process.

5.4. Impact of Memory Length on Multi-turn RL with LLM Agents (RQ4)

In this subsection, we investigate the effect of memory length on multi-turn RL fine-tuning for LLM agents. As shown in Table 4, the LLM agent achieves higher performance when using a shorter memory length during fine-tuning. In particular, memory lengths of one and two consistently yield the highest performance across settings. We observe that longer memory lengths can lead to lower zero-shot performance, which results in sparser reward signals and less efficient RL fine-tuning. Additionally, the agent's reasoning paths often reference or revise plans from previous turns. This behavior appears to enhance planning consistency across turns, suggesting that RL fine-tuning with contextual information is more effective than fine-tuning without context.

6. Conclusion

In this work, we propose Context-lite Multi-turn RL, a framework designed to address key challenges in fine-tuning LLM agents for multi-turn tasks. By introducing customizable memory length and a dual-discounting GAE, our approach tackles issues of long interaction histories, sparse reward signals, and inefficiencies arising from variable-length trajectories. Through extensive experiments on four BabyAI scenarios, we systematically investigate the impact of memory design and discounting strategies on LLM performance and demonstrate the state-of-the-art performance of our algorithm in multi-turn reasoning tasks. A limitation of our approach is that it does not support value-function-free RL fine-tuning methods, such as GRPO (Shao et al., 2024) and RLOO (Ahmadian et al., 2024).

References

Ahmadian, A., Cremer, C., Gallé, M., Fadaee, M., Kreutzer, J., Pietquin, O., Üstün, A., and Hooker, S. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *arXiv preprint arXiv:2402.14740*, 2024.

Cao, S., Hegde, S., Li, D., Griggs, T., Liu, S., Tang, E., Pan, J., Wang, X., Malik, A., Neubig, G., Hakhamaneshi, K., Liaw, R., Moritz, P., Zaharia, M., Gonzalez, J. E., and Stoica, I. Skyrl-v0: Train real-world long-horizon agents via reinforcement learning, 2025.

Carta, T., Romac, C., Wolf, T., Lamprier, S., Sigaud, O., and Oudeyer, P.-Y. Grounding large language models in interactive environments with online reinforcement learning. In *International Conference on Machine Learning*, pp. 3676–3713. PMLR, 2023.

Chen, W., Chen, J., Tajwar, F., Zhu, H., Duan, X., Salakhutdinov, R., and Schneider, J. Fine-tuning LLM agents with retrospective in-context online learning. In *Adaptive Foundation Models: Evolving AI for Personalized and Efficient Learning*, 2024.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Laban, P., Hayashi, H., Zhou, Y., and Neville, J. Llms get lost in multi-turn conversation. *arXiv preprint arXiv:2505.06120*, 2025.

Pan, J., Zhang, J., Wang, X., Yuan, L., Peng, H., and Suhr, A. Tinyzero. https://github.com/Jiayi-Pan/TinyZero, 2025. Accessed: 2025-01-24.

Schulman, J., Moritz, P., Levine, S., Jordan, M., and Abbeel, P. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Sheng, G., Zhang, C., Ye, Z., Wu, X., Zhang, W., Zhang, R., Peng, Y., Lin, H., and Wu, C. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*, 2024.

Team, K., Du, A., Gao, B., Xing, B., Jiang, C., Chen, C., Li, C., Xiao, C., Du, C., Liao, C., et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Wang, Z., Wang, K., Wang, Q., Zhang, P., Li, L., Yang, Z., Yu, K., Nguyen, M. N., Liu, L., Gottlieb, E., et al. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Zhou, Y., Jiang, S., Tian, Y., Weston, J., Levine, S., Sukhbaatar, S., and Li, X. Sweet-rl: Training multi-turn llm agents on collaborative reasoning tasks. *arXiv preprint arXiv:2503.15478*, 2025.