

Enhancing Decision-Making of Large Language Models via Actor-Critic

Heng Dong^{* 1} Kefei Duan^{* 2} Chongjie Zhang²

英寸

Abstract

大型语言模型 (LLMs) 在自然语言处理任务中取得了显著进展,但在复杂的决策场景中遇到挑战,这些场景需要长期推理以及与高层次目标的对齐。现有方法要么依赖于短期的自回归动作生成,要么在准确模拟滚动和评估结果上存在局限性,导致次优决策。本文引入了一种新的基于 LLM 的 Actor-Critic 框架,被称为 LAC,该框架能够以系统化且可扩展的方式有效改善 LLM 策略的长期动作评估。我们的方法解决了两个关键挑战:(1) 通过计算与正面/负面结果相关联的令牌 logits 的 Q 值,提取出稳健的动作评估,增强未来轨迹滚动和推理;以及(2) 通过一种无梯度机制实现高效的策略改进。在不同环境中的实验——包括高层决策 (ALFWorld)、低层动作空间 (BabyAI-Text) 和大型动作空间 (WebShop)——展示了该框架的通用性和优越性,相较于当前最先进的方法尤其突出。值得注意的是,我们的方法使用 7B/8B 参数的 LLMs 即达到竞争性表现,甚至在复杂任务中胜过使用 GPT-4 的基线方法。这些结果强调了将结构化政策优化与 LLMs 的内在知识相结合的潜力,以提升多步环境中的决策能力。

1. 引言

大型语言模型 (LLMs) (Touvron et al., 2023; Jiang et al., 2023; Team et al., 2024) 在各种自然语言处理任务中展现了卓越的能力,包括文本生成 (Li et al., 2024a)、问答 (Li et al., 2024b) 和摘要 (Jin et al., 2024)。LLMs 在这些领域的成功应用,加上其广泛的内部知识,激发了人们极大地关注其在数据稀缺环境中解决复杂决策问题的潜力。

早期的工作 (ichter et al., 2023; Huang et al., 2022a) 已经将 LLM 作为策略,采用自回归的方式生成动作,直接

^{*}Equal contribution ¹ 清华大学 III 系统研究所 ² 圣路易斯华盛顿大学. Correspondence to: Heng Dong <drdhsi@gmail.com>.

利用模型的先验知识进行决策。这些方法对于短期动作生成来说简单而有效,但往往缺乏长期规划的能力。尽管诸如链式思维这样的推理技术已经被引入以增强 LLM (Yao et al., 2023; Shinn et al., 2024) 的推理能力并改善动作选择,但从事全面长期规划的能力仍然不够发达。因此,在更复杂、多步骤的环境中,决策可能看似局部最优,但未能实现整体目标。

其他方法 (Hao et al., 2023; Liu et al., 2023; Fu et al., 2024; Brooks et al., 2024) 结合了规划,或通过与真实环境的交互,或通过利用 LLM 的内部想象能力,然后根据规划结果进行动作评估。这些评估随后用于选择能产生最佳结果的动作。然而,这些方法严重依赖于模型在执行策略推演和评估结果方面的准确性,当模型的推演与现实脱节或动作评估不准确时,会导致次优的动作选择。

这两方面研究的根本限制在于将 LLM 的先验策略和动作评估相分离。它们要么 (1) 僵硬地遵循 LLM 的先验而缺乏足够的规划,要么 (2) 过度依赖可能存在缺陷的模拟推演,二者均未能将 LLM 的固有知识与纠错的规划见解相结合。

为了解决上述问题,我们提出了一种新颖的基于 LLM 的 Actor-Critic (LAC) 方法,该方法通过动作评估在约束条件下优化 LLM 的先验策略,促进将 LLM 的先验知识中固有的决策信息与包含长期规划的强大动作评估中得出的见解有效结合。虽然已经尝试同时利用这两种信息 (LLM 的先验知识和动作评估) 进行决策,但这些努力通常导致简单的组合 (Zhang et al., 2023)。

实现这一理念面临两个主要挑战。首先,如何从 LLM 中提取动作评估信息。一种简单的方法是直接提示 LLM 输出表示动作质量的评估值;然而,这种直接提示方法可能不稳定,而且容易受到提供给 LLM 的指令和示例的影响。为了解决这个问题,我们使用特定标记的 logits 来计算 Q 值 (动作值)。具体来说,我们采用传达正面或负面意义的标记作为 LLM 关于动作质量或其成功完成任务的可能性的内在信念。我们推导出一个简单但有效的公式来利用这些 logits 进行 Q 值计算。此外,为了提供长期规划能力并提高这一计算的准确性,我们针对每个候选动作执行展开以预测未来轨迹,基于这些轨迹进行简短的推理过程,然后收集正面/负面标记的 logits 来计算 Q 值。

第二个挑战是如何利用这些动作评估见解来优化策略。一种直接的方法可能涉及基于梯度的 actor-critic 更新;然而,对于大型语言模型 (LLMs),这种方法由于计

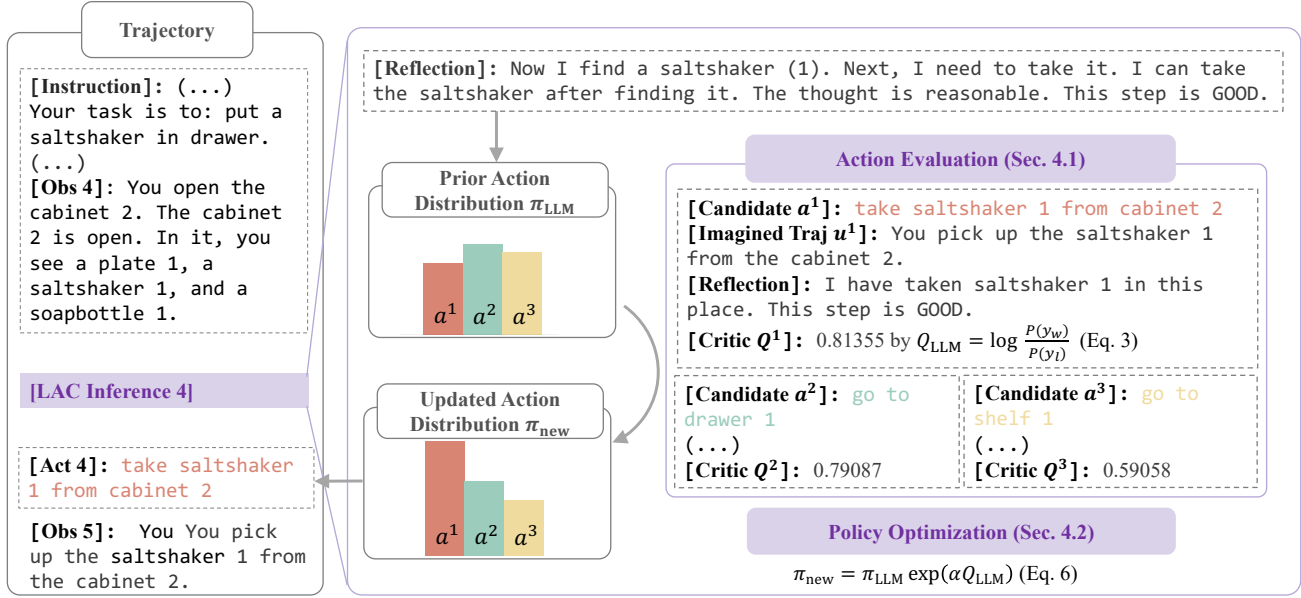


Figure 1. 我们 LAC 的框架。在每个时间步骤中，LAC 通过两个步骤优化策略：（1）评论者 Q_{LLM} 评估从策略 π_{LLM} 采样的每个候选动作；（2）根据动作评估使用无梯度方法优化策略 π_{LLM} 。

算需求高而效率低下，导致决策延迟增加。为了解决这一挑战，我们将策略改进问题表述为一个 KL 散度优化问题，并推导出一个闭式解，从而使我们能够以无梯度的方式优化策略，并提高决策的准确性。

通过实证研究，我们展示了 LAC 在不同环境中的有效性和通用性，包括高层决策任务（ALFWorld、(Shridhar et al., 2021)）、低层动作空间（BabyAI-Text、(Carta et al., 2023b)）以及可能无限的动作空间（WebShop、(Yao et al., 2022)）。结果显示，我们的方法始终优于最新的先进方法，包括 e.g.、RAP (Hao et al., 2023)、LATS (Zhou et al., 2024)。值得注意的是，在几个具有挑战性的任务中，使用 7B/8B LLM 的 LAC 显著超越了带有 GPT-4 (Achiam et al., 2023) 的 ReAct (Yao et al., 2023)。

我们的贡献有两点：（1）我们提出了一种新颖的 Q-函数估计方法，以从 LLMs 中提取动作评估信息，该方法利用了 LLMs 对当前任务成功或失败的内部信念；（2）我们将策略改进问题表述为一个受 KL 散度约束的优化问题，并推导出一个闭合形式的解，使我们能够在不使用梯度的情况下通过动作评估优化策略。

2. 相关工作

用于序列决策的大型语言模型 序列决策是智能代理的关键能力，涉及生成动作以实现目标 (Barto et al., 1989; Littman, 1996; McCarthy et al., 1963; Bylander, 1994)。最近，基于大语言模型 (LLM) 的代理在各个领域的决策中获得了广泛的关注，只需说明或少量示例即可适应新的任务 (Huang et al., 2022b; Singh et al., 2023; Ding et al., 2023)，这得益于在大型数据集上的预

训练。以前的工作主要将 LLM 分为两个角色：政策，用于从轨迹中生成动作；以及评论员，用于根据轨迹和动作评价动作。研究分为两类：第一类直接使用 LLM 从以前的轨迹中生成动作 (ichter et al., 2023; Huang et al., 2022b; Yao et al., 2023; Huang et al., 2022a; Shinn et al., 2024)。然而，LLM 的自回归特性限制了其长期规划能力，使其难以处理复杂任务 (Huang & Chang, 2022; Mialon et al., 2023)。第二类使用另一个 LLM 通过模拟结果来评价动作，并选择最佳方案 (Hao et al., 2023; Liu et al., 2023; Fu et al., 2024; Koh et al., 2024)。虽然这些方法可以实现长期规划，但它们高度依赖于评价准确性，这可能导致次优解决方案。

为了解决这些问题，我们建议在约束条件下使用行动评估来优化 LLM 的先验策略，从而增强长期规划同时减轻评估不准确。之前尝试将 LLM 的先验策略和行动评估 (Zhang et al., 2023) 结合的方法过于简单且缺乏理论保证，导致了不理想的表现。

具有强化学习的大型语言模型 传统的序列决策方法，例如强化学习 (RL)，已经在具身环境中广泛应用 (Schulman et al., 2017; Fujimoto et al., 2018; Huang et al., 2020; Dong et al., 2022)。然而，这些基于 RL 的方法通常在样本上效率不高，需要大量样本进行训练。另一方面，包含丰富世界知识的 LLM 可能缓解这种负担。为了结合 RL 和 LLM，一种直接的方法是将 LLM 用作基本模型，并在 LLM 上添加策略/价值头 (Carta et al., 2023a; Tan et al., 2024)。然后，可以使用像 PPO 这样的经典 RL 方法进行训练 (Schulman et al., 2017)。然而，这些方法仍然需要大量相同任务的训练样本，这在某种程度上降低了使用 LLM 的好处，并与我们的设

置相矛盾。还有其他结合的范式。RLEM (Zhang et al., 2024) 采用 Q 学习 (Watkins & Dayan, 1992) 和经验记忆来更新策略, 但在诸如 ALFWorld 和 BabyAI-Text 等极度稀疏奖励的任务中可能会陷入困境。Retroformer (Yao et al.) 通过 PPO 训练一个较小的 LLM 来为特定任务生成合适的提示, 以便更大的 LLM 使用, 而我们的方法只需要一个小模型。ICPI (Brooks et al., 2024) 通过预测未来轨迹和累积未来奖励来实现策略迭代, 但在稀疏奖励设置中也可能遇到困难。我们在 Section 5 中进行了实证比较。

3. 初步 & 前期工作

在本节中, 我们描述了任务设置和之前的 LLM 决策方法, 以便更好地理解。

任务设置。考虑一个通用的设置, 其中一个智能体与环境交互以达到给定的目标 g , e.g., $g =$ “把一个干净的鸡蛋放进微波炉” (来自 ALFWorld) 或 $g =$ “捡起绿色的球” (来自 BabyAI-Text)。在时间步 t , 智能体从环境中接收到一个自然语言描述的观察 $o_t \in \mathcal{O}$ 。接着, 智能体采取一个行动 $a_t \in \mathcal{A}$, 该行动从策略 $\pi(a|g, h_t)$ 中采样, 其中 $h_t := (o_1, a_1, o_2, a_2, \dots, o_t)$ 是智能体的历史。在执行过程中, 没有即时的奖励, 只有在每个回合结束时, 环境才会给予一个信号来评估任务的完成情况。对于每个测试任务, 智能体只能尝试一次, 不能通过反复试验进行改进。

直接使用 LLM 先验作为策略的方法。为了用 LLM 解决上述任务, 一个简单的方法是直接使用 LLM 的先验作为策略: $a_t \leftarrow \arg \max_a \pi_{\text{LLM}}(a|g, h_t)$, 这可以通过如 Yao et al. (2023) 中建议的那样简单地将指令或少量示例注入提示来实现。尽管简单, 策略 π_{LLM} 生成的动作仅依赖于其自回归能力, 并没有明确进行长期规划, 这通常是顺序决策任务所必需的。此外, 当使用像 CodeLlama-7B (Roziere et al., 2023) 和 Mistral-7B (Jiang et al., 2023) 这样的轻量级模型时, 这个问题会更加严重。这个问题在 Section 5 中得到了验证。

结合规划和行动评估的方法。为了应对缺乏长期规划的问题, 另一条研究途径是将规划和行动评估纳入决策过程 (Hao et al., 2023; Liu et al., 2023; Fu et al., 2024)。基本思路是首先从策略 $\{a_t^1, a_t^2, \dots, a_t^n\} \sim \pi_{\text{LLM}}(\cdot|g, h_t)$ 中采样若干候选行动, 然后由其他 LLM 评估每个候选行动, 最终选择评价价值最高的行动。评估过程是这些方法的关键, 并且可以采用多种方法。例如, 可以直接要求 LLM 评估行动候选 (Fu et al., 2024), 或者通过将 LLM 视为前向世界模型 f_{LLM} , 预测每个行动候选的未来轨迹 u_t , 并将未来结果用作评估, 或者使用蒙特卡洛树搜索 (MCTS) (Kocsis & Szepesvári, 2006; Coulom, 2006) 等树搜索方法扩展每个行动候选 (Hao et al., 2023)。尽管有这些进展, 这些方法在执行推演和评估结果时, 仍严重依赖于模型的准确性。如果推演与现实背离且评估不准确, 这在使用轻量级 LLM 时可能较为常见, 则行动选择可能不是最优的。

Algorithm 1 LAC: 基于大型语言模型的演员-评论家算法。

- 1: **Input:** current task goal g , history h_t , actor π_{LLM} , forward model f_{LLM} , value-based critic Q_{LLM} , hyperparameter α , candidate action size n .
- 2: **Output:** selected action a_t^*
- 3: $\{a_t^i\}_{i=1}^n \sim \pi_{\text{LLM}}(\cdot|g, h_t)$; ▷ 候选动作
- 4: **for** $i = 1$ **to** n **do**
- 5: $u_t^i \leftarrow f_{\text{LLM}}(g, h_t, a_t^i)$; ▷ 预测未来轨迹
- 6: $Q_{\text{LLM}}(g, h_t, a_t^i, u_t^i) \leftarrow \log \frac{P(y_w|g, h_t, a_t^i, u_t^i)}{P(y_l|g, h_t, a_t^i, u_t^i)}$; ▷ 动作评估 (第 4.1 节)
- 7: **end for**
- 8: $\pi_{\text{new}}(a_t^i|g, h_t) \leftarrow \pi_{\text{LLM}}(a_t^i|g, h_t) \exp(\alpha Q_{\text{LLM}}(g, h_t, a_t^i, u_t^i))$; ▷ 策略优化 (第 4.2 节)
- 9: $a_t^* \leftarrow \arg \max_{a_t^i} \pi_{\text{new}}(a_t^i|g, h_t)$

4. 方法

在本节中, 我们介绍了我们新颖的基于 LLM 的 Actor-Critic (LAC) 算法, 该算法以一种有原则且可扩展的方式有效整合了 LLM 的生成能力 (作为 actor) 和其评估推理能力 (作为 critic)。由基于 LLM 的先验策略 (π_{LLM}) 表示的 actor 生成潜在行动, 而 critic (Q_{LLM}) 通过结合长远推理来评估这些行动。这种协同作用通过一种创新的无梯度优化方法实现了有效的策略改进。

我们首先解释如何利用 LLMs 提取有意义的行动评估信息 (Section 4.1)。在此基础上, 我们提出了一种无需梯度的策略优化方法, 该方法能够根据评论者的反馈高效地改进策略 (Section 4.2)。算法的整体工作流程总结在 Algorithm 1 中。

4.1. 长远动作评估的评论者开发

我们提出了一种新颖的方法来构建评论器 Q_{LLM} , 旨在提供对从策略 $\pi_{\text{LLM}}(\cdot|g, h_t)$ 采样的候选动作的数值评估。该方法针对的是以目标为导向的决策问题, 这类问题的特点是奖励稀疏和结果二元化, 只有在实现目标时代理才会得到奖励。通过将 Q_{LLM} 与代理完成任务的成功概率联系起来, 我们能够更有效地指导其朝着最大化期望回报的方向前进。此外, 我们展示了如何直接从与正、负结果相关的标记对数中得出此评估。

