arXiv:2601.00167v1 [cs.LG] 1 Jan 2026

使用纯强化学习梯度在线微调决策变换器

Junkai Luo加利福尼亚大学
河滨分校 junkail@ucr.edu

Yinglun Zhu†加利福尼亚大
学河滨分校 yzhu@ucr.edu

摘要

决策变换器（DT）通过将离线强化学习（RL）表述为序列建模问题，已成为序列决策领域一种强有力的框架。然而，如何将决策变换器扩展到带有 *pure RL gradients* 的在线设置中，仍鲜有研究，因为现有方法在在线微调期间仍高度依赖监督式序列建模目标。我们发现，事后回报重标记——在线决策变换器中的标准组成部分——是基于强化学习进行微调的一大障碍：尽管它有利于监督学习，但从根本上与 GRPO 等基于重要性采样的强化学习算法不兼容，导致训练不稳定。基于这一发现，我们提出了新的算法，使决策变换器能够使用纯强化学习梯度进行在线微调。我们将 GRPO 适配到决策变换器，并引入了若干关键改进，包括用于改善信用分配的子轨迹优化、用于提升稳定性和效率的序列级似然目标，以及用于鼓励在不确定区域进行探索的主动采样。通过大量实验，我们证明所提出的方法优于现有的在线决策变换器基线，并在多个基准测试中取得了新的最先进性能，彰显了基于纯强化学习的在线微调对决策变换器的有效性。

1 引言

变换器架构（Vaswani 等人，2017年）是现代基础模型的核心。尤其是大语言模型（LLM），通过一种简单而强大的范式展现出卓越的泛化与推理能力：先进行大规模预训练，再通过监督学习和基于强化学习（RL）的微调（Radford 等人，2018年；Brown 等人，2020年；Ouyang 等人，2022年；Achiam 等人，2023年；Comanici 等人，2025年）。受此成功启发，决策变换器（DT，Chen 等人，2021年）通过将经典强化学习重新表述为条件序列建模问题，把 Transformer 引入序列决策。作为一种离线强化学习方法，决策变换器在预先收集的轨迹上以监督目标进行训练，实际上执行的是模仿学习（Hussein 等人，2017年），同时以剩余回报（RTG）词元为条件。

在线决策变换器（ODT，Zheng 等人，2022年）通过支持离线预训练后的微调，将决策变换器扩展到在线环境。在在线微调期间，ODT 收集新的轨迹，并应用事后回报重标记，将预期的回报目标替换为实际获得的回报，使轨迹与其结果对齐，从而沿用离线训练流程。后续工作进一步将 TD3（Fujimoto 等人，2018年）梯度融入 ODT，形成 ODT+TD3（Yan 等人，2024年）。尽管取得了这些进展，现有的决策变换器在线微调方法仍以监督目标为主：ODT 完全依赖监督损失，而 ODT+TD3 仅赋予强化学习梯度较小的权重。

相比之下，大语言模型微调领域的最新进展表明，纯强化学习方法——例如组相对策略优化（GRPO，Shao 等人，2024年）——能够显著提升推理能力和对齐效果（Guo 等人，2025年；Team，2025年）。这种反差自然引出了一个重要问题：

能否使用纯强化学习梯度在线微调决策变换器？

为探究这一问题，我们重新审视现有在线决策变换器方法的训练范式，并发现了一项根本性局限。我们表明，事后回报重标记(Zheng 等人，2022年）虽然有利于

†项目负责人和通讯作者。

监督学习，但从根本上与依赖重要性采样的同策略强化学习算法不兼容。在轨迹展开期间使用的回报目标，会被事后观测到的回报重新标记，这会导致重要性比率不匹配，进而造成优化不稳定和性能下降（图 1）。因此，移除事后回报重标记是将 GRPO 和 PPO 等基于重要性采样的算法应用于决策变换器在线微调的必要第一步。

基于这一发现，我们提出一种利用纯强化学习梯度在线微调决策变换器的新框架。具体而言，我们将 GRPO 改编为决策变换器（GRPO-决策变换器），并引入三项关键改进：（i）基于子轨迹的优化目标，可实现细粒度的信用分配，并通过环境重置（Mhammedi 等人，2024年；Kazemnejad 等人，2025年）或辅助Q函数提供支持；（ii）序列级重要性比率，可提高训练稳定性和效率；（iii）主动采样策略，优先选择不确定状态以增强探索。这些组件相结合，使预训练决策变换器能够仅通过强化学习进行微调，并在多个基准测试中取得新的最先进性能。此外，我们还将近端策略优化（PPO，Schulman 等人，2017年）改编为决策变换器（PPO-决策变换器），在此前基于 PPO 的方法未能奏效的任务中取得了有竞争力的结果（Yan 等人，2024年）。