4.1.1. 将评论家与成功概率连接

令 $Q_{\text{LLM}}(g, h_t, a_t^i)$ 为每个候选动作 a_t^i 基于任务目标 g 和历史 h_t 的价值评估。我们考虑具有稀疏奖励的场景, 奖励仅在每集结束时提供。考虑到二元结果, 我们希望 $Q_{\text{LLM}}(g, h_t, a_t^i)$ 能够有效地反映实现成功的可能性。我们采用一个逻辑函数 (Jordan et al., 1995) 来将 $Q_{\text{LLM}}(g, h_t, a_t^i)$ 与成功概率关联起来。令 $P(y_w|g, h_t, a_t^i) \in [0, 1]$ 表示在执行动作 a_t^i 后成功完成任务目标 g 的概率, 其中 y_w 表示在集结束时的成功信

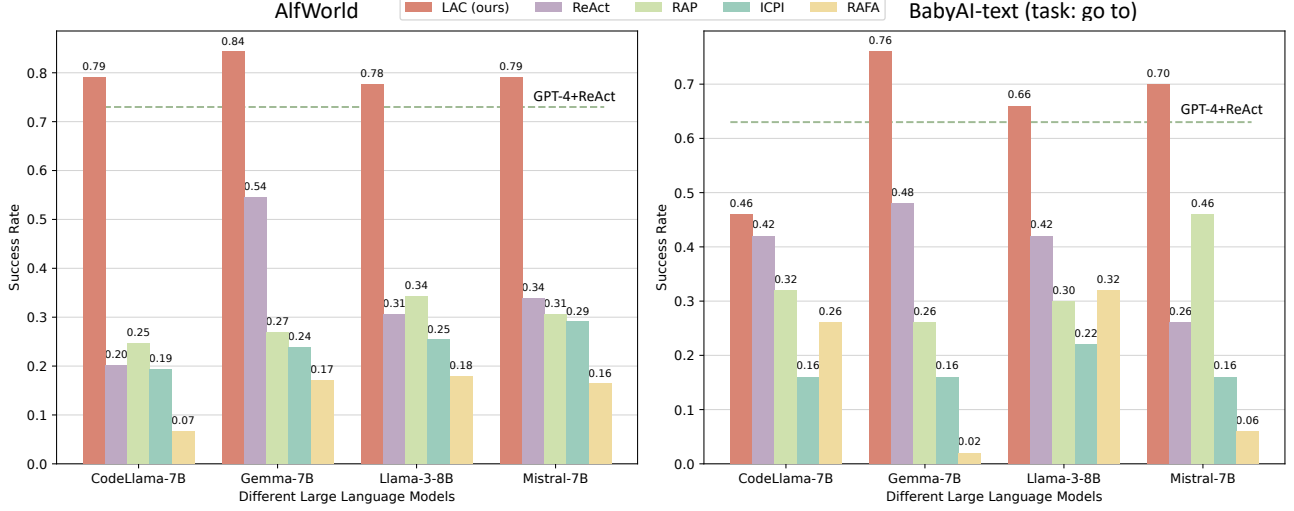


Figure 2. 我们 LAC 在基准测试 ALFWorld 和 BabyAI-Text 中与各种基线的性能比较。

号。同样，令 $P(y_l|g, h_t, a_t^i)$ 表示失败概率。我们使用以下公式来连接 $P(y_w|g, h_t, a_t^i)$ 和 $Q_{LLM}(g, h_t, a_t^i)$ ：

$$P(y_w|g, h_t, a_t^i) = \frac{1}{1 + \exp(-Q_{LLM}(g, h_t, a_t^i))}. \quad (1)$$

使用该公式， $Q_{LLM}(g, h_t, a_t^i)$ 与成功概率 $P(y_w|g, h_t, a_t^i)$ 正相关。较高的 $Q_{LLM}(g, h_t, a_t^i)$ 值映射到更大的成功可能性，从而使评论者能够引导策略朝向最大化长期回报的动作。通过这种方式，最大化 Q 函数对应于最大化当前任务的成功概率。

虽然可以使用其他公式，但我们发现 Equation (1) 对于广泛的任务来说既简单又有效。关于其他公式的对比，请参阅 Appendix A.2 和 Appendix A.3。

为了利用 LLMs 估计 $Q_{LLM}(g, h_t, a_t^i)$ ，我们对 Equation (1) 进行了等效变换：

$$Q_{LLM}(g, h_t, a_t^i) = \log \frac{P(y_w|g, h_t, a_t^i)}{P(y_l|g, h_t, a_t^i)}, \quad (2)$$

，其使用方程式 $P(y_w) + P(y_l) = 1$ 。通过 Equation (2)，我们可以使用 LLM 来获得 $Q_{LLM}(g, h_t, a_t^i)$ ，首先通过估计 $P(y_{\{w,l\}}|g, h_t, a_t^i)$ 。基本思路是提示 LLM 在给定当前轨迹 (g, h_t) 和动作 a_t^i 的情况下预测结果。具体来说，我们使用具有正面/负面意义的特殊配对标记 *i.e.*，如 “GOOD” / “BAD” 或 “SUCCESS” / “FAILURE”，以指示成功/失败结果。LLM 为这些特殊标记生成的相应概率反映了 LLM 在采取动作 a_t^i 后的内部成功/失败信念。我们让 “GOOD” 和 “BAD” 生成的概率分别表示 $P(y_w|g, h_t, a_t^i)$ 和 $P(y_l|g, h_t, a_t^i)$ 。最后，使用 Equation (2)，我们可以计算动作为 a_t^i 的 $Q_{LLM}(g, h_t, a_t^i)$ 。虽然我们的动作评估是为二元结果设计的，但我们实验表明，这种公式在连续奖励设置中也是有效的，如在 Section 5 中所示。

为了提高 $Q_{LLM}(g, h_t, a_t^i)$ 的准确性，我们引入了使用前向世界模型 f_{LLM} 的未来轨迹展开，这可以通过提示 LLMs，*e.g.*，添加几个样例，或者通过这些样例上进行微调来实现。对于每个候选动作 a_t^i ，我们展开几个未来步骤来预测产生的轨迹 $u_t^i \sim f_{LLM}(g, h_t, a_t^i)$ 。通过考虑未来轨迹 u_t^i ，我们获得了更为明智的成功和失败概率估计 $P(y_{\{w,l\}}|g, h_t, a_t^i, u_t^i)$ 。该方法考虑了动作的延迟后果，并确保 $Q_{LLM}(g, h_t, a_t^i, u_t^i)$ 反映每个动作的长期价值：

$$Q_{LLM}(g, h_t, a_t^i, u_t^i) = \log \frac{P(y_w|g, h_t, a_t^i, u_t^i)}{P(y_l|g, h_t, a_t^i, u_t^i)}. \quad (3)$$

轨迹展开在需要过多个步骤来展开动作结果的任务中特别重要。通过模拟动作的未来影响，评估者可以提供更准确的评估，从而指导策略朝着最大化长期成功概率的动作方向进行。

根据经验，我们发现对先前动作的情境反思有助于提高表现。具体来说，给定任务目标 g 和历史 h_t ，其中可能包含预测的未来轨迹，我们提示 LLMs 生成简短的反思，例如 “我已经找到了对象-X。这一步是好的” 或 “我应该先拿对象-X 而不是对象-Y。这一步是坏的。” 这些反思提供了关于先前动作是否恰当以及为什么恰当的判断。有关这些反思的更多示例，请参见 ?? 的 Table 27 和 Table 28。

这些反思作为简单的理由，类似于链式思维 (CoT) (Wei et al., 2022; Kojima et al., 2022; Prystawski et al., 2024)，通过避免过去的错误，允许策略采样更好的候选动作。这些反思还使评论者能够更准确地评估候选动作，从而最终提高决策性能。关于我们的反思和 CoT 之间的差异比较，请参考 Appendix A.4。

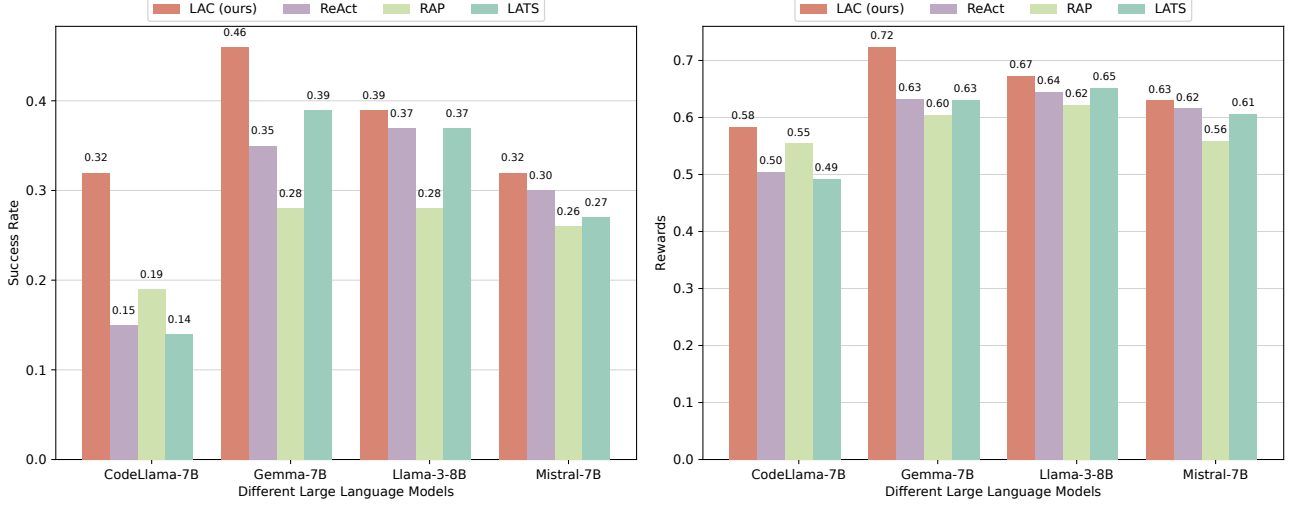


Figure 3. LAC 在基准测试 WebShop 中的表现

4.2. 无梯度策略优化

在本小节中，我们使用上述基于估值的评估推导了一种无梯度的策略优化方法。为了有效改进 LLM 的先验策略，我们提出使用以下具有 KL 约束的策略优化问题来最大化期望值函数：

$$\max_{\pi} \mathbb{E}_{a_t^i \sim \pi(\cdot|g, h_t), u_t^i \sim f_{\text{LLM}}(g, h_t, a_t^i)} [Q_{\text{LLM}}(g, h_t, a_t^i, u_t^i)] - \frac{1}{\alpha} \mathbb{D}_{\text{KL}} [\pi(a_t^i|g, h_t) \| \pi_{\text{LLM}}(a_t^i|g, h_t)], \quad (4)$$

，其中 α 是控制与原始策略 π_{LLM} 偏差的超参数。KL 散度项可以防止新策略 π_{new} 偏离原始策略太远，从而在策略的先验知识和评论者的指导之间取得平衡。

根据之前的工作 (Rafailov et al., 2024; Go et al., 2023; Peng et al., 2019; Jain et al., 2013; Peters & Schaal, 2007)，我们可以证明，在 Equation (4) 中 KL 约束最大化目标的最优解形式如下：

$$\pi_{\text{new}}(a_t^i|g, h_t) = \frac{1}{Z(g, h_t)} \pi_{\text{LLM}}(a_t^i|g, h_t) \cdot \exp(\alpha Q_{\text{LLM}}(g, h_t, a_t^i, u_t^i)), \quad (5)$$

其中 $Z(g, h_t) = \sum_{a_t^i} \pi_{\text{LLM}}(a_t^i|g, h_t) \exp(\alpha Q_{\text{LLM}}(g, h_t, a_t^i, u_t^i))$ 是分区函数。请参考 ?? 获取完整的推导。由于分区函数不依赖于动作 a_t^i ，我们在实践中可以忽略它：

$$\pi_{\text{new}}(a_t^i|g, h_t) \propto \pi_{\text{LLM}}(a_t^i|g, h_t) \exp(\alpha Q_{\text{LLM}}(g, h_t, a_t^i, u_t^i)). \quad (6)$$

我们只需选择具有最大比例 $a_t \leftarrow \arg \max_{a_t^i} \pi_{\text{new}}(a_t^i|g, h_t)$ 的动作。值得一提的是，如果我们在 Equation (6) 中让 $\alpha = 0$ ，我们恢复的方法是直接使用 LLM 的先验作为策略；如果我们让

$\alpha \rightarrow +\infty$ ，我们则恢复那些结合计划和行动评估的方法。

使用 Equation (6) 有两个主要优势。首先，它以一种无梯度方式在评论家 Q_{LLM} 建议的方向上更新策略 π_{LLM} 的动作分布，这实现了政策改进，与基于梯度的方法相比，计算负担要低得多，尤其是当演员由 LLMs 实现时。其次，新策略 π_{new} 的动作分布是过去信息为基础的策略优先项与预测未来信息为基础的评论员后验项的平衡整合。

5. 实验

在本节中，我们对我们的方法 LAC 在三个基准上进行了测试，这些基准覆盖高级动作空间 (ALFWorld (Shridhar et al., 2021))、低级动作空间 (BabyAI-Text (Chevalier-Boisvert et al., 2018)) 和潜在无限动作空间 (WebShop (Yao et al., 2022))。我们通过回答以下问题来评估 LAC 的有效性：(1) LAC 能否优于其他基于 LLM 的方法进行决策？(Section 5.2) (2) LAC 的每个组件如何对其性能做出贡献？(Section 5.3) (3) 不同的大型语言模型如何影响性能？(Section 5.2 和 Section 5.3) (4) 我们的方法在计算上是否耗费资源？(Section 5.4) (5) 为什么 LAC 有效？(Section 5.5)。LAC 的代码在 GitHub¹ 和网站² 上公开可用。

5.1. 实验设置

我们将我们的方法与各种基于大语言模型 (LLM) 的决策方法进行了比较。这里我们简要介绍这些方法，更多详细信息请参见 Appendix B.2。

¹<https://github.com/drdrh/LAC>

²<https://sites.google.com/view/lang-ac>

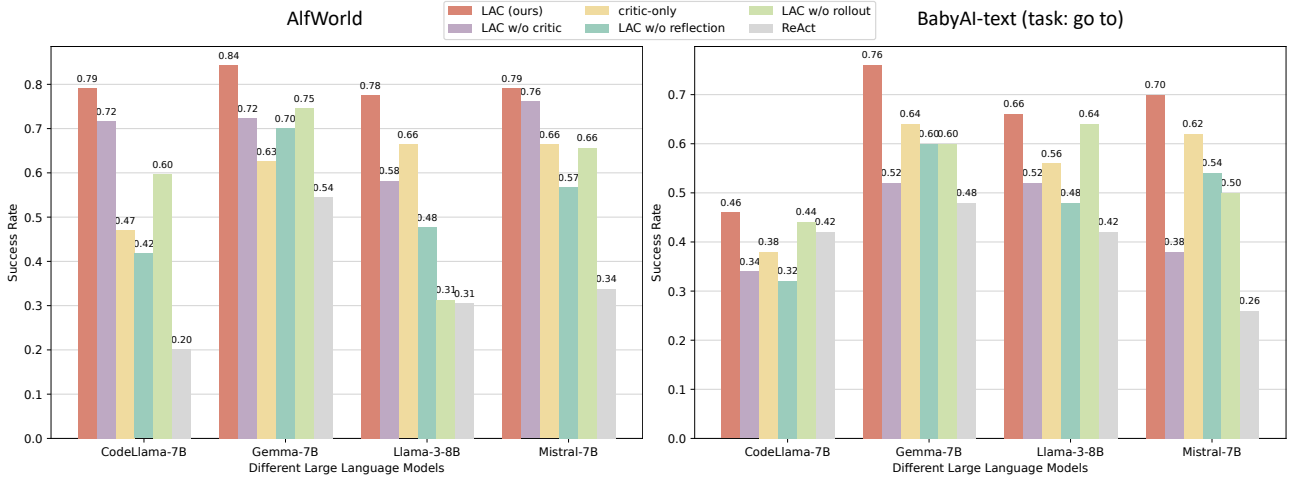


Figure 4. 在基准测试 ALFWorld 和 BabyAI-Text 中的消融研究。

各种基线：(1) ReAct (Yao et al., 2023) 在与环境的互动中结合推理和行动，并利用 LLMs 的推理能力来提高 LLM 正确作为策略行动的概率。(2) RAP (Hao et al., 2023) 利用 LLMs 作为策略和世界模型，并采用树搜索规划方法来评估每个可能的动作候选项。(3) ICPI (Brooks et al., 2024) 通过预测未来轨迹并选择预期累计奖励最高的动作来实施使用 LLMs 的策略迭代。(4) RAFA (Liu et al., 2023) 通过树搜索评估每个动作候选项，并选择可能完成最多子目标的动作。(5) LATS (Zhou et al., 2024) 结合 LLMs 的推理、行动和规划能力与 MCTS (Kocsis & Szepesvári, 2006) 及外部反馈机制，以增强各领域的决策能力，在网络导航中取得具有竞争力的结果。

我们在三个具有稀疏奖励和不同类型动作空间的决策制定基准上评估了 LAC。

高级动作：ALFWorld (Shridhar et al., 2021) 是一个广泛使用的文本型家庭环境，包含 134 项不同的评估任务，需要代理通过一系列高级动作来实现目标，*e.g.* “去某地-X”，“从某地-X 取对象-Y”，*etc.* 如果代理实现目标，则获得 1 的奖励，否则为 0。这个基准的主要挑战在于用大型语言模型的常识知识定位目标对象并完成家务工作。根据 ReAct，我们在任务特定设置下评估所有 134 个未见过的评估游戏。

低级动作：BabyAI-Text (Carta et al., 2023b) 是一个从 BabyAI 平台 (Chevalier-Boisvert et al., 2018) 扩展而来的网格世界环境，其中代理和物体放置在一个 8×8 方块的房间中。代理具有 6 种基本动作：左转、右转、前进、拾起、放下、切换，以解决以自然语言描述的任务 (例如 *e.g.* “拾起红色盒子”)。与 ALFWorld 类似，该基准测试也具有二进制奖励，值为 1 或 0。这些任务可能很困难，因为代理必须制定长期计划，避开障碍物，并根据以自然语言描述的部分观察找到目标物体的

最短路径。

可能的无限动作：WebShop (Yao et al., 2022) 要求代理基于指示 (例如，“我需要自然外观的长发夹式发束，价格低于 20.00 美元”) 通过网络互动购买产品 (例如，搜索“长发夹式发束”，点击诸如 “[商品 ID]” 或 “返回搜索” 等按钮)。在此情况下，“搜索” 和 “点击” 动作确实可以引导至一组无限的潜在动作，因为代理可以基于动态的网络结果持续优化其查询和选择。与 ALFWorld 和 BabyAI-Text 不同，此基准测试中的最终奖励是介于 0 和 1 之间的连续值，取决于最终购买的产品满足要求的程度。

为了展示 LAC 的稳定性，我们采用了来自不同组织的四个开源大型语言模型：CodeLlama-7B (Roziere et al., 2023)，Mistral-7B (Jiang et al., 2023)，Gemma-7B (Team et al., 2024)，和 Llama-3-8B (Meta, 2024a)。

5.2. 主要性能

我们报告了我们的方法 LAC 与其他基线在 Figure 2 (ALFWorld 和 BabyAI-Text)、Figure 3 (WebShop) 和 Figure 6 (更多 BabyAI-Text 任务) 中的比较结果。对于所有实验，我们设置了 LLMs 的温度为 0，因此生成是确定性的。基于这个原因，图中没有误差条。

LAC 在 ALFWorld 和 BabyAI-Text 中均优于其他所有基线，并且在大多数设置中甚至比 GPT-4+ ReAct 更好，这验证了我们方法的有效性和稳定性。在 WebShop 中，LAC 始终在累计奖励和成功率方面胜过其他基线，跨各种基础模型。这进一步证明了我们方法在处理更复杂和开放式动作空间时的鲁棒性。

LAC 的卓越表现源于其有效整合了在 LLM 先验中包含的决策信息，以及从包含长期规划的动作评估中得出的决策见解。为了更好地理解，我们还在 Figure 10

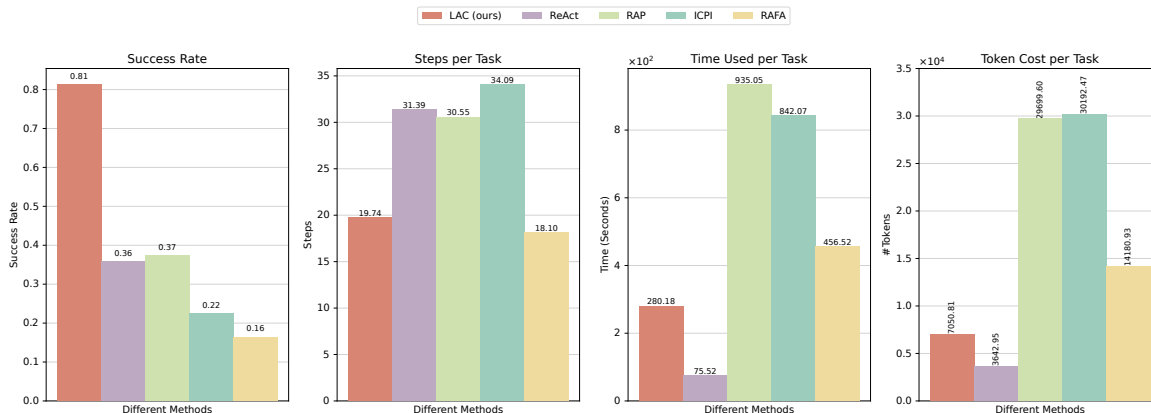


Figure 5. LAC 和基准的计算成本分析。

和 Figure 11 中分别为 ALFWorld 和 BabyAI-Text 提供了示例。总之，单独的策略和评论员可能会在不同时间步骤中犯错误，我们的 LAC 可以通过它们的整合来选择正确的动作。

关于 LAC 在不同基础模型上的性能，我们强调两个关键发现：（1）我们的方法是通用的，可以适应各种基础模型，和（2）更强大的基础模型，例如 Gemma-7B，在与我们的方法结合时表现出更高的性能。然而，由于这些基础模型的训练细节公开程度不完整，进一步的深入分析需要额外的调查。

在此我们仅将 LAC 与基于 LLM 的方法进行比较，关于其他基线、e.g.、策略微调、基于 LLM 的 RL 变体以及决策转换器 (Chen et al., 2021) 的更多结果，请参考 Appendix A.5。

5.3. 消融研究

为了研究 LAC 各个组成部分的贡献，我们进行了详细的消融研究。我们的方法主要由以下几个组成部分：（1）使用评论者提供的动作评估进行策略优化步骤；（2）通过提取 LLMs 的内部信息进行动作评估。因此，为了展示每个组成部分的贡献，我们设计了以下消融研究：（1）LAC w/o critic 从 LAC 中移除策略优化步骤；（2）LAC w/o rollout 在动作评估过程中不预测未来轨迹；（3）LAC w/o reflection 在采样和评估候选动作时移除反思程序；（4）critic-only 仅使用评论者的动作评估信息进行动作选择。

我们在 Figure 4 中报告了结果。在 ALFWorld 和 BabyAI-Text 中，LAC 优于所有其他变体。具体来说，与 LAC 相比，LAC w/o critic 和 critic-only 中的性能下降验证了 LAC 的策略优化结合评论员的动作评估的有效性。而 LAC w/o reflection 和 LAC w/o rollout 比 LAC 表现更差的结果也表明了结合反思和未来轨迹预测的必要性。

5.4. 计算成本分析

我们的方法通过预测未来轨迹来进行动作评估，这可能会在每一步带来额外的计算成本。在 Figure 5 中，我们比较了 LAC 和其他基线在花费的令牌数量和运行时间方面的计算成本。具体来说，尽管由于评论者和前向模型的额外推理过程，LAC 的每步计算成本较高，但 LAC 的总成本仍低于大多数基于 LLM 的基线，因为 LAC 完成每个任务所需的步骤较少。这是因为 LAC 的成功率更高，使得它能够在最大步数限制内完成任务，而其他基线通常在没有完成任务的情况下就已达到此限制。如果我们只考虑成功的任务，方法之间的步骤成本是相似的：LAC：15.32 步，ReAct：17.75 步，RAP：16.36 步。有关其他基线的更多计算成本分析，请参阅 Appendix A.6。

5.5. 统计分析

直观地解释基于 LLM 的决策是本质上具有挑战性的，因为有两个因素：（1）LLM 是黑箱模型，使得直接解释其输出变得困难；（2）在序列决策任务中，通常无法获得分析策略选择的每个动作的真实行动——只能观察到任务级别的成功或失败信号。

因此，我们提供了统计分析，以阐明 LAC 的两个关键组成部分——Q 函数估计和无梯度策略改进——为何在实践中能有效发挥作用。

Q 函数估计反映任务进展 我们使用 LLM 内部信念（“GOOD” / “BAD” 标记的对数概率）来估计 Q 值。为了验证这些 Q 值能否有效跟踪任务成功，我们计算了每条轨迹内的 Q 值与时间步长之间的 Pearson 相关性。直观上，如果代理处于成功路径上，Q 值应随着进程而增加（否则则减少）。Table 1 中的结果，平均跨越 134 个 ALFWorld 任务，证实了这一趋势：成功的轨迹显示出增加的 Q 值，而失败的轨迹则显示出相反的结果。这证明我们的 Q 函数捕捉了整个轨迹中策略成功可能性的变化。

Table 1. Q 值与时间步长之间的皮尔逊相关性。

	Successful Trajectory (1843 steps in total)	Failed Trajectory (1092 steps in total)
$\log P$ (“GOOD”)	0.35 ± 0.18	-0.37 ± 0.19
$\log P$ (“BAD”)	-0.32 ± 0.19	0.38 ± 0.19
Q-value of chosen action	0.34 ± 0.18	-0.41 ± 0.18

Table 2. 策略改进的置信度分析。

Case (Proportion)	Prior Policy Conf.	Q-function Conf.	the Improved Policy Conf.
Both agree (49.81 %)	0.49 (✓)	0.21 (✓)	0.74
Prior agrees, Q disagrees (29.12 % cases)	0.34 (✓)	0.06 (✗)	0.28
Prior disagrees, Q agrees (19.28 % cases)	0.21 (✗)	0.34 (✓)	0.22
Prior disagrees, Q disagrees (1.79 % cases)	0.06 (✗)	0.03 (✗)	0.07

策略改进反映了模型的信心 我们的无梯度策略改进基于相对置信度平衡先验策略和 Q 函数。具体来说，我们将模型的置信度定义为其前两个得分之间的差异：(1) 对于先验策略： $\log P(a_1|s) - \log P(a_2|s)$ ；(2) 对于 Q 函数： $Q(a_1|s) - Q(a_2|s)$ ，其中 a_1 和 a_2 分别是相应模型下最高和次高概率（或 Q 值）的动作。

我们分析了改进策略选择了先验策略或 Q 函数中的哪个动作，以及每个模型在其选择中的信心。Table 2 总结了结果。此分析表明：(1) 当两个模型一致时，改进策略的信心最高；(2) 当它们不一致时，改进策略倾向于信任更自信的模型；(3) 在信心低的情况下，策略保持保守。这些结果暗示我们的方法通过基于模型信心进行加权，隐式地汇集了来自两个来源的决策知识，从而能够在没有显式梯度的情况下实现有效的策略改进。

6. 讨论

在这项工作中，我们介绍了一种新颖的基于 LLM 的演员-评论家算法 LAC，该算法利用动作评估来优化 LLM 的先验策略，从而有效整合从 LLM 先验和包含长期规划的动作评估中得出的决策洞察。与以往的方法相比，LAC 在覆盖各种动作空间的三个基准测试中，即便使用轻量级开源 LLM，也能实现高性能。

尽管取得了进展，我们的方法也有局限性，并启发了未来可能的研究方向。首先，LAC 的反思仅在动作生成之前使用，这也可以应用于动作生成之后。例如，它可以为预测的未来轨迹提供反思，以便在之前的候选动作全部无法完成目标任务时重新采样候选动作。其次，尽管我们为简化和效率每个候选动作只扩展一个节点，并且发现这样行之有效，LAC 也可以采用树搜索为候选动作提供更准确的评估。第三，虽然 LAC 可以通过仅将获得最高奖励视为成功来经验性地处理具有连续最终奖励的场景，但开发一种更根本的方法来处理这种情况是一个令人兴奋的未来方向。最后但同样重要的是，应用于更大和更强大的 LLM 模型时，LAC 的有效性需要进一步研究。

7.

影响声明

我们的方法基于开源的大型语言模型 (LLMs)。和其他使用 LLMs 的方法一样，我们的方法也继承了 LLMs 的一些优势和挑战。在优势方面，我们的方法直接利用了 LLMs 的先验知识，这可能比从头开始训练策略更加减少潜在的碳成本。在挑战方面，我们的方法可能在面对有害输入时容易产生意想不到的输出，例如不道德的文本或用于对抗性攻击的输入。为了解决这个问题，我们建议对我们的方法进行慎重的部署，比如添加一个过滤组件。

References

- Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- Barto, A. G., Sutton, R. S., and Watkins, C. *Learning and sequential decision making*, volume 89. University of Massachusetts Amherst, MA, 1989.
- Brooks, E., Walls, L., Lewis, R. L., and Singh, S. Large language models can implement policy iteration. *Advances in Neural Information Processing Systems*, 36, 2024.
- Bylander, T. The computational complexity of propositional strips planning. *Artificial Intelligence*, 69(1-2): 165–204, 1994.
- 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, 2023a.
- 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, 2023b.
- Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., and Mordatch, I. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.
- Chevalier-Boisvert, M., Bahdanau, D., Lahlou, S., Willems, L., Saharia, C., Nguyen, T. H., and Bengio, Y. Babyai: A platform to study the sample efficiency of grounded language learning. *arXiv preprint arXiv:1810.08272*, 2018.
- Coulom, R. Efficient selectivity and backup operators in monte-carlo tree search. In *International conference on computers and games*, pp. 72–83. Springer, 2006.
- Ding, Y., Zhang, X., Paxton, C., and Zhang, S. Task and motion planning with large language models for object rearrangement. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2086–2092. IEEE, 2023.
- Dong, H., Wang, T., Liu, J., and Zhang, C. Low-rank modular reinforcement learning via muscle synergy. *Advances in Neural Information Processing Systems*, 35: 19861–19873, 2022.
- Fu, D., Huang, J., Lu, S., Dong, G., Wang, Y., He, K., and Xu, W. Preact: Predicting future in react enhances agent’s planning ability. *arXiv preprint arXiv:2402.11534*, 2024.
- Fujimoto, S., Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pp. 1587–1596. PMLR, 2018.
- Go, D., Korbak, T., Kruszewski, G., Rozen, J., Ryu, N., and Dymetman, M. Aligning language models with preferences through f -divergence minimization. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11546–11583. PMLR, 23–29 Jul 2023. URL <https://proceedings.mlr.press/v202/go23a.html>.
- Hafner, D. Benchmarking the spectrum of agent capabilities. In *International Conference on Learning Representations*, 2022. URL <https://openreview.net/forum?id=1W0z96MFEoH>.
- Hao, S., Gu, Y., Ma, H., Hong, J., Wang, Z., Wang, D., and Hu, Z. Reasoning with language model is planning with world model. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 8154–8173, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.507. URL <https://aclanthology.org/2023.emnlp-main.507/>.
- Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models, 2021.
- Huang, J. and Chang, K. C.-C. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*, 2022.
- Huang, W., Mordatch, I., and Pathak, D. One policy to control them all: Shared modular policies for agent-agnostic control. In *International Conference on Machine Learning*, pp. 4455–4464. PMLR, 2020.
- Huang, W., Abbeel, P., Pathak, D., and Mordatch, I. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International Conference on Machine Learning*, pp. 9118–9147. PMLR, 2022a.
- Huang, W., Xia, F., Xiao, T., Chan, H., Liang, J., Florence, P., Zeng, A., Tompson, J., Mordatch, I., Chebotar, Y., et al. Inner monologue: Embodied reasoning through planning with language models. *arXiv preprint arXiv:2207.05608*, 2022b.
- ichter, b., Brohan, A., Chebotar, Y., Finn, C., Hausman, K., Herzog, A., Ho, D., Ibarz, J., Irpan, A., Jang, E., Julian, R., Kalashnikov, D., Levine, S., Lu, Y., Parada, C., Rao, K., Sermanet, P., Toshev, A. T., Vanhoucke, V., Xia, F., Xiao, T., Xu, P., Yan, M., Brown, N., Ahn, M., Cortes, O., Sievers, N., Tan, C., Xu, S., Reyes, D., Rettinghouse, J., Quiambao, J., Pastor, P., Luu, L., Lee, K.-H., Kuang, Y., Jesmonth, S., Joshi, N. J., Jeffrey, K., Ruano, R. J., Hsu, J., Gopalakrishnan, K., David, B., Zeng, A., and Fu, C. K. Do as i can, not as i say: Grounding language in robotic affordances. In Liu, K., Kulic, D., and Ichnowski, J. (eds.), *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 287–318. PMLR, 14–18 Dec 2023. URL <https://proceedings.mlr.press/v205/ichter23a.html>.
- Jain, A., Wojcik, B., Joachims, T., and Saxena, A. Learning trajectory preferences for manipulators via iterative improvement. *Advances in neural information processing systems*, 26, 2013.
- Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

- Jin, H., Zhang, Y., Meng, D., Wang, J., and Tan, J. A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. *arXiv preprint arXiv:2403.02901*, 2024.
- Jordan, M. I. et al. Why the logistic function? a tutorial discussion on probabilities and neural networks, 1995.
- Kocsis, L. and Szepesvári, C. Bandit based monte-carlo planning. In *European conference on machine learning*, pp. 282–293. Springer, 2006.
- Koh, J. Y., McAleer, S., Fried, D., and Salakhutdinov, R. Tree search for language model agents. *arXiv preprint arXiv:2407.01476*, 2024.
- Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213, 2022.
- Li, J., Tang, T., Zhao, W. X., Nie, J.-Y., and Wen, J.-R. Pre-trained language models for text generation: A survey. *ACM Comput. Surv.*, 56(9), April 2024a. ISSN 0360-0300. doi: 10.1145/3649449. URL <https://doi.org/10.1145/3649449>.
- Li, X., Zhou, Y., and Dou, Z. Unigen: A unified generative framework for retrieval and question answering with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8688–8696, 2024b.
- Littman, M. L. *Algorithms for sequential decision-making*. Brown University, 1996.
- Liu, Z., Hu, H., Zhang, S., Guo, H., Ke, S., Liu, B., and Wang, Z. Reason for future, act for now: A principled framework for autonomous llm agents with provable sample efficiency. *arXiv preprint arXiv:2309.17382*, 2023.
- McCarthy, J. et al. *Situations, actions, and causal laws*. Comtex Scientific, 1963.
- Meta. Meta llama 3. <https://llama.meta.com/llama3/>, 2024a.
- Meta. Meta llama 3.1. <https://ai.meta.com/blog/meta-llama-3-1/>, 2024b.
- Mialon, G., Dessi, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Roziere, B., Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., Grave, E., LeCun, Y., and Scialom, T. Augmented language models: a survey. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL <https://openreview.net/forum?id=jh7wH2AzKK>. Survey Certification.
- Paglieri, D., Cupiał, B., Coward, S., Piterbarg, U., Wołczyk, M., Khan, A., Pignatelli, E., Kuciński, Ł., Pinto, L., Fergus, R., Foerster, J. N., Parker-Holder, J., and Rocktäschel, T. Benchmarking agentic llm and vlm reasoning on games. *arXiv preprint arXiv:2411.13543*, 2024.
- Peng, X. B., Kumar, A., Zhang, G., and Levine, S. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.
- Peters, J. and Schaal, S. Reinforcement learning by reward-weighted regression for operational space control. In *Proceedings of the 24th international conference on Machine learning*, pp. 745–750, 2007.
- Prystawski, B., Li, M., and Goodman, N. Why think step by step? reasoning emerges from the locality of experience. *Advances in Neural Information Processing Systems*, 36, 2024.
- Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.
- Roziere, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X. E., Adi, Y., Liu, J., Remez, T., Rapin, J., et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.
- Shridhar, M., Thomason, J., Gordon, D., Bisk, Y., Han, W., Mottaghi, R., Zettlemoyer, L., and Fox, D. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10740–10749, 2020.
- Shridhar, M., Yuan, X., Côté, M.-A., Bisk, Y., Trischler, A., and Hausknecht, M. Alfworld: Aligning text and embodied environments for interactive learning, 2021.
- Singh, I., Blukis, V., Mousavian, A., Goyal, A., Xu, D., Tremblay, J., Fox, D., Thomason, J., and Garg, A. Prog-prompt: Generating situated robot task plans using large language models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11523–11530. IEEE, 2023.