贡献。 我们的主要贡献总结如下：

(i) 我们发现，事后回报重标记——现有在线决策变换器方法中的标准组成部分——是基于强化学习的微调的一大障碍：虽然它适用于监督目标，却与基于重要性采样的策略梯度算法（如 GRPO 和 PPO）不兼容。(ii) 我们提出 GRPO-DT，这是一种针对决策变换器改造的 GRPO，通过整合子轨迹优化以改进信用分配、序列级重要性比率以提升稳定性和效率，以及主动状态采样以加强探索，实现纯强化学习在线微调 。这些改动具有通用性，其价值或不局限于决策变换器。(iii) 我们进行了广泛的实验，并证明纯强化学习微调——通过 GRPO-DT 和 PPO-DT 实现——优于现有在线决策变换器基线，并在多个基准测试中取得了新的最先进性能。

论文结构。 本文其余部分安排如下： 第2节回顾背景知识和预备内容。 第3节 介绍我们的方法。 第4节 报告实验结果。 第5节 讨论相关工作，并由第6节 总结全文。

2 预备知识

马尔可夫决策过程。 我们将强化学习环境表述为一个马尔可夫决策过程 （MDP），由元组 $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R)$ 定义。其中， $\mathcal{S}$ 是状态空间， $\mathcal{A}$ 是动作空间，$P(s_{h+1} \mid s_h, a_h)$ 是概率转移函数， $R(s_h, a_h)$ 是奖励函数。在每个时间步 $h = 1, \ldots, H$，智能体观测到状态 $s_h \in \mathcal{S}$，根据策略 $\pi(a_h \mid s_h)$ 选择动作 $a_h \in \mathcal{A}$，转移到下一个状态 $s_{h+1} \sim P(\cdot \mid s_h, a_h)$，并获得奖励 $r_h = R(s_h, a_h)$。目标是学习一种策略 π，使期望累积奖励 $\mathbb{E}_{\mathcal{M}, \pi}\left[\sum_{h=1}^{H} r_h\right]$ 最大化。

决策变换器（DT）。决策变换器（Chen 等人，2021年）是离线强化学习的一种强大范式，它将决策表述为基于预先收集的训练轨迹的序列建模问题。一条决策变换器轨迹由三种词元组成：回报目标（RTG）、状态和动作，其中 RTG $g_h := \sum_{h=h}^{H} r_h$ 表示从步骤 h 开始及之后的未来奖励总和。决策变换器利用变换器架构（Vaswani 等人，2017年）从预先收集的轨迹中自回归地学习策略。决策变换器在固定长度的轨迹片段上进行训练（Chen 等人，2021年）：令 m 表示上下文长度，决策变换器根据过去的交互 $((g, s, a)_{h-m+1:h-1}, g_h, s_h)$ 学习生成下一个动作 $a_h := (g_{h-m+1}, s_{h-m+1}, a_{h-m+1}, \ldots, g_h, s_h)$，其上下文长度为 m。模型通过监督学习进行训练，最小化预测动作 $\pi(a_h \mid ((g, s, a)_{h-m+1:h-1}, g_h, s_h))$ 与真实动作 a_h 之间的均方误差（MSE）。上下文明确时，我们使用

简写 $\pi(a_h \mid s_h,\ g_h) := \pi(a_h \mid ((g,\ s,\ a)_{h-m+1:h-1},\ g_h,\ s_h))$。在评估和部署期间，由于无法预先得知真实的未来回报目标，学习器会指定一个期望的初始回报目标，并利用决策变换器自回归地生成下一个动作，与环境交互。

决策变换器的在线微调。在线决策变换器（ODT，Zheng 等人，2022年）将离线决策变换器扩展到在线环境：先进行离线预训练，再利用交互收集的数据进行在线微调。离线预训练阶段基本遵循标准的决策变换器训练流程。在线微调阶段，将决策变换器部署到环境中，并设定期望的初始回报目标 g_{online} 以收集新的轨迹，逐步替换回放缓冲区中存储的旧轨迹；回放缓冲区以离线轨迹初始化。对于在线收集的轨迹，ODT 会根据实际达到的回报目标进行事后回报重标记(Andrychowicz 等人，2017年；Ghosh 等人，2019年），以重新标记回报目标词元 g_{actual}。ODT 采用随机高斯策略，以适应在线环境中的探索。不过，与离线决策变换器类似，ODT 仍通过监督学习目标进行学习，即最小化负对数似然损失。