- Tan, W., Zhang, W., Liu, S., Zheng, L., Wang, X., and An, B. True knowledge comes from practice: Aligning large language models with embodied environments via reinforcement learning. In *ICLR*, 2024.
- Team, G., Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.
- Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- Watkins, C. J. and Dayan, P. Q-learning. *Machine learning*, 8:279–292, 1992.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837, 2022.
- Yao, S., Chen, H., Yang, J., and Narasimhan, K. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022.
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.
- Yao, W., Heinecke, S., Niebles, J. C., Liu, Z., Feng, Y., Xue, L., Rithesh, R., Chen, Z., Zhang, J., Arpit, D., et al. Retroformer: Retrospective large language agents with policy gradient optimization. In *The Twelfth International Conference on Learning Representations*.
- Zhang, B., Mao, H., Ruan, J., Wen, Y., Li, Y., Zhang, S., Xu, Z., Li, D., Li, Z., Zhao, R., et al. Controlling large language model-based agents for large-scale decision-making: An actor-critic approach. *arXiv preprint arXiv:2311.13884*, 2023.
- Zhang, D., Chen, L., Zhang, S., Xu, H., Zhao, Z., and Yu, K. Large language models are semi-parametric reinforcement learning agents. *Advances in Neural Information Processing Systems*, 36, 2024.
- Zhou, A., Yan, K., Shlapentokh-Rothman, M., Wang, H., and Wang, Y.-X. Language agent tree search unifies reasoning, acting, and planning in language models. In *Forty-first International Conference on Machine Learning*, 2024.

A. 额外结果

A.1. BabyAI-Text 中的其他任务结果

为了进行全面的比较，我们展示了 LAC 和在 Figure 6 中来自 BabyAI-Text 的其他任务的基线的表现。我们的 LAC 超过了所有其他基线，进一步验证了 LAC 的有效性。

A.2. 使用其他定义 Q_{LLM} 的结果

在 LAC 中，我们将判别器 Q_{LLM} 定义为 $Q_{LLM}(g, h_t, a_t^i) = \log \frac{P(y_w|g, h_t, a_t^i)}{P(y_l|g, h_t, a_t^i)}$ 。此外，还有其他定义，例如，最简单的变体是 $Q_{LLM}(g, h_t, a_t^i) = \log P(y_w|g, h_t, a_t^i)$ 。在本小节中，我们在 Figure 7 中提供了它们之间的性能比较。LAC 在大多数任务和模型中均优于该变体。我们推测这是因为 LAC 使用了比变体更多的信息，包括 *i.e.*、 $P(y_w|g, h_t, a_t^i)$ 和 $P(y_l|g, h_t, a_t^i)$ ，因此评估可能更准确，更稳定。可能会有其他的 Q_{LLM} 定义，其中我们的 Q_{LLM} 既简单又有效。

A.3. 直接提示 LLMs 输出动作评估的结果

为了进一步展示我们的批评者 Q_{LLM} 的优势。我们进行了额外的实验，将 LAC 和 LAC 的表现与在 WebShop 基准上的直接评估进行比较。LAC 通过提取 LLM 的内部信念（成功和失败）来生成基于数值的评估，如 Equation (2) 所述。对于 LAC 的直接评估，我们提示 LLM 直接输出成功的概率 $P(y_w)$ ，同时保持所有其他组件不变。然后 Q 值被计算为 $\log \frac{P(y_w)}{1-P(y_w)}$ 。结果如 Table 3 和 Table 4 所示，表明我们的 LAC 方法在大多数基础模型中比直接评估 LAC 在成功率和奖励方面表现更佳。对结果轨迹的分析揭示了 LAC 的直接评估往往产生无信息的成功概率（*e.g.*， $P(y_w) = 0.5$ ），导致策略改进无效。

Table 3. LAC 和 LAC 在 WebShop 中直接评估的成功率

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC	32 %	46 %	39 %	32 %
LAC w/ direct evaluation	27 %	29 %	33 %	24 %

Table 4. LAC 和 LAC 的奖励在 WebShop 中直接评估

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC	0.5840	0.7237	0.6733	0.6299
LAC w/ direct evaluation	0.5636	0.6975	0.6453	0.6333

A.4. 我们的反射组件和思维链的比较

在本小节中，我们通过实验比较了 LAC 中使用的反射法和 (Wei et al., 2022; Kojima et al., 2022) 中的思维链（CoT）方法。

尽管反思受到 CoT 的启发，但确定反思内容不易。我们的方法根据先前的动作及其结果产生判断，而 CoT 则使用任意的思考生成，而没有对过去错误的结构化关注。反思特别为决策任务设计，并且相比于简单的 CoT 具有两个优势：

强度 1。 我们的反思可以通过避免在策略生成时出现的先前错误来直接改善策略。

为了证实这一主张，我们进行了实验，比较了在 WebShop 基准上，带反思的先前策略与带 CoT 的先前策略的性能。以下是这两种变体的详细信息：

(1) 先前策略带反思：我们移除 LAC 的所有组件，保留策略和反思不变。具体地说，每一步中，在观察到动作结果后，反思首先对之前的动作产生一些判断，然后策略根据这些判断选择下一个动作。(2) 先前策略带 CoT：我们移除 LAC 的所有组件，除了策略之外。此外，我们通过添加“让我们一步步思考”来为策略提供 CoT。具体来说，在每一步中，在选择下一个动作之前，CoT 提示组件首先输出一些可能有助于解决任务的任意想法。

如 Table 5 和 Table 6 所示，在大多数基础模型中，带有反思的先前策略在成功率和奖励方面始终超过带有链式推

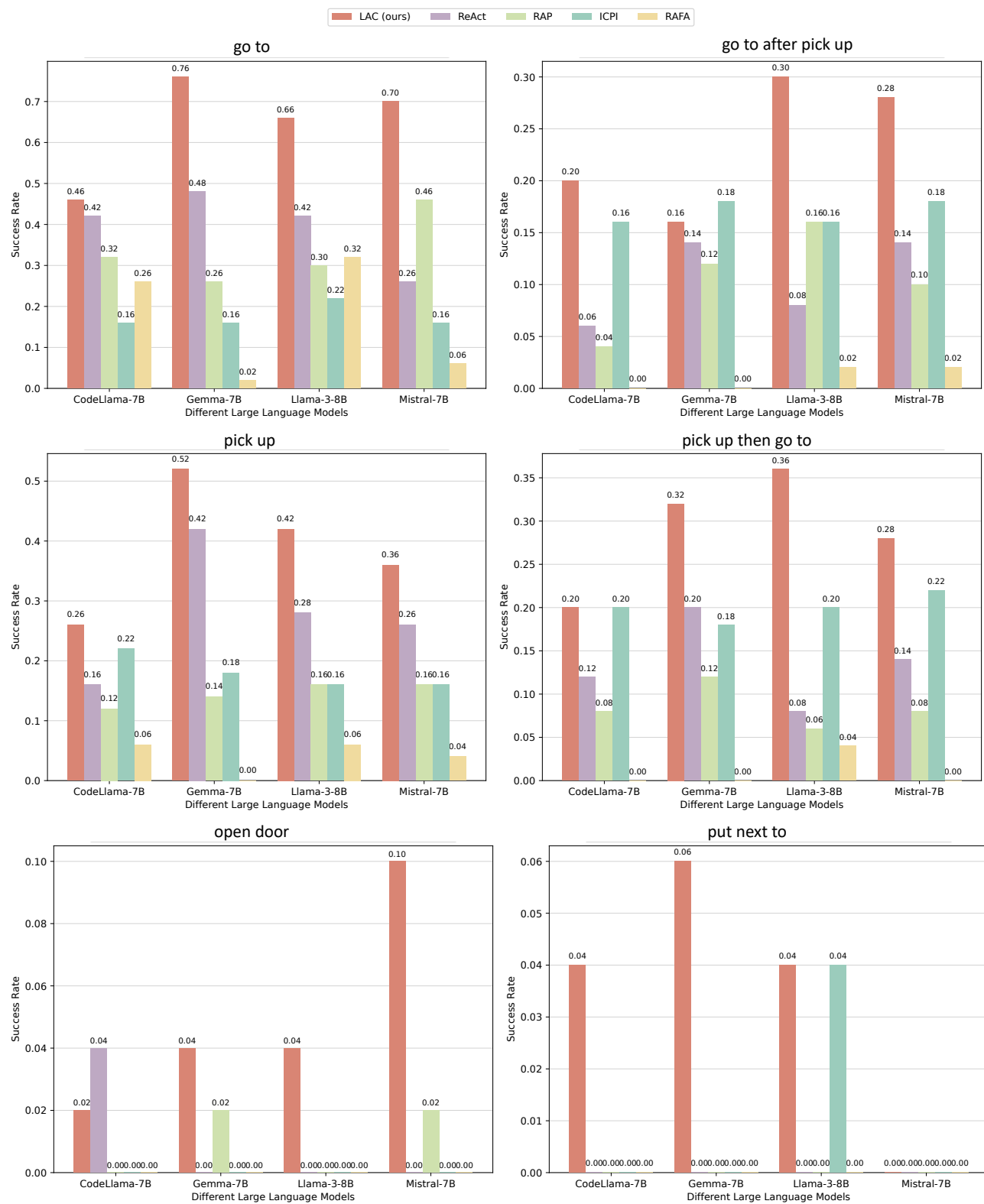


Figure 6. 我们 LAC 在所有任务中与各种基准在 BabyAI-Text 中的表现对比。

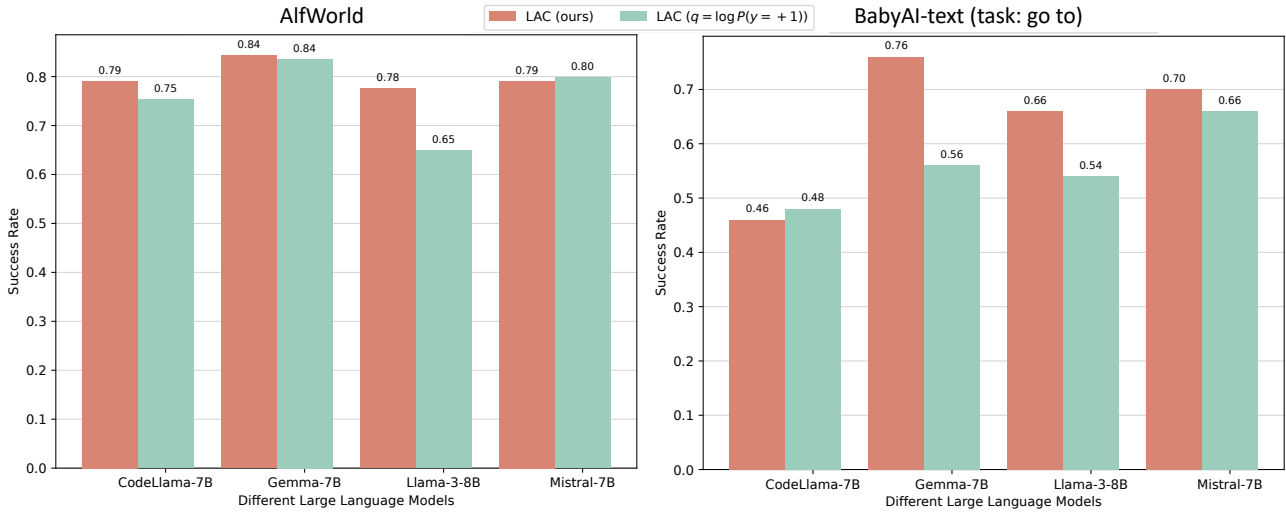


Figure 7. 使用不同定义的评论者 Q_{LLM} 时 LAC 的性能

理的先前策略。通过分析结果，我们发现带有链式推理的先前策略可能会多次犯相同的错误并陷入其中，而带有反思的先前策略则可以在很大程度上避免出现的错误。

Table 5. 在 WebShop 中先验策略结合反射和先验策略结合 CoT 的成功率

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
prior policy w/ reflection	21 %	46 %	39 %	31 %
prior policy w/ CoT	21 %	20 %	31 %	20 %

Table 6. 在 WebShop 中具有反思的先前策略和具有链式思维的先前策略的奖励

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
prior policy w/ reflection	0.5739	0.6564	0.6556	0.6288
prior policy w/ CoT	0.5520	0.5347	0.6379	0.4671

我们的反思可以无缝地与评论 Q_{LLM} 整合，这有助于评论生成更准确的基于价值的评估。

为了进一步评估这种集成，我们在 WebShop 基准上比较了 LAC 和 LAC 与 CoT 的性能。两种方法的详细信息如下：

(1) LAC：我们原始的方法。(2) LAC 与 CoT：我们用 CoT 替换 LAC 的反射成分，其他成分保持不变。

我们在 Table 7 和 Table 8 中显示了结果。我们的 LAC 方法在所有评估的基础模型中，一贯在成功率和奖励方面优于带有 CoT 的 LAC。这是因为，没有对前一步骤的反思判断，评价者可能会输出不准确的基于价值的估计，从而阻碍策略改进进行阶段。

总之，反思对过去行动的集中评估提供了相较于简单反思（CoT）的关键优势，从而在决策任务中提升了性能。

A.5. 将 LAC 与更多基线在 ALFWorld 上进行比较

在本小节中，我们将 LAC 与包括一些使用 LLMs 在 ALFWorld (Shridhar et al., 2021) 上实现的传统 RL 方法在内的更多基线进行比较。比较结果如 Figure 8 所示。

Table 7. WebShop 中使用 CoT 的 LAC 和 LAC 的成功率

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC	32 %	46 %	39 %	32 %
LAC w/ CoT	27 %	29 %	33 %	24 %

Table 8. WebShop 中具有 CoT 的 LAC 和 LAC 的奖励

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC	0.5840	0.7237	0.6733	0.6299
LAC w/ CoT	0.5734	0.5270	0.6515	0.5101

A.5.1. 微调策略

虽然在 LAC 中，我们微调模型以使用少量轨迹生成反射，但也可以微调策略以使用这些轨迹生成动作。因此，一个潜在的基线可能是在 ReAct 中微调策略。为了展示微调策略带来的改进，我们在 ReAct (Yao et al., 2023) 中微调策略，并在 Figure 8 中展示结果。我们也展示了在 Figure 8 中微调策略后的 LAC 的结果。简而言之，微调策略后的 ReAct 作为基线相比其他基线是一种强大方式，但仍然不如我们的方法 LAC 和微调策略后的 LAC。与 LAC 相比，LAC 微调策略后的表现不佳源于其倾向于过拟合训练轨迹。这种过拟合导致策略偏好于数据集中更频繁出现的动作，可能导致非最优的动作选择。

例如，在 ALFWorld 训练数据集中，动作“从 X 拿一个苹果”经常出现。在微调之后，策略可能会不成比例地生成这个动作，即使它与当前目标无关。一种情况是当前目标是“加热一些鸡蛋并把它放进垃圾桶里”。当代理在“冰箱 1”中看到“苹果 2”时，它生成并选择了一个无关的动作“从冰箱 1 拿出苹果 2”，这与任务不一致。

这种过拟合的趋势产生于策略函数的复杂性，它将状态 s 映射到动作 a ，通常超过评估者的复杂性。策略往往必须为每一种状态捕捉多种潜在的动作，特别是在复杂的环境中。然而，在我们的设置中，相当有限的训练数据集限制了其有效泛化的能力，导致了对特定动作的记忆而非灵活的决策。相比之下，用于反射生成和模拟的模型则侧重于捕捉环境更可预测的动态和更简单的评价标准。这通常需要比策略所需的更简单的映射，从而避免过拟合。

A.5.2. 基于 LLM 的 RL 变体

我们还包括一些基于 LLM 的 RL 变体作为基线，以展示 LAC 相对于传统 RL 算法的优势。我们设计了三种基于 LLM 的 RL 变体，这些变体构建在预训练的 LLM 之上，并直接从 LLM 中提取动作/价值信息，而无需添加动作/价值头，即 Figure 8 中的传统策略梯度、带密集奖励的传统策略梯度和传统演员-评论者。

为了实现传统策略梯度，我们需要得到动作的概率和回报。为了获得动作的概率，我们直接使用 LLM 来计算在目标 g 和历史 h_t 给定的情况下，动作 $a_i = [w_1, w_2, \dots, w_{|a_i|}]$ 中每个标记的条件概率，然后计算它们的乘积：

$$\pi(a_t|g, h_t) = \prod_{j=1}^{|a_t|} P_{LLM}(w_j|g, h_t, w_{<j})$$

，其中 $P_{LLM}(w_j|g, h_t, w_{<j})$ 是在目标 g 、历史 h_t 和先前标记 $w_{<j}$ 给定的情况下由 LLM 计算出的标记 w_j 的概率。然后我们将累积的未来奖励视为回报 G_t ，它在我们考虑的任务中成功轨迹为 +1，失败轨迹为 -1。最后，策略的梯度是 $\mathbb{E}[\sum_t \nabla \log \pi(a_t|g, h_t) G_t]$ 。带有密集奖励的传统策略梯度与传统策略梯度相似，只是我们手动为每一步添加了中间奖励，然后使用累积的未来奖励作为回报 G_t 。

为了实现传统的 Actor-Critic 方法，我们还需要一个评论家来估计动作值。由于可以仅使用 18 条轨迹训练一个新的价值头部，我们转而在方法 LAC, i.e.

$$Q_{LLM}(g, h_t, a_t, u_t) = \log \frac{P_{LLM}(y_w|g, h_t, a_t, u_t)}{P_{LLM}(y_l|g, h_t, a_t, u_t)}$$

中采用类似 Q_{LLM} 的方法来近似动作值，其中 $P_{LLM}(y_{\{w,l\}}|g, h_t, a_t, u_t)$ 是特殊的正/负标记（如 GOOD 或 BAD）的输出概率，这些标记表明 LLM 对成功/失败结果的正/负信念。最后，策略的梯度为 $\mathbb{E}[\sum_t \nabla \log \pi(a_t|g, h_t) Q_{LLM}(g, h_t, a_t, u_t)]$ 。

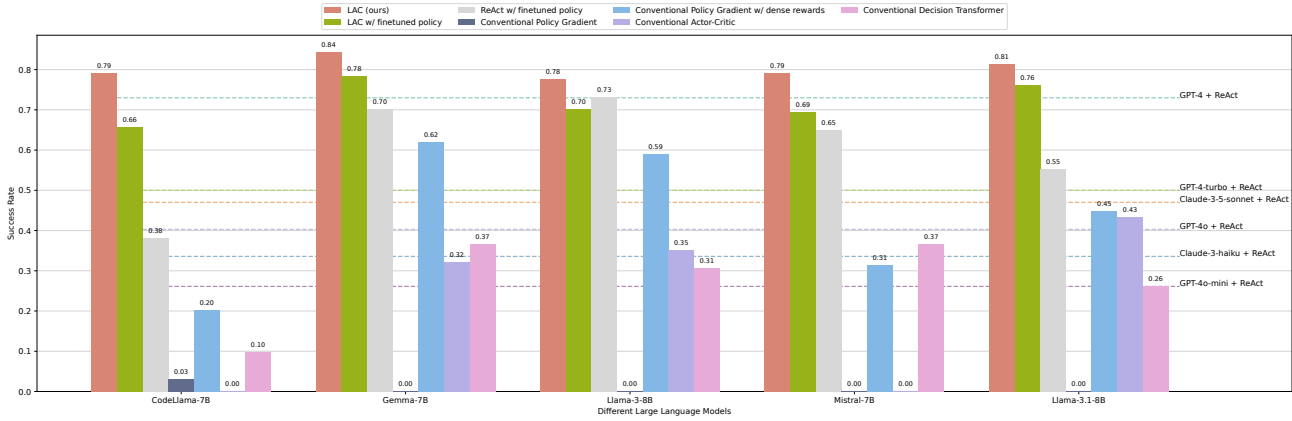


Figure 8. 我们 LAC 和 LAC 的变体在基准 ALFWorld 中与各种基线的性能比较。

总之，与具有稠密奖励的传统策略梯度相比，传统策略梯度表现几乎为零，这是由于极其稀疏的奖励问题。传统演员-评论家方法仅在一些较强的 LLM 上，例如 Gemma-7B (Team et al., 2024)、Llama-3-8B (Meta, 2024a) 和 Llama-3.1-8B (Meta, 2024b)，表现出非零性能，这可能是因为传统演员-评论家方法的优化方法不适合在数据不足的情况下使用。

A.5.3. 决策 TRANSFORMER

除了前面提到的基于 LLM 的 RL 变体外，Decision Transformer (Chen et al., 2021) 也是结合 RL 和基于 Transformer 的 LLM 的潜在解决方案。我们以类似于常规决策变压器的方式微调预训练的 LLM。我们使用决策变压器的轨迹表示构建数据集： $\tau = [R_1, s_1, a_1, R_2, s_2, a_2, \dots]$ ，其中 R_t 是 return-to-go，在我们极其稀疏的奖励设置中，对成功的轨迹 +1，对于失败的轨迹 -1。然后，我们使用下一个预测损失在这些轨迹上微调 LLM。在执行期间，我们在状态 s_t 之前插入 +1 以指定期望的结果。结果如 Figure 8 中所示，作为常规决策变压器。简而言之，常规决策变压器表现出与 ReAct 相似的性能，这可能是因为 18 条轨迹不足以微调决策变压器。

我们的 LAC 方法优于所有已考虑的基线，因为它能够利用 LLM 的先验知识处理极度稀疏的奖励问题，并充分利用不足的数据。

A.6. LAC 计算成本分析，加入更多基线到 ALFWorld

在这个小节中，我们展示了 LAC 和在 Figure 9 中的其他基线的计算成本。我们分别展示成功率、每个任务的步骤、每个任务所用的时间以及每个任务的 token 成本。具体而言，尽管 LAC 由于评论者和前向模型的额外推理过程导致了每步更高的计算成本，但总成本仍然低于大多数基于 LLM 的基线，这是因为 LAC 具有更高的成功率并且完成每个任务所需的步骤更少。

A.7. ALFWorld 和 BabyAI-Text 的说明

我们分别在 Figure 10 和 Figure 11 中展示 ALFWorld 和 BabyAI-Text 的示例。

Figure 10 展示了一个具体情境，其中代理的目标是“将盐罐放入抽屉中”。在一个关键决策步骤（步骤 4），我们观察到组件之间的以下直观区别：（1）仅基于 LLM 的先前策略错误地建议“去抽屉 1”，因为基础的 LLM 模型忽略了代理已经在橱柜 2 中找到了正确的物品（“盐罐 1”）。这一错误例证了 LLM 中常见的幻觉问题，即模型忽视了之前的状态并错误地推荐了不相关的行动。（2）相比之下，评论者建议“从橱柜 2 中取出盐罐 1”，因为它通过预测未来的轨迹来评估潜在的行动，并确定这一行动将成功获得正确的物品。（3）我们的方法通过根据评论者的评估优化先前策略的行动分配，利用这些不同的见解。它有效地纠正了先前策略引入的错误，在先前策略的灵活性（灵活但有时不准确）和评论者的评估（准确但计算量大）之间取得了平衡。

该分析表明：（1）当两个模型都一致时，改进策略的置信度最高；（2）当两者不一致时，改进策略倾向于信任置信度更高的模型；（3）在低置信度情况下，策略保持保守。

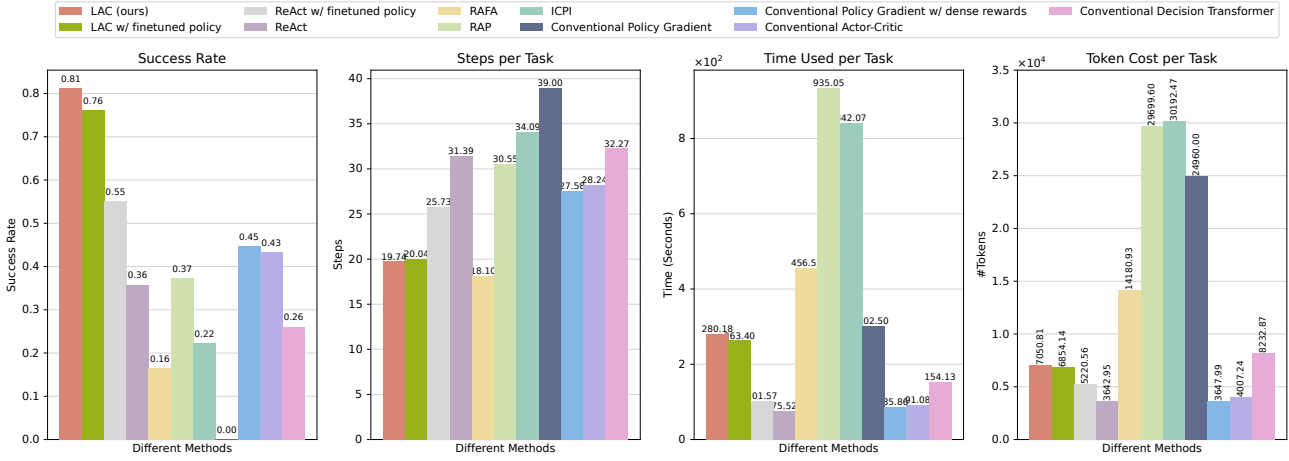


Figure 9. 我们的 LAC 与基准测试中的各种基线的计算成本分析。虽然由于评论者和前向模型的额外推理过程，LAC 每一步的计算成本可能更高，但 LAC 的总成本仍低于大多数基于 LLM 的基线，因为 LAC 完成每项任务所需的步骤较少。

这些结果表明，我们的方法通过基于模型置信度的加权，隐式地聚合了来自两种来源的决策知识，从而在没有显式梯度的情况下实现了有效的策略改进。

A.8. 不同评价改进方法的结果

根据经验，我们发现可以通过在少量轨迹上进行微调来改进评论者。有关更详细的微调信息，请参阅 Appendix A.12。为了展示微调的有效性，我们展示了在将这些示例仅添加到提示 *i.e.* 中，即上下文学习时，LAC 和其他变体在“go to”和“pick up”任务上的表现，分别来自 Table 9 和 Table 10 的 BabyAI-Text。我们还在括号中展示了如果进行微调的情况下性能的提升。结果表明，在我们的案例中，微调比上下文学习更能将额外知识融入到大语言模型中，并提高动作评估的质量，从而实现更好的性能。值得一提的是，我们的 LAC 即使没有微调也仍优于基线。

Table 9. 两种评论改进方法的性能：上下文学习或微调。

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC	0.30 (↑0.16)	0.62 (↑0.14)	0.32 (↑0.34)	0.24 (↑0.46)
LAC w/o reflection	0.30 (↑0.02)	0.58 (↑0.02)	0.38 (↑0.10)	0.26 (↑0.28)
LAC w/o critic	0.28 (↑0.06)	0.48 (↑0.04)	0.42 (↑0.10)	0.10 (↑0.28)
critic-only	0.42 (↑0.04)	0.40 (↑0.24)	0.34 (↑0.22)	0.38 (↑0.24)

Table 10. 两种评论者改进方法的性能：上下文学习或微调。

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC	0.20 (↑0.06)	0.22 (↑0.20)	0.34 (↑0.08)	0.20 (↑0.16)
LAC w/o reflection	0.16 (↑0.08)	0.32 (↑0.04)	0.32 (↑0.04)	0.22 (↑0.06)
LAC w/o critic	0.12 (↑0.14)	0.36 (↑0.20)	0.28 (↑0.06)	0.26 (↑0.04)
critic-only	0.22 (↑0.04)	0.24 (↑0.26)	0.16 (↑0.16)	0.16 (↑0.26)

A.9. 对 LAC 微调过程的分析

为了提高由 LLM 生成的反射的质量，我们对生成反射的 LLM 进行微调。在本节中，我们分析微调过程，展示微调的影响，以及不同数据量和正负样本比例对任务成功率的影响。比较可以见于 Figure 12。

在 Figure 12 (a) 中，我们展示了微调数据规模的影响。我们使用 9、18、27 和 36 条轨迹来微调 LLM，并展示在

Prompt

[Instruction]: Interact with a household to solve a task. Here are two examples. (...)

[Task]: You are in the middle of a room. Looking quickly around you, you see a cabinet 6, a cabinet 5, a cabinet 4, a cabinet 3, a cabinet 2, a cabinet 1, a coffeemachine 1, a countertop 3, a countertop 2, a countertop 1, a drawer 3, a drawer 2, a drawer 1, a fridge 1, a garbagecan 1, a microwave 1, a shelf 3, a shelf 2, a shelf 1, a sinkbasin 1, a stoveburner 4, a stoveburner 3, a stoveburner 2, a stoveburner 1, and a toaster 1.

Your task is to: put a saltshaker in drawer.

Inputs & Outputs of LAC (Ours)

(...)

[Act 1]: go to cabinet 1

[Obs 2]: On the cabinet 1, you see a glassbottle 1.

[LAC Inference 2]

[Act 2]: go to cabinet 2 ✓

[Obs 3]: The cabinet 2 is closed.

[LAC Inference 3]

[Act 3]: open cabinet 2 ✓

[Obs 4]: You open the cabinet 2. The cabinet 2 is open. In it, you see a plate 1, a saltshaker 1, and a soapbottle 1.

[LAC Inference 4]

[Act 4]: take saltshaker 1 from cabinet 2 ✓

[Obs 5]: You pick up the saltshaker 1 from the cabinet 2.

(...)

[Obs 7]: You open the drawer 1. The drawer 1 is open. In it, you see a fork 1, and a peppershaker 1.

[LAC Inference 7]

[Act 7]: put saltshaker 1 in/on drawer 1 ✓

[Obs 8]: You put the saltshaker 1 in/on the drawer 1.

Action Probs. (Partial)

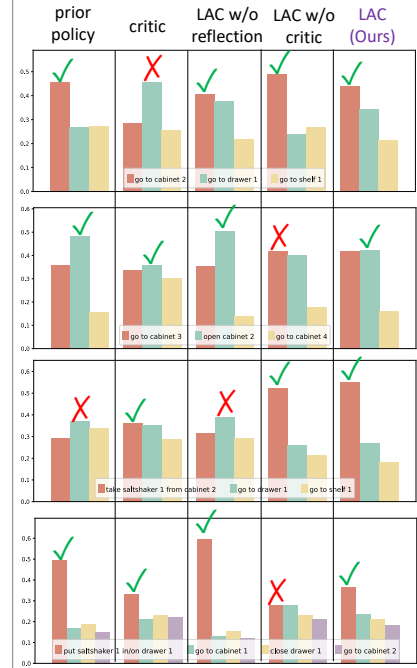


Figure 10. 我们的方法 LAC 在 ALFWorld 中的一个说明性解释。右侧的直方图显示了不同方法的动作概率。虽然 LLM 的先验策略 (π_{LLM}) 和评论者, 以及没有反思的 LAC 在不同的时间步骤中会犯错, 但 LAC (我们的方法) 可以通过优化给定动作评估的策略来选择正确的动作。LAC 推理步骤在 Figure 1 中详细说明。

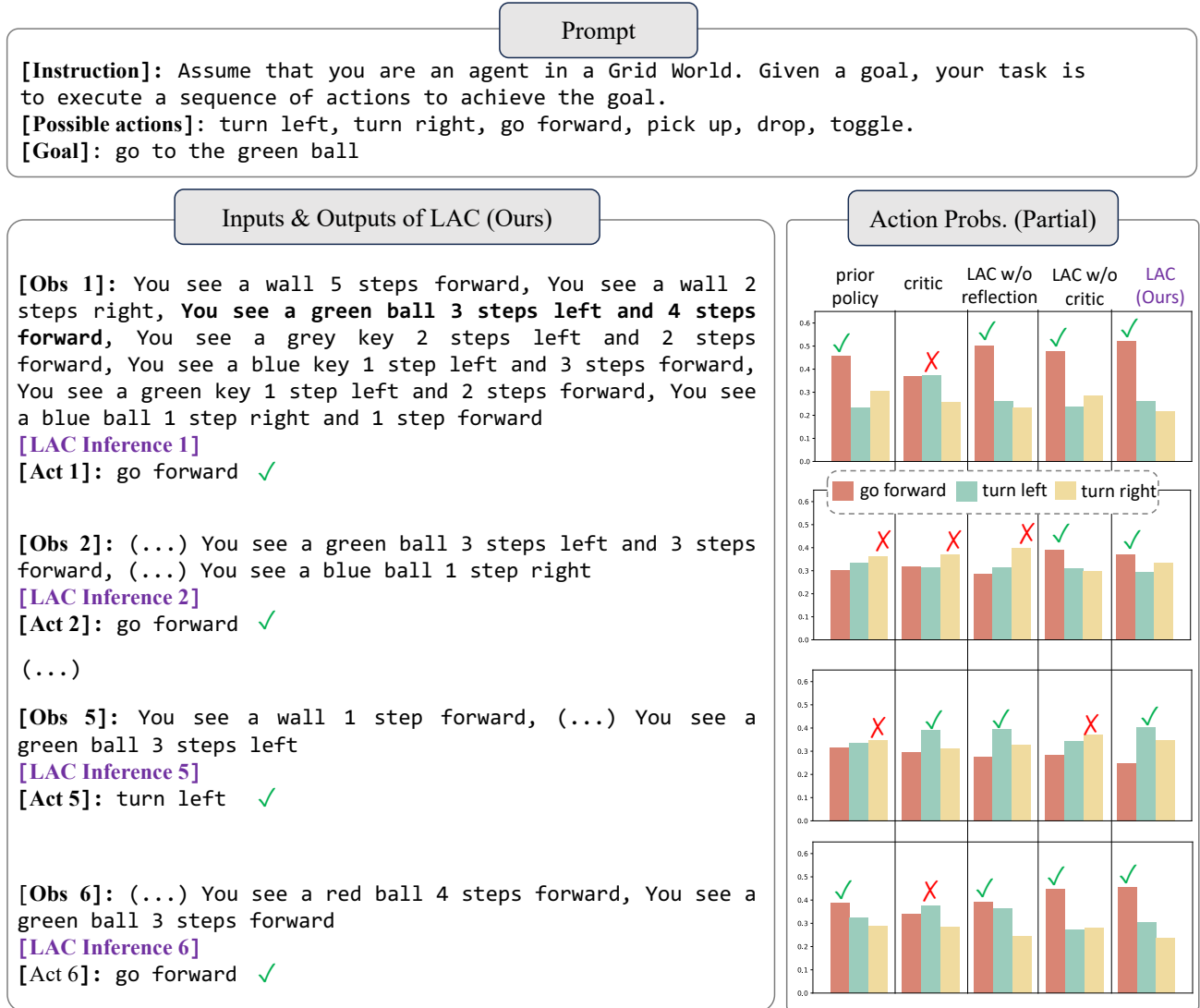


Figure 11. 我们的方法 LAC 在 BabyAI-Text 中的一个示例说明。右侧的直方图展示了不同方法的动作概率。当 LLM 的先验策略 (π_{LLM}) 和评论者, 以及没有反思的 LAC 在不同时间步分别犯错时, LAC (我们的) 通过基于动作评估优化策略来选择正确的动作。请参阅 Table 30 以获取完整的轨迹。

134 个评估任务中的最终成功率。总的来说，较大的数据规模（27 或 36 条轨迹）通常带来更高的成功率，而较小的数据规模（在某些情况下甚至是 18 或 9 条轨迹）已经足够让 LAC 实现超常表现。