尽管 ODT 能在在线微调期间提升模型性能，但最近，Yan 等人（2024年）指出，由于仅使用监督学习目标，ODT 在离线数据质量中等或较低的情形下会失效。监督学习目标学习$\partial^{\partial a}$ 回报目标，即动作如何随目标回报目标的变化而变化，因为决策变换器模型以回报目标为条件生成动作。然而，真正驱动在线策略改进的是 $^{\partial}$回报目标∂a ，即回报目标如何响应动作调整，尤其是在离线预训练数据质量不高时；更多详情请参见Yan 等人（2024年）的第 3.1 节。为实现更好的在线改进，Yan 等人（2024年）提出了 ODT+TD3，该方法将 TD3 的强化学习梯度（Fujimoto 等人，2018年；Fujimoto 和 Gu，2021年）与监督式 ODT 目标函数（结合事后回报重标记）相结合，以引导在线探索与适应。然而，ODT+TD3 仍然优先考虑监督式 ODT 损失，在优化期间仅赋予强化学习梯度较小的权重。

组相对策略优化（GRPO）。GRPO 是一种强化学习算法，最初用于大语言模型（LLM）的微调（Shao 等人，2024年；Guo 等人，2025年）。它通过移除估计优势值所需的价值模型，简化了近端策略优化（PPO，Schulman 等人，2017年）。GRPO 会针对每个问题采样多个回答，并将其平均奖励用作计算优势值的基线。具体而言，对于从问题分布 $q \sim \Delta(Q)$ 中采样的每个查询 $\Delta(Q)$，模型会基于策略 G 生成一组 $\{o_1, \cdots, o_G\}$ 个回答 $\pi_{\theta_{\text{old}}}$。每个回答 r_i 都会被计算一个奖励 o_i，通常借助奖励模型完成。GRPO 通过最大化以下目标函数来优化策略模型：

$$J_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim\Delta(Q),\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{h=1}^{|o_i|}\min\left(w_{i,h}(\theta)\widehat{A}_i,\ \text{clip}\left(w_{i,h}(\theta), 1-\varepsilon, 1+\varepsilon\right)\widehat{A}_i\right) - \beta D_{\text{KL}}\left(\pi_\theta \parallel \pi_{\theta_{\text{ref}}}\right), \tag{1}$$

其中，$\widehat{A}_i = \frac{r_i - \text{mean}(\{r_1, r_2, \cdots, r_G\})}{\text{std}(\{r_1, r_2, \cdots, r_G\})}$ 表示第 i 条轨迹展开的优势值，$w_{i,\ h}(\theta) = \frac{\pi_\theta(o_{i,\ h}|q,\ o_{i,\ <h})}{\pi_{\theta_{\text{old}}}(o_{i,\ h}|q,\ o_{i,\ <h})}$ 表示重要性采样比，而 $D_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_{\text{ref}}})$ 表示 KL 惩罚，用于防止策略偏离参考策略 $\pi_{\theta_{\text{ref}}}$ 太远。在我们的实现中，参考策略 $\pi_{\theta_{\text{ref}}}$采用缓慢更新方案，每对 π_θ 更新四次，便更新一次参考策略。

3 方法

在第3.1节中，我们首先分析使用基于重要性采样的强化学习算法，将决策变换器适配到在线环境时的一项关键局限，并讨论如何克服这一局限。基于这些见解，第3.2节介绍了我们对 GRPO 的适配，将其应用于决策变换器，并融入了若干关键改动。我们在第3.3节中介绍了其他扩展。

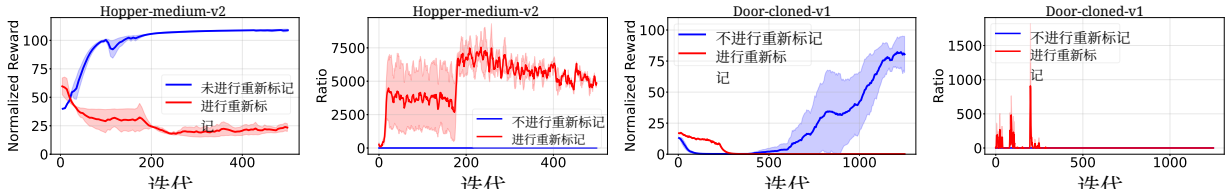


图1：比较 GRPO-DT 在对采样子轨迹进行处理时，采用和不采用事后回报重标记的效果。事后回报重标记会导致重要性采样比率 $\frac{\pi_\theta(a|s,g)}{\pi_{\theta_{\text{old}}}(a|s,g)}$ 变得极不稳定，进而降低性能。

3.1 移除事后回报重标记

部署决策变换器时，学习者必须指定期望的初始回报目标，因为未来的真实回报目标事先未知。对于在线决策变换器（ODT），学习者通常会在 g_{online} 轨迹展开期间设定一个相对较高的回报目标，以鼓励乐观探索。在训练期间，ODT 的一个关键组成部分——称为事后回报重标记——会根据实际达到的回报目标替换回报目标词元g_{actual} （Zheng 等人，2022年）。