Figure 12 (b) 展示了不同正/负样本比率（正: 负 = 0:1, 1:3, 1:1, 3:1 和 1:0）对最终性能的影响。我们保持样本总数不变，仅仅改变正/负比例。简而言之，我们的 LAC 在合理的正/负比例（e.g. 1:3, 1:1, 3:1）下是稳健的，而基于 CodeLlama-7B (Roziere et al., 2023) 和 Gemma-7B (Team et al., 2024) 的 LAC 在给定全部正样本（1:0）时甚至表现更好。

Figure 12 (c) 显示了微调过程的学习曲线。我们绘制了 CodeLlama-7B (Roziere et al., 2023) 的下一个预测损失和正/负标记预测准确率。简而言之，随着微调过程中下一个标记预测损失的减少，预测特殊标记（GOOD 或 BAD）的准确率提高，这体现了微调过程的效果。

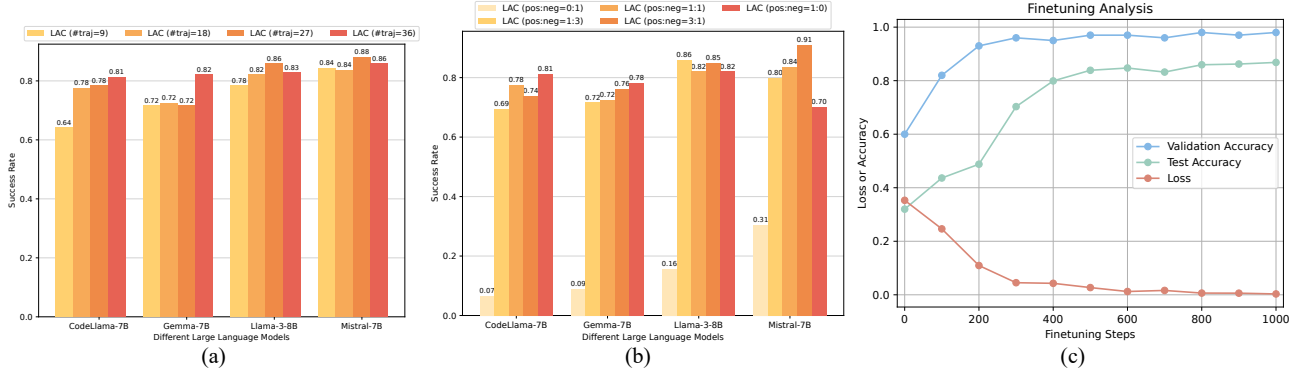


Figure 12. 关于我们 LAC 微调过程的分析。(a) 微调数据规模的影响。较大的数据规模（27 个、36 个轨迹）通常能带来更高的性能，但较小的数据规模（18 个甚至 9 个轨迹）已经足够让我们的方法取得较好的表现。(b) 正负数据比例的影响。LAC 对合理的正负比例（1:3、1:1、3:1）具有鲁棒性，而基于 CodeLlama-7B 和 Gemma-7B 的 LAC 在全部为正数据（1:0）时表现甚至更好。(c) CodeLlama-7B 和 ALFWorld 的下一个词预测损失和正负词预测准确性的学习曲线。

A.10. 关于微调影响的消融研究

微调过程在我们的方法中是重要的，但它也可以通过上下文学习来替代，其中几个示例被插入到每个输入中。在这里，我们将我们的方法与 ReAct 比较，以观察微调在 ALFWorld 中的效果。Table 11 中展示的结果表明，我们的方法在有无微调的情况下都优于基准，而且性能可以通过微调进一步提高。通过微调，我们的方法在不同的 LLM 中提供了更一致的表现。

Table 11. 关于微调影响的消融研究。

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC (w/o finetuning)	0.39	0.59	0.71	0.57
ReAct (w/o finetuning)	0.20	0.54	0.31	0.34
LAC (w/ finetuning)	0.79	0.84	0.78	0.79
ReAct (w/ finetuning)	0.38	0.70	0.73	0.65

A.11. 不同超参数 α 的结果

在 Equation (4) 中，超参数 α 控制相对于原始策略 π_{LLM} 的偏差。在本小节中，我们针对 BabyAI-Text 的任务 “go to” 在超参数 $\{1/2, 1, 2, 5, 10\}$ 上进行了网格搜索，然后我们为其他任务固定 α ：对于模型 CodeLlama-7B 使用 $\alpha = 1$ ，模型 Gemma-7B 使用 $\alpha = 2$ ，模型 Llama-3-8B 使用 $\alpha = 2$ ，模型 Mistral-7B 使用 $\alpha = 10$ 。

对于基准 ALFWorld，我们在所有实验中固定了 $\alpha = 1$ 。

为了进一步展示我们方法的强大之处，我们在 Crafter (Hafner, 2022) 上进行了初步实验，使用的是来自 BALROG (Paglieri et al., 2024) 的实现。Crafter 是一款 2D 生存游戏，专门设计用于测试长时间推理和顺序决策，任务涉及

Table 12. 不同超参数 α 的结果

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC ($\alpha = 1/2$)	0.46	0.54	0.62	0.68
LAC ($\alpha = 1$)	0.46	0.62	0.64	0.58
LAC ($\alpha = 2$)	0.44	0.76	0.66	0.64
LAC ($\alpha = 5$)	0.46	0.72	0.62	0.64
LAC ($\alpha = 10$)	0.40	0.58	0.60	0.70

资源收集、制作和战斗。它代表了一个比 ALFWorld 和 WebShop 复杂得多的环境。

我们使用 Llama-3.1-8B-It 在此基准上评估了我们的方法，遵循 BALROG 的官方评估协议。我们将 LAC 与 BALROG 的 GitHub 库中的几个代表性基准进行了比较。初步结果汇总在 Table 13 中。在相同的评估设置下，我们的方法比其他现有基准实现了更高的性能。这些初步结果进一步证实了我们提出的演员-评论家方法的稳健性、有效性和适应性，特别是在显著更具挑战性和复杂性的决策环境中。需要注意的是，CoT 在 Crafter 基准上比基准 Naive 表现更差。我们推测这是由于模型在思维链中不一致导致的。两个连续的思维链可能导致模型采取朝向不同目标的行动，这是不理想的。作者也指出，这尤其是较小、较弱的模型的一个问题。

Table 13. LAC（我们的）与其他基线在 Crafter 中的性能比较（来自 BALROG）

	Llama-3.1-8B-It
LAC (Ours)	25.91 % $\pm$ 1.93 %
Naive (direct action generation)	20.45 % $\pm$ 4.26 %
Robust Naive (formatted actions)	4.55 % $\pm$ 1.57 %
CoT (ReAct, reason then act)	18.64 % $\pm$ 3.24 %
Robust CoT (reason + formatted actions)	15.46 % $\pm$ 3.59 %
Few-Shot (in-context examples)	12.73 % $\pm$ 1.25 %

我们进行了初步实验，使用推理大型语言模型，如 DeepSeek-R1-Distill-Qwen-7B。然而，我们观察到它们通常倾向于过度思考，而不是直接输出环境动作，无论是在我们的方法中还是在基准测试中。例如，即使我们明确提示推理大型语言模型输出动作（例如，“请确保你的回应采用以下格式：n> 你选择的动作”），这些模型仍然生成详细的解释，但避免选择下一个动作。一个典型的回应可能是：“我需要找到一个钥匙来打开保险箱或在抽屉中找到铅笔。由于我无法（……），我不能（……），我必须（……）”。这一问题在之前的工作中也有提到。我们相信，对于决策任务使用推理大型语言模型需要更深入的探索，以平衡内部推理与有效的环境互动。

在这个小节中，我们将推导。我们优化以下目标：我们现在有：其中我们有分区函数：由于分区函数仅是的函数，而与优化的策略无关，我们定义这种策略的定义是一个有效的概率分布，因为对于所有和来说是成立的。由于不是的函数，我们可以重新组织目标为：

由于 $Z(g, h_t)$ 不依赖于 π ，我们只需关注 KL 散度，它仅当两个分布相同时才在 0 时达到最小值。因此，最优解是

$$\pi(a_t^i | g, h_t) = \pi^*(a_t^i | g, h_t) = \frac{1}{Z(g, h_t)} \pi_{\text{LLM}}(a_t^i | g, h_t) \exp(\alpha Q_{\text{LLM}}(g, h_t, a_t^i, u_t^i)), \quad (7)$$

，这完成了推导。

A.12. LAC 的评论改进

我们使用的反思评估器 Q_{LLM} 和前向模型 f_{LLM} 可以通过提供类似任务的说明或少量样例（如以往工作的 (Yao et al., 2023; Liu et al., 2023)）来提示大语言模型 (LLMs) 轻松实现。然而，根据经验，我们发现可以通过在从训练任务中收集的几个样本上使用简单的下一个 token 预测损失对 LLMs 进行微调，从而进一步提高性能。在这项工作中，我们考虑对 ALFWorld 和 BabyAI-Text 进行微调，以展示微调的影响。我们为每个基准测试使用 18 个轨迹进行微调，其中包括每种任务类型的两个成功轨迹和一个失败轨迹。虽然 18 个轨迹显著少于传统强化学习算法所需的数量，但对于我们的方法来说通常是足够的。每个轨迹的格式如下：(g, o₀, a₁, o₁, c₁, ..., a_H, o_H, c_H)，其中 H 是情节长度，c_i 是动作 a_t 的一个反思。每个 c_t 包括关于动作 a_t 的解释 (e.g., “我找到了物体-X。这

一步是”等或者“我应该首先取物体-X 而不是物体-Y。这一步是”等) 以及一个表示正/负判断的特殊标记 (*e.g.*, “好” 或 “坏”)。

实际上, 我们只需微调一次 LLM, 并利用它来生成所有的反思, 构建评论者 Q_{LLM} , 以及前向模型 f_{LLM} , 这得益于上述数据格式的微调。具体而言, 在最小化预测未来轨迹的损失时, 前向模型 f_{LLM} 得到了改善。在最小化生成反思的损失时, 反思和评论者 Q_{LLM} 都得到了改善。后者是因为反思也包含指示正面/负面判断的特殊标记, 其生成概率用于在 Equation (3) 中计算 Q_{LLM} 。我们在 Appendix A.9 中分析了这个微调过程。在 ALFWorld 和 BabyAI-Text 中标记轨迹的一些例子分别展示在 Table 27 和 Table 28 中。

B. 实验细节

B.1. 基准详情

B.1.1. ALFWORLD: 具有高级操作的基准

我们选择 ALFWorld (Shridhar et al., 2021), 一个基于文本的家庭环境, 以展示 LAC 在高层规划中的有效性。ALFWorld 是一个与 ALFRED (Shridhar et al., 2020) 基准对齐的合成基于文本的游戏。在这个环境中六种类型的任务, 要求智能体通过一系列高层动作实现高层目标, 例如 *e.g.* “前往地点-X”、“从地点-X 取回对象-Y”。关于 ALFWorld 中六种任务类型的详细信息见 Table 14。

Table 14. 所有任务类型及其对应的目标对于 ALFWorld

Type	Description
Pick & Place	The agent needs to put a target object to a target place, <i>e.g.</i> put some spraybottle on toilets, find some apple and put it in sidetable, <i>etc.</i>
Clean & Place	The agent needs to find a target object, clean it and put it to a target place, <i>e.g.</i> clean some apple and put it in sidetable, put a clean lettuce in diningtable, <i>etc.</i>
Heat & Place	The agent needs to find a target object, heat it and put it to a target place, <i>e.g.</i> heat some egg and put it in diningtable, put a hot apple in fridge, <i>etc.</i>
Cool & Place	The agent needs to find a target object, cool it and put it to a target place, <i>e.g.</i> cool some pan and put it in stoveburner, put a cool mug in shelf, <i>etc.</i>
Examine & Place	The agent needs to find a target object, and examine it with desk lamp, <i>e.g.</i> look at bowl under the desk lamp, examine the pen with the desk lamp, <i>etc.</i>
Pick Two & Place	The agent needs to put two target objects to a target place, <i>e.g.</i> put two saltshaker in drawer, find two pen and put them in dresser, <i>etc.</i>

ALFWorld 中的一个挑战是代理需要探索环境以找到目标物品。大型语言模型中关于常见家居物品可能位置的常识知识使得这个环境适合大型语言模型来解决。只有当代理达到目标时奖励为 1。按照 ReAct, 我们在任务特定设置中评估了 134 个未见过的评估游戏。

B.1.2. BABYAI-TEXT: 带有低级别动作的基准测试

对于进行低级规划的决策任务, 我们采用 BabyAI-Text (Carta et al., 2023b) 作为我们的测试平台。BabyAI-Text 是从 BabyAI 平台 (Chevalier-Boisvert et al., 2018) 扩展的纯文本版本环境。BabyAI-Text 是一个网格世界环境, 其中代理和对象被放置在一个 8×8 瓦片的房间里。代理具有 6 个原始动作: 左转、右转、向前走、拾起、放下、切换, 以解决用自然语言描述的任务 (举例: *e.g.* 拾起红色箱子)。代理可以访问 7×7 部分视图, 这意味着它只能观察到位于它面前 7×7 网格中的对象。除了与完成给定任务相关的对象外, 房间里还有其他干扰物。BabyAI-Text 中的所有任务类型都在 Table 15 中展示。

Table 15. BabyAI-Text 的所有任务类型及对应目标

Type	Description
go to	The agent needs to find target object and go to it, <i>e.g.</i> go to the green key, go to the red ball, <i>etc.</i>
pick up	The agent needs to find target object, go to it and pick up it, <i>e.g.</i> pick up the blue key, pick up the purple ball, <i>etc.</i>
go to after pick up	The agent needs to find and pick up one object, then go to another object, <i>e.g.</i> go to the blue key after you pick up the green key <i>etc.</i>
pick up then go to	The agent needs to find and pick up one object, then go to another object, <i>e.g.</i> pick up the green box, then go to the purple box <i>etc.</i>
put next to	The agent needs to find and pick up one object, then go to another object and put the first object next to it, <i>e.g.</i> put the grey key next to the yellow ball <i>etc.</i>
open door	The agent needs to know which key to pick up, then find and pick up it to open the door, <i>e.g.</i> open the door, open the blue door, <i>etc.</i>

与 ALFWorld 不同, 代理在与 BabyAI-Text 交互时需要找出在每一步执行合适的低级动作。我们在 BabyAI-Text 中的测试环境中进行评估。房间内的物体是随机选择的, 它们的位置以及代理的位置也是随机的。考虑到时间和

计算资源的限制，我们对每种任务类型评估 50 个任务，总计 300 个任务。

B.1.3. WEBSHOP：具有潜在无限动作空间的基准

在某些场景中，可能的动作空间可以是无限的。我们采用了一个简化环境，WebShop (Yao et al., 2022)，具有潜在的无限动作空间，以展示我们方法的有效性。WebShop 要求代理在与模拟的网络环境互动时，根据文字指令购买所需的产品。可用的动作包括使用文本描述进行搜索以及选择按钮进行点击，比如 “[项目 ID]”、“返回搜索”、“立即购买”，*etc.* 由于自然语言的特性，人们实际上可以搜索任何东西，从而导致潜在动作集无限。我们在 WebShop 中对我们的方法和所有基准进行了 100 个任务的评估。

需要提及的一点是，这个基准测试的最终奖励是一个介于 0 和 1 之间的值，这取决于最终购买的产品在多大程度上符合要求。因此，在这项工作中，我们同时报告成功率和最终奖励，其中成功率表示代理最终购买的完全符合要求的产品所占的比例，最终奖励表示对所有评估任务的平均最终奖励。

B.2. 基线的实现细节

B.2.1. REACT

ReAct (Yao et al., 2023) 直接使用 LLM 作为策略来生成动作。他们在与环境交互的过程中结合推理和行动，以改善动作选择。

Table 16 展示了在 ALFWorld 中动作和观察之间交错的轨迹示例。为了利用 LLM 的推理能力，ReAct 在 ALFWorld 中的原始动作集上添加了一个额外的思考动作，使得 LLM 代理可以在任务执行期间进行任务分解和目标实现的推理。ReAct 给出的 ALFWorld 的注释轨迹展示在 Table 17 中。标记为浅绿色的文本是注释的思考，以帮助 LLM 更好地完成任务。

Table 16. ALFWorld 成功轨迹的一个例子

```
You are in the middle of a room. Looking quickly around you, you see a cabinet 4, a cabinet 3, a
cabinet 2, a cabinet 1, a countertop 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a
sinkbasin 2, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, and a towelholder 1.
Your task is to: put some spraybottle on toilet.
> go to cabinet 1
On the cabinet 1, you see a cloth 1, a soapbar 1, a soapbottle 1.
> go to cabinet 2
The cabinet 2 is closed.
> open cabinet 2
You open the cabinet 2. The cabinet 2 is open. In it, you see a candle 1, and a spraybottle 2.
> take spraybottle 2 from cabinet 2
You pick up the spraybottle 2 from the cabinet 2.
> go to toilet 1
On the toilet 1, you see a soapbottle 2.
> put spraybottle 2 in/on toilet 1
You put the spraybottle 2 in/on the toilet 1.
```

在与环境交互的过程中，像 Table 17 中这样的两个成功轨迹被提供给 LLMs，以利用 LLMs 的上下文学习能力。LLMs 会生成一个将在环境中执行的动作，或对如何实现最终目标进行一些推理。所选择的动作和新的观察将被附加到 LLMs 的提示中，以形成一个顺序决策过程。

类似于 ALFWorld，我们在 BabyAI-Text 中标注了一些成功的轨迹，以教导 ReAct 代理在 BabyAI-Text 中如何完成一项任务。没有思考和有思考的成功轨迹示例分别在 Table 18 和 Table 19 中展示。

对于 WebShop，我们采用了 ReAct 的实现，并使用了一个用于上下文学习的例子。没有思考和有思考的成功例子分别展示在 Table 20 和 Table 21 中。

B.2.2. RAP

RAP(Hao et al., 2023) 利用 LLMs 作为策略和世界模型，并采用树搜索规划方法来评估每个可能的动作候选。像在 Table 16、Table 18 和 Table 20 中的行动和观察交替出现的轨迹被提供给 LLMs，以学习如何作为策略选择动作以及作为世界模型的环境动态。

每一步的评估是通过一个奖励函数来完成的，该函数可以是动作的对数概率或者 LLM（大型语言模型）给出的自我评价，或者基于某些任务特定的启发式方法。在我们的实现中，我们采用 LLM 给出的动作的对数概率作为奖励。为了简化实现，我们采用了一种贪心的方法来扩展树，每次仅生成一个动作。更具体地说，在每一步

Table 17. 在 ReAct 中 ALFWorld 的成功轨迹示例

You are in the middle of a room. Looking quickly around you, you see a cabinet 4, a cabinet 3, a cabinet 2, a cabinet 1, a countertop 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a sinkbasin 2, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, and a towelholder 1.

Your task is to: put some spraybottle on toilet.

思考：要完成这个任务，我需要找到一个喷雾瓶，然后把它放在马桶上。

OK.

思考：首先，我需要找到一个喷雾瓶。喷雾瓶更有可能出现在橱柜（1-4）、台面（1）、厕所（1）、洗脸盆（1-2）、垃圾桶（1）中。我可以一个个检查，从橱柜 1 开始。

OK.

> go to cabinet 1

On the cabinet 1, you see a cloth 1, a soapbar 1, a soapbottle 1.

> go to cabinet 2

The cabinet 2 is closed.

> open cabinet 2

You open the cabinet 2. The cabinet 2 is open. In it, you see a candle 1, and a spraybottle 2.

想：现在我找到一个喷雾瓶（2）。接下来，我需要拿起它。

> take spraybottle 2 from cabinet 2

You pick up the spraybottle 2 from the cabinet 2.

想：现在我要拿起喷雾瓶（2）。接下来，我需要把它放到厕所 1 里/上。

OK.

> go to toilet 1

On the toilet 1, you see a soapbottle 2.

> put spraybottle 2 in/on toilet 1

You put the spraybottle 2 in/on the toilet 1.

Table 18. BabyAI-Text 的成功轨迹示例

Goal of the agent: go to the green key

Observation: You see a wall 2 steps right, You see a wall 3 steps forward, You see a grey box 3 steps left, You see a grey key 3 steps left and 1 step forward, You see a blue box 3 steps left and 2 steps forward

Action: turn left

Observation: You see a wall 3 steps right, You see a blue key 3 steps left and 2 steps forward, You see a green key 2 steps left and 1 step forward, You see a green ball 1 step left and 3 steps forward, You see a grey box 3 steps forward, You see a grey key 1 step right and 3 steps forward, You see a blue box 2 steps right and 3 steps forward

Action: go forward

Observation: You see a wall 3 steps right, You see a blue key 3 steps left and 1 step forward, You see a green key 2 steps left, You see a green ball 1 step left and 2 steps forward, You see a grey box 2 steps forward, You see a grey key 1 step right and 2 steps forward, You see a blue box 2 steps right and 2 steps forward

Action: turn left

Observation: You see a wall 3 steps left, You see a green box 2 steps left and 3 steps forward, You see a green box 2 steps left and 1 step forward, You see a green key 2 steps forward, You see a blue key 1 step right and 3 steps forward, You see a green ball 2 steps right and 1 step forward, You see a grey box 2 steps right

Action: go forward

Observation: You see a green box 2 steps left and 2 steps forward, You see a green box 2 steps left, You see a green key 1 step forward, You see a blue key 1 step right and 2 steps forward, You see a green ball 2 steps right

Table 19. 在 ReAct 中标注成功轨迹的 BabyAI-Text 示例

```

Goal of the agent: go to the green key
Observation:You see a wall 2 steps right, You see a wall 3 steps forward, You see a grey box 3 steps
left, You see a grey key 3 steps left and 1 step forward, You see a blue box 3 steps left and 2 steps
forward
行动: 思考: 为了完成任务, 我需要找到一把绿色钥匙并前往那里。
Observation:OK
执行: 思考: 我还没有看见一个绿色的钥匙。首先, 我需要找到一个绿色的钥匙。
Observation:OK
Action:turn left
Observation:You see a wall 3 steps right, You see a blue key 3 steps left and 2 steps forward, You see
a green key 2 steps left and 1 step forward, You see a green ball 1 step left and 3 steps forward, You
see a grey box 3 steps forward, You see a grey key 1 step right and 3 steps forward, You see a blue box
2 steps right and 3 steps forward
行动: 思考: 我看到一个绿色的钥匙在左边两步和前面一步。接下来, 我需要去拿绿色钥匙。
Observation:OK
Action:go forward
Observation:You see a wall 3 steps right, You see a blue key 3 steps left and 1 step forward, You see a
green key 2 steps left, You see a green ball 1 step left and 2 steps forward, You see a grey box 2 steps
forward, You see a grey key 1 step right and 2 steps forward, You see a blue box 2 steps right and 2
steps forward
Action:turn left
Observation:You see a wall 3 steps left, You see a green box 2 steps left and 3 steps forward, You see
a green box 2 steps left and 1 step forward, You see a green key 2 steps forward, You see a blue key 1
step right and 3 steps forward, You see a green ball 2 steps right and 1 step forward, You see a grey
box 2 steps right
Action:go forward
Observation:You see a green box 2 steps left and 2 steps forward, You see a green box 2 steps left, You
see a green key 1 step forward, You see a blue key 1 step right and 2 steps forward, You see a green
ball 2 steps right

```

中, LLM 会采样一些动作候选。对于每个动作候选, LLM 会生成一个回滚路径, 直到达到最大步数或终端状态。LLM 访问的回滚路径上所有动作的对数概率之和被用作每个动作候选的 Q 值。具有最高 Q 值的候选项被选中在环境中执行。

B.2.3. ICPI

ICPI (Brooks et al., 2024) 提出了通过上下文学习使用 LLMs 实现策略迭代。在每一步中, ICPI 中的策略将抽样一些动作候选, 评论者将计算每个动作候选的 Q 值。具有最高 Q 值的动作候选被选择执行。

该策略是通过 LLMs 实施的, 并为其提供如 Table 16 和 Table 18 中的成功轨迹。

至于批评者, ICPI 促使 LLMs 直接给出每个步骤的数值奖励。给定当前历史和一个动作候选者, ICPI 中的批评者将从动作候选者开始展开一个轨迹。除了预测观察结果外, 批评者还将为展开轨迹的每一步提供数值奖励。展开的 (折扣) 回报被视为动作候选者的 Q 值。对于 ALFWorld 和 BabyAI-Text, 当智能体到达目标时, 我们定义奖励为 1。所有其他步骤的奖励为 0。提供给批评者的示例如在 Table 22 中用于 ALFWorld 和 Table 23 中用于 BabyAI-Text。

B.2.4. RAFA

RAFA (Liu et al., 2023) 的框架也类似于 RAP 或 ICPI。主要区别在于动作评估是如何进行的。

RAFA 使用 LLM 来评估每个动作候选项, 从而实现树搜索。与 ICPI 不同, RAFA 使用任务完成进度作为每一步的取值。他们让 LLM 将目标分解为子目标, 并使用每一步之后子目标的完成状态作为该步的取值。RAFA 根据预测观察评估子目标的完成状态。在 RAFA 中提供的示例类似于 Table 24 中的 ALFWorld, 以及 Table 25 中的 BabyAI-Text。

B.2.5. LATS

LATS (Zhou et al., 2024) 将大型语言模型的推理、行动和计划能力与蒙特卡洛树搜索 (Kocsis & Szepesvári, 2006) 以及外部反馈机制相结合, 以增强决策能力, 在网络导航上取得了竞争性的结果。我们采用官方实现来评估其在 WebShop 上的性能。

LATS 不能直接应用于我们使用的其他基准测试, 主要有两个原因。(1) LATS 需要能够将代理人返回到环境中的较早状态, 而 ALFWorld 和 BabyAI-Text 不支持这种操作。LATS 依赖于无模型的 MCTS, 使用环境模拟器作

Table 20. WebShop 成功轨迹的一个例子

```

Webshop
Instruction:
i would like a 3 ounce bottle of bright citrus deodorant for sensitive skin, and price lower than 50.00
dollars
[Search]

Action: search[3 ounce bright citrus deodorant sensitive skin]
Observation:
[Back to Search]
Page 1 (Total results: 50)
[Next >]
[B078GWRC1J]
Bright Citrus Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding,
Contains Organic Calendula 3-Ounce
$ 10.99
[B078GTKVXY]
Ginger Fresh Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding,
Contains Organic Calendula 3-Ounce
$ 10.99
[B08KBVJ4XN]
Barrel and Oak - Aluminum-Free Deodorant, Deodorant for Men, Essential Oil-Based Scent, 24-Hour Odor
Protection, Cedar & Patchouli Blend, Gentle on Sensitive Skin (Mountain Sage, 2.7 oz, 2-Pack)
$ 15.95

Action: click[B078GWRC1J]
Observation:
[Back to Search]
[< Prev]
scent [assorted scents][bright citrus][calming lavender][ginger fresh][simply non-scents]
size [travel set (4-pack)][3 ounce (pack of 1)][3-ounce (2-pack)]
Bright Citrus Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding,
Contains Organic Calendula 3-Ounce
Price: $ 10.99
Rating: N.A.
[Description]
[Features]
[Reviews]
[Buy Now]

Action: click[bright citrus]
Observation: You have clicked bright citrus.

Action: click[3 ounce (pack of 1)]
Observation: You have clicked 3 ounce (pack of 1).

Action: click[Buy Now]
Observation: Your score (min 0.0, max 1.0): 1.0

```

Table 21. ReAct 中 WebShop 的成功轨迹示例

Webshop
Instruction:
i would like a 3 ounce bottle of bright citrus deodorant for sensitive skin, and price lower than 50.00 dollars
[Search]

Action: search[3 ounce bright citrus deodorant sensitive skin]
Observation:
[Back to Search]
Page 1 (Total results: 50)
[Next >]
[B078GWRC1J]
Bright Citrus Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding, Contains Organic Calendula 3-Ounce
\$ 10.99
[B078GTKVXY]
Ginger Fresh Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding, Contains Organic Calendula 3-Ounce
\$ 10.99
[B08KBVJ4XN]
Barrel and Oak - Aluminum-Free Deodorant, Deodorant for Men, Essential Oil-Based Scent, 24-Hour Odor Protection, Cedar & Patchouli Blend, Gentle on Sensitive Skin (Mountain Sage, 2.7 oz, 2-Pack)
\$ 15.95

行动: 认为 [[B078GWRC1J] 和 [B078GTKVXY] 是价格低于 50 美元的明亮柑橘香除臭剂。我可以先检查 [B078GWRC1J]。]
Observation: OK.

Action: click[B078GWRC1J]
Observation:
[Back to Search]
[< Prev]
scent [assorted scents][bright citrus][calming lavender][ginger fresh][simply non-scents]
size [travel set (4-pack)][3 ounce (pack of 1)][3-ounce (2-pack)]
Bright Citrus Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding, Contains Organic Calendula 3-Ounce
Price: \$ 10.99
Rating: N.A.
[Description]
[Features]
[Reviews]
[Buy Now]

动作: 思考 [对于 3 盎司装的明亮柑橘味适用于敏感肌肤的除臭剂, 商品有 [明亮柑橘] 和 [3 盎司 (1 包)] 的选项, 似乎值得购买。]
Observation: OK.

Action: click[bright citrus]
Observation: You have clicked bright citrus.

Action: click[3 ounce (pack of 1)]
Observation: You have clicked 3 ounce (pack of 1).

Action: click[Buy Now]
Observation: Your score (min 0.0, max 1.0): 1.0

Table 22. 在 ALFWorld 中提供给 ICPI 的一个例子

You are in the middle of a room. Looking quickly around you, you see a cabinet 4, a cabinet 3, a cabinet 2, a cabinet 1, a countertop 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a sinkbasin 2, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, and a towelholder 1.
 Your task is to: put some spraybottle on toilet.
 > go to cabinet 1
 On the cabinet 1, you see a cloth 1, a soapbar 1, a soapbottle 1.
 奖励: 0
 > go to cabinet 2
 The cabinet 2 is closed.
 奖励: 0
 > open cabinet 2
 You open the cabinet 2. The cabinet 2 is open. In it, you see a candle 1, and a spraybottle 2.
 奖励: 0
 > take spraybottle 2 from cabinet 2
 You pick up the spraybottle 2 from the cabinet 2.
 奖励: 0
 > go to toilet 1
 On the toilet 1, you see a soapbottle 2.
 奖励: 0
 > put spraybottle 2 in/on toilet 1
 You put the spraybottle 2 in/on the toilet 1.
 奖励: 1

Table 23. 在 ICPI 中提供给 BabyAI-Text 的一个例子

Goal of the agent: go to the green key
 Observation: You see a wall 2 steps right, You see a wall 3 steps forward, You see a grey box 3 steps left, You see a grey key 3 steps left and 1 step forward, You see a blue box 3 steps left and 2 steps forward
 Action: turn left
 Observation: You see a wall 3 steps right, You see a blue key 3 steps left and 2 steps forward, You see a green key 2 steps left and 1 step forward, You see a green ball 1 step left and 3 steps forward, You see a grey box 3 steps forward, You see a grey key 1 step right and 3 steps forward, You see a blue box 2 steps right and 3 steps forward
 奖励: 0
 Action: go forward
 Observation: You see a wall 3 steps right, You see a blue key 3 steps left and 1 step forward, You see a green key 2 steps left, You see a green ball 1 step left and 2 steps forward, You see a grey box 2 steps forward, You see a grey key 1 step right and 2 steps forward, You see a blue box 2 steps right and 2 steps forward
 奖励: 0
 Action: turn left
 Observation: You see a wall 3 steps left, You see a green box 2 steps left and 3 steps forward, You see a green box 2 steps left and 1 step forward, You see a green key 2 steps forward, You see a blue key 1 step right and 3 steps forward, You see a green ball 2 steps right and 1 step forward, You see a grey box 2 steps right
 奖励: 0
 Action: go forward
 Observation: You see a green box 2 steps left and 2 steps forward, You see a green box 2 steps left, You see a green key 1 step forward, You see a blue key 1 step right and 2 steps forward, You see a green ball 2 steps right
 奖励: 1

Table 24. 为 ALFWorld 在 RAFA 中提供的一个批评例子

You are in the middle of a room. Looking quickly around you, you see a cabinet 4, a cabinet 3, a cabinet 2, a cabinet 1, a countertop 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a sinkbasin 2, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, and a towelholder 1.
 Your task is to: put some spraybottle on toilet.
 评论者: 我的任务需要按顺序完成两个子目标: 拿一个喷雾瓶并将喷雾瓶放在马桶上。我的当前状态满足这两个子目标中的零个。其值为 $0/2=0$ 。
 > OK.
 On the cabinet 1, you see a cloth 1, a soapbar 1, a soapbottle 1.
 > OK.
 The cabinet 2 is closed.
 > OK.
 You open the cabinet 2. The cabinet 2 is open. In it, you see a candle 1, and a spraybottle 2.
 > OK.
 You pick up the spraybottle 2 from the cabinet 2.
 > 批评者: 现在我拿了一个喷壶。我当前的状态满足了两个子目标中的第一个: 拿一个喷壶。这个值是 $1/2=0.5$ 。
 > OK.
 On the toilet 1, you see a soapbottle 2.
 > OK.
 You put the spraybottle 2 in/on the toilet 1.
 评论者: 现在我把喷雾瓶放在马桶上。我当前的状态满足了所有两个子目标。值为 $2/2=1$ 。

Table 25. 在 BabyAI-Text 的 RAFA 中提供给评论家的一个例子

Goal of the agent: go to the green key
 You see a wall 2 steps right, You see a wall 3 steps forward, You see a grey box 3 steps left, You see a grey key 3 steps left and 1 step forward, You see a blue box 3 steps left and 2 steps forward
 评论者: 我的任务需要两个子目标按顺序完成: 找到绿色钥匙, 前往绿色钥匙。我的当前状态不满足这两个子目标中的任何一个。该值为 $0/2=0$ 。

>OK.
 You see a wall 3 steps right, You see a blue key 3 steps left and 2 steps forward, You see a green key 2 steps left and 1 step forward, You see a green ball 1 step left and 3 steps forward, You see a grey box 3 steps forward, You see a grey key 1 step right and 3 steps forward, You see a blue box 2 steps right and 3 steps forward
 > 批评者: 现在我找到了绿色钥匙。我的当前状态满足了两个子目标中的第一个: 找到绿色钥匙。这个值是 $1/2=0.5$ 。

>OK.
 You see a wall 3 steps right, You see a blue key 3 steps left and 1 step forward, You see a green key 2 steps left, You see a green ball 1 step left and 2 steps forward, You see a grey box 2 steps forward, You see a grey key 1 step right and 2 steps forward, You see a blue box 2 steps right and 2 steps forward
 >OK.
 You see a wall 3 steps left, You see a green box 2 steps left and 3 steps forward, You see a green box 2 steps left and 1 step forward, You see a green key 2 steps forward, You see a blue key 1 step right and 3 steps forward, You see a green ball 2 steps right and 1 step forward, You see a grey box 2 steps right
 >OK.
 You see a green box 2 steps left and 2 steps forward, You see a green box 2 steps left, You see a green key 1 step forward, You see a blue key 1 step right and 2 steps forward, You see a green ball 2 steps right
 评论者: 现在我去拿绿色的钥匙。我当前的状态满足所有两个子目标。这个值是 $2/2=1$ 。

为世界模型, 在树搜索过程中将模拟器返回到较早状态。在他们的原始论文(第 9 页)中也注意到了这一限制。(2) 虽然可以修改这些环境以使它们可逆, 但这会造成不公平的比较。我们的方法和其他基线在 ALFWorld 和 BabyAI-Text 中的推理过程中不依赖于模拟器, 而 LATS 会从这种修改中获得优势。

尽管如此, 我们仍尝试通过使用大型语言模型作为世界模型来调整 LATS 以适应 ALFWorld, 类似于我们的方法, 以进行公平比较。结果在 Table 26 中展示, 显示 LATS 在几乎所有任务中都失败了。这是因为其树搜索严重依赖于环境模拟器以实现精确的状态转换。仅凭基于 LLM 的世界模型, 状态转换往往偏离实际环境, 这是由于 LLM 固有的幻想和 ALFWorld 的部分可观测性所致。

Table 26. 在 ALFWorld 中, LAC (我们的) 和 LATS 的性能比较。

	CodeLlama-7B	Gemma-7B	Llama-3-8B	Mistral-7B
LAC (Ours)	0.79	0.84	0.78	0.79
LATS	0.00	0.00	0.03	0.00

B.3. LAC 的实现细节、消融研究和变体

我们的方法基于 ReAct (Yao et al., 2023) 构建。具体来说, 我们采用来自 ReAct 的上下文示例, 并为这些轨迹注释反思。对于 ALFWorld 中的每种任务类型, 我们从 ReAct 中注释两个轨迹作为成功示例, 以教导 LLMs 如何完成这种类型的任务。对于 BabyAI-Text, 我们为每种任务类型注释两个带有来自 ReAct 的思考以及反思的轨迹。至于 WebShop, 我们注释来自 ReAct 的示例以及反思, 并删除了在 ReAct 中注释的思考, 因为我们发现没有思考可以取得更好的结果。在 ALFWorld、BabyAI-Text 和 WebShop 中注释的轨迹分别展示在 Table 27、Table 28 和 Table 29 中。我们注释的反思标记为浅橙色。我们还使用 ReAct 算法从训练集中收集每种任务类型在 ALFWorld 和 BabyAI-Text 中的一些失败轨迹。所收集的失败轨迹也被标注了反思, 旨在教导 LLMs 区分对于实现目标有益、有害或不确定的步骤。

我们使用以下标准对反思进行注释:

- (1) 如果某个动作对于实现最终目标是必要的, 那么接下来的反思将以“这一步是好的。”结束, 以指示完成了一个必要步骤, 例如对于目标“把喷瓶放在马桶上”来说, 动作“从柜子 2 取喷瓶 2”。
- (2) 如果某个动作对于获取完成目标所必需的信息是至关重要的, 那么随后反思将以“这一步是好的。”结束, 例如在 Table 27 中执行的动作“打开柜子 2”找到了目标对象。

Table 27. 用于 LAC 中的 ALFWorld 的成功轨迹示例

You are in the middle of a room. Looking quickly around you, you see a cabinet 4, a cabinet 3, a cabinet 2, a cabinet 1, a countertop 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a sinkbasin 2, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, and a towelholder 1.
Your task is to: put some spraybottle on toilet.
想：为了解决这个任务，我需要找到并拿一个喷雾瓶，然后把它放在马桶上。
OK.
批评者：这个想法是合理的。这一步是好的。
想：首先我需要找到一个喷雾瓶。喷雾瓶更有可能出现在橱柜（1-4）、台面（1）、马桶（1）、洗脸盆（1-2）、垃圾桶（1）。我可以一个一个地检查，从橱柜 1 开始。
OK.
批评者：我需要首先找到喷雾瓶。这个想法是合理的。这一步是好的。
> go to cabinet 1
On the cabinet 1, you see a cloth 1, a soapbar 1, a soapbottle 1.
==> 评论者：我在这个地方没有看到喷雾瓶。这个步骤是未知的。
> go to cabinet 2
The cabinet 2 is closed.
==> 批评：柜子 2 是关闭的。这一步是未知的。
> open cabinet 2
You open the cabinet 2. The cabinet 2 is open. In it, you see a candle 1, and a spraybottle 2.
评论家：我在这个地方见过喷嘴 2。这个步骤很好。
想：现在我找到了一个喷雾瓶（2）。接下来，我需要拿起来。
OK.
评论者：我可以在找到喷雾瓶后拿走它。这个想法是合理的。这个步骤很好。
> take spraybottle 2 from cabinet 2
You pick up the spraybottle 2 from the cabinet 2.
==> 评论员：我在这个地方拿了喷雾瓶 2。这个步骤很好。
想：现在我拿一个喷雾瓶（2）。接下来，我需要把它放在马桶 1 内/上。
OK.
==> 批评者：我可以在拿到喷雾瓶后把它放在马桶上。这个想法是合理的。这个步骤是好的。
> go to toilet 1
On the toilet 1, you see a soapbottle 2.
评论家：如果我想把喷嘴放在上面，我应该去厕所。这个步骤是好的。
> put spraybottle 2 in/on toilet 1
You put the spraybottle 2 in/on the toilet 1.
==> 评论者：我已经把喷雾瓶 2 放在马桶 1 上/内。这个步骤是好的。

Table 28. LAC 中 BabyAI-Text 的一个标注成功轨迹示例

Goal of the agent: go to the green key
Observation: You see a wall 2 steps right, You see a wall 3 steps forward, You see a critic box 3 steps left, You see a critic key 3 steps left and 1 step forward, You see a blue box 3 steps left and 2 steps forward
行动：思考：为了完成任务，我需要找到一把绿色的钥匙并去那里。
Observation: OK
评论者：这个想法是合理的。这一步很好。
行动：思考：我还没有看到绿色钥匙。首先，我需要找到一把绿色钥匙。
Observation: OK
评审：我首先需要找到一把绿色的钥匙。这个想法是合理的。这一步很好。
Action: turn left
Observation: You see a wall 3 steps right, You see a blue key 3 steps left and 2 steps forward, You see a green key 2 steps left and 1 step forward, You see a green ball 1 step left and 3 steps forward, You see a critic box 3 steps forward, You see a critic key 1 step right and 3 steps forward, You see a blue box 2 steps right and 3 steps forward
评论者：我向左转，看到了一个绿色的钥匙，距离左边 2 步，向前 1 步。这个步骤是好的。
行动：思考：我看到一个绿色钥匙在左边 2 步和前面 1 步。接下来我需要去拿绿色钥匙。
Observation: OK
评论者：我在看到绿色钥匙后可以过去。这一想法是合理的。这个步骤不错。
Action: go forward
Observation: You see a wall 3 steps right, You see a blue key 3 steps left and 1 step forward, You see a green key 2 steps left, You see a green ball 1 step left and 2 steps forward, You see a critic box 2 steps forward, You see a critic key 1 step right and 2 steps forward, You see a blue box 2 steps right and 2 steps forward
评论者：我往前走了。绿色钥匙在左边两步。这一步是未知的。
Action: turn left
Observation: You see a wall 3 steps left, You see a green box 2 steps left and 3 steps forward, You see a green box 2 steps left and 1 step forward, You see a green key 2 steps forward, You see a blue key 1 step right and 3 steps forward, You see a green ball 2 steps right and 1 step forward, You see a critic box 2 steps right
评估者：我已经向左转了。绿色钥匙在前方两步。这一步是未知的。
Action: go forward
Observation: You see a green box 2 steps left and 2 steps forward, You see a green box 2 steps left, You see a green key 1 step forward, You see a blue key 1 step right and 2 steps forward, You see a green ball 2 steps right
评论员：我已经向前走了。绿色钥匙是向前一步。这个步骤是好的。

Table 29. 一个在 LAC 中成功的 WebShop 轨迹的例子

Webshop
Instruction:
i would like a 3 ounce bottle of bright citrus deodorant for sensitive skin, and price lower than 50.00 dollars
[Search]

Action: search[3 ounce bright citrus deodorant sensitive skin]
Observation:
[Back to Search]
Page 1 (Total results: 50)
[Next >]
[B078GWRC1J]
Bright Citrus Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding, Contains Organic Calendula 3-Ounce
\$ 10.99
[B078GTKVXY]
Ginger Fresh Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding, Contains Organic Calendula 3-Ounce
\$ 10.99
[B08KBVJ4XN]
Barrel and Oak - Aluminum-Free Deodorant, Deodorant for Men, Essential Oil-Based Scent, 24-Hour Odor Protection, Cedar & Patchouli Blend, Gentle on Sensitive Skin (Mountain Sage, 2.7 oz, 2-Pack)
\$ 15.95
评论者: 说明中要求的物品是“适合敏感肌肤的明亮柑橘香味除臭剂”。我搜索了“3 盎司明亮柑橘香味适合敏感肌肤的除臭剂”,这与所需物品相似。我还看到 [B078GWRC1J] 是明亮的柑橘香味除臭剂。这个步骤不错。

Action: click[B078GWRC1J]
Observation:
[Back to Search]
[< Prev]
scent [assorted scents][bright citrus][calming lavender][ginger fresh][simply non-scents]
size [travel set (4-pack)][3 ounce (pack of 1)][3-ounce (2-pack)]
Bright Citrus Deodorant by Earth Mama | Natural and Safe for Sensitive Skin, Pregnancy and Breastfeeding, Contains Organic Calendula 3-Ounce
Price: \$ 10.99
Rating: N.A.
[Description]
[Features]
[Reviews]
[Buy Now]
评论者: 我点击了 [B078GWRC1J], 发现该商品是“清新柑橘除臭剂”,并且它符合指令中的属性,即“3 盎司”和“价格低于 50.00 美元”。这一步是好的。

Action: click[bright citrus]
Observation: You have clicked bright citrus.
批评者: 我点击了 [bright citrus] 来选择满足指令的选项。这个步骤很好。

Action: click[3 ounce (pack of 1)]
Observation: You have clicked 3 ounce (pack of 1).
评论者: 我点击了 [3 盎司 (1 包)] 来选择符合说明的选项。这个步骤是好的。

Action: click[Buy Now]
Observation: Your score (min 0.0, max 1.0): 1.0
评论者: 我已购买满足指令的物品。此步骤是好的。

- (3) 如果某个动作对于实现目标明显是无用或不正确的，接下来的反思将以“这一步是坏的。”结束，以指示这个动作是不合适的，例如在 Table 27 中的任务中执行动作“从柜子 1 拿布 1”。
- (4) 如果无法从历史中评估某个动作是好是坏，接下来的反思将以“这一步是未知的。”结束，例如在 Table 27 中执行动作“去柜子 1”或“去柜子 2”。

所有标注的成功和失败的轨迹都被用来构建微调示例，以微调 LLMs 以生成更好的反射。

考虑到完全微调大语言模型 (LLMs) 的计算成本，我们使用 LoRA (Hu et al., 2021) 来微调我们的模型。在 ALFWorld 中，对于每种任务类型有两个成功的轨迹和一个失败的轨迹，我们共有六种任务类型的 485 个 (输入, 输出) 对来微调模型。在 BabyAI-Text 中，微调 (输入, 输出) 对的数量是 418。我们以学习率 $2.5e-5$ 和批量大小为 2 对模型进行 1,000 步的微调。我们使用具有 80GB 内存的 A100 GPU 来微调我们的模型。仅凭大约 400-500 个 (输入, 输出) 对和 1,000 个微调步骤，我们可以在一个半小时内完成微调过程。

在测试期间，微调后的模型用于在执行环境中的动作后生成反思，并预测每个动作候选的潜在结果。

候选动作的数量 n 是一个超参数，在我们的实验中设定为 5，也就是说，我们评估从 π_{LLM} 针对每个状态采样的前 5 个候选动作。值得注意的是，采样顶级候选动作会引入丢失真实 $\argmax$ 动作的非零概率，特别是当候选动作之间的区分较细微时。然而，这种选择主要是出于计算实用性的考虑：显式计算或评估在以大型语言模型为基础的决策中常见的大型或开放式动作空间的完整动作分布通常是不可行的。根据实验，我们发现从一个强大的语言模型先验中生成一小部分候选动作通常足以包含有前途的动作，从而使得计算效率和准确性之间的权衡是可以接受的。在采样动作候选项之后，我们使用微调后的模型预测每个动作候选项的未来结果。模型需要预测可能的观察结果并为每个预测步骤生成反思。我们将最大预测步骤设定为 4，模型将继续预测，直到它生成一个以“这一步是好的。”或“这一步是不好的”结束的反思，或者达到最大预测步骤为止。

对于 π_{LLM} 的优化，我们在 Equation (4) 中解决一个优化问题，其中包含一个超参数 α ，该参数平衡了 π_{LLM} 的生成概率与由 Q_{LLM} 给出的值。对于 ALFWorld，我们将 α 设置为 1，这在基准测试中表现出色。对于 BabyAI-Text，我们对 $\{1/2, 1, 2, 5, 10\}$ 进行了 α 的网格搜索，发现不同的 LLM 在不同的 α 下会表现最佳。结果可以在 Table 12 中看到。对于 WebShop，我们也在 $\{1/10, 1, 10\}$ 上进行网格搜索以找到最佳的 α 。Equation (4) 是原始策略与动作评估值的加权组合。它通过调整顶级候选动作的概率更新分布，而其他动作的概率保持不变。

我们将 ALFWorld 的最大视界长度设置为 40，BabyAI-Text 设置为 30，WebShop 设置为 15。如果智能体在最大步数后还未达到最终目标，则此回合将被标记为失败。

我们使用具有 80GB 内存的 A100 GPU 来评估我们的方法。对于 LAC，使用单个 A100 GPU 执行 134 个任务的时间约为 10 小时。而对于 BabyAI-Text，执行时间可因不同任务类型而异，使用一个 A100 GPU 执行 50 个任务的时间范围为 4 到 10 小时。对于 WebShop，100 个任务将在 3 或 4 小时内完成。在交互过程中，GPU 内存使用可能根据 LLMs 输入的长度从 15GB 增加到 70GB 以上。

我们将我们的方法与所有上述基线进行比较，证明我们的方法在决策任务中对高层操作和低层操作都具有有效性，即便在可能有无限操作的情况下也是如此。为了展示我们方法中每个组件的有效性，我们进行了每个组件的消融研究。在前操作生成阶段，我们去除了 LAC 的反射以及集成。该变体称为 LAC w/o reflection。我们还通过去除 LAC 中的 Q_{LLM} 以及在后操作生成阶段的集成来评估 Q_{LLM} 的角色。这个变体称为 LAC w/o critic。我们还通过仅使用评论者 Q_{LLM} 进行决策，展示了由 LLM 策略给出的动作先验的角色。我们将这一变体称为 critic-only。在评估期间，这些变体的执行时间可能因性能而异，因为性能较差的方法通常需要更长的执行时间。在 ALFWorld 上，可能需要 10-20 小时。这些变体之间的比较显示在 Figure 4 中。

我们发现，LAC 中的每个组件对卓越的性能都至关重要。移除某些组件可能导致错误的动作候选选择。这样一个例子在 Table 30 中展示。LAC 可以成功完成这个任务，而在 LAC 中去除一些组件将导致失败。比较结果显示在 Figure 10 中。

Table 30. 由 LAC 给出的 BabyAI-Text 测试环境中的成功轨迹示例

Goal of the agent: go to the green ball

Observation: You see a wall 5 steps forward, You see a wall 2 steps left, You see a yellow ball 1 step left and 4 steps forward, You see a blue ball 1 step left and 3 steps forward, You see a red ball 1 step left and 1 step forward, You see a grey key 2 steps right and 2 steps forward, You see a green key 2 steps right and 1 step forward, You see a blue key 3 steps right and 1 step forward

行动: 思考: 为了完成任务, 我需要找到一个绿色的球并走向它。

Observation: OK

评论者: 这个想法是合理的。这一步是好的。

行动: 思考: 我看到一个绿色的球在左边 1 步, 前进 4 步。接下来, 我需要去拿那个绿色的球。

Observation: OK

批评者: 这个想法是合理的。我可以在看到绿色球后去那里。这个步骤是好的。

Action: turn right

Observation: You see a wall 5 steps forward, You see a wall 2 steps right, You see a green ball 3 steps left and 4 steps forward, You see a grey key 2 steps left and 2 steps forward, You see a blue key 1 step left and 3 steps forward, You see a green key 1 step left and 2 steps forward, You see a blue ball 1 step right and 1 step forward

批评者: 我已经向右转了。绿色球在左边 3 步, 向前 4 步。这一步是未知的。

Action: go forward

Observation: You see a wall 4 steps forward, You see a green ball 3 steps left and 3 steps forward, You see a grey key 2 steps left and 1 step forward, You see a blue key 1 step left and 2 steps forward, You see a green key 1 step left and 1 step forward, You see a blue ball 1 step right

评论者: 我已经前进了。绿色球在左边 3 步和前面 3 步。这一步未知。

Action: go forward

Observation: You see a wall 3 steps forward, You see a wall 2 steps right, You see a green ball 3 steps left and 2 steps forward, You see a grey key 2 steps left, You see a blue key 1 step left and 1 step forward, You see a green key 1 step left

评论者: 我向前走了。绿球在左边 3 步和前面 2 步。这一步未知。

Action: go forward

Observation: You see a wall 2 steps forward, You see a wall 2 steps right, You see a green ball 3 steps left and 1 step forward, You see a blue key 1 step left

批评者: 但是我已经向前走了。绿色球在左边 3 步, 前面 1 步。这一步是未知的。

Action: go forward

Observation: You see a wall 1 step forward, You see a wall 2 steps right, You see a green ball 3 steps left

评论者: 但我已经往前走了。绿色球向左移动了 3 步。这个步骤是未知的。

Action: turn left

Observation: You see a wall 1 step right, You see a grey key 2 steps left and 2 steps forward, You see a green key 2 steps left and 1 step forward, You see a blue key 1 step left and 1 step forward, You see a red ball 4 steps forward, You see a green ball 3 steps forward

批评者: 但是我已经左转了。绿色球向前三步。这一步是未知的。

Action: go forward

Observation: You see a wall 1 step right, You see a grey key 2 steps left and 1 step forward, You see a green key 2 steps left, You see a blue key 1 step left, You see a red ball 3 steps forward, You see a green ball 2 steps forward

批评者: 但是我已经向前走了。绿球向前走了 2 步。这一步是未知的。

Action: go forward

Observation: You see a wall 1 step right, You see a grey key 2 steps left, You see a red ball 2 steps forward, You see a green ball 1 step forward

评论者: 但我已经向前迈进了。绿色球向前迈了一步。这一步是好的。