当使用 TD3 的强化学习梯度增强监督式 ODT 目标函数时，ODT+TD3（Yan 等人，2024年）也采用事后回报重标记步骤。 Yan 等人（2024年）还尝试使用 PPO（Schulman 等人，2017年）对决策变换器进行在线微调——PPO 是一种基于重要性采样的强化学习算法——但他们发现，PPO 梯度会导致性能不佳，因此最终改用 TD3 梯度。

尽管事后回报重标记在监督学习目标函数下效果良好（参见 Zheng 等人 (2022) 中的图5.4），但我们发现，它与依赖比率$\frac{\pi_\theta(a|s,g)}{\pi_{\theta_{\text{old}}}(a|s,g)}$ 的基于重要性采样的强化学习梯度并不兼容，PPO 和 GRPO 就是如此。问题源于条件变量 g 不匹配：学习器以较高的回报目标 g_{online} 为条件进行滚动采样，以实现乐观探索，但实际达到的回报目标 g_{actual} 往往低得多。如果应用事后回报重标记，从 $\pi_{\theta_{\text{old}}}(a \mid s, g_{\text{online}})$ 中采样的动作之后会被当作来自 $\pi_{\theta_{\text{old}}}(a \mid s, g_{\text{actual}})$ 进行训练，从而产生不可靠的重要性权重和不稳定更新。这种不一致解释了为何将 PPO 直接用于 ODT 往往会失败（Yan 等人，2024）。如图 1 所示，移除事后回报重标记能显著提升稳定性和整体性能。在 MuJoCo Hopper 这类较简单的环境中，重新标记可能带来短暂收益，但最终会导致崩溃；而在 Adroit Door 这类更复杂的环境中，启用重新标记后，模型则完全无法学会任务。

此外，事后回报重标记需要获取整条轨迹的回报，这在对子轨迹展开进行优化时（见第3.2节中介绍）图1所示的实验中，我们通过展开完整轨迹，对子轨迹应用事后回报重标记。

3.2 将 GRPO 应用于决策变换器

我们在算法1 (GRPO-DT)中概述了使用 GRPO 对决策变换器进行在线微调的方法，该方法通过对子轨迹而非原始 GRPO 公式（Shao 等人，2024年；Guo 等人，2025年）中使用的完整轨迹进行优化来实现。每次迭代中，当前策略与环境交互以收集完整轨迹展开，并从中采样重置点，针对每个重置点生成一组子轨迹。重置点还会根据策略动作分布的方差进行主动选择。算法根据公式（2）计算每条子轨迹的优势值。随后，利用这些子轨迹及其优势值，通过序列级重要性比率（见公式（3））更新策略。

与原始 GRPO 相比，我们的方法引入了三项关键设计改动，以更好地使 GRPO 适配决策变换器。具体而言，(i) 我们重新设计了优化目标，使其针对一组子轨迹而非完整轨迹展开进行优化，这可以通过重置或学习一个额外的 Q函数来实现；(ii) 我们在序列级计算重要性权重，以更好地对齐计算得到的优势值；(iii) 我们

算法1 使用 GRPO 在线微调决策变换器（GRPO-DT） 输入：预训练策略 π_{θ_1}、完整轨迹缓冲区 $\mathcal{T}_{\text{replay}}$、子轨迹缓冲区 $\mathcal{T}_{\text{sub}}$、迭代次数 T、初始回报目标 g_{online}、每条轨迹中的重置点数量 K、子轨迹长度 L_{traj}、评估步数L_{eval}、GRPO 的组大小 G。1: 对于迭代 $t = 1, \cdots, T$，执行 2: 使用当前策略 $\pi_{\theta_t}(\cdot \mid s_1, g_{\text{online}})$，以（随机化的）初始状态 s_1 和回报目标 g_{online}为条件，执行完整轨迹 τ；使用 τ更新 $\mathcal{T}_{\text{replay}}$。// 收集完整轨迹；更新先进先出缓冲区。3: 从分布 p中，从 $\mathcal{T}_{\text{replay}}$采样一批完整轨迹 $\mathcal{B}$，其中 $p(\tau) := \frac{|\tau|}{\sum_{\tau \in \mathcal{T}} |\tau|}$。4: 对每条完整轨迹 $\tau \in \mathcal{B}$，执行 5: 从动作方差分布中采样 K 个重置点 $\{(s_k, g_k)\}_{k=1}^{K}$。6: 对每个重置点 (s_k, g_k)，使用当前策略 π_{θ_t}生成 G 条长度为 L_{traj}的子轨迹 $\{\tau_{k_i}^{\text{sub}}\}_{i=1}^{G}$；对每条子轨迹额外评估 L_{eval}步，以获得奖励 $R(\tau_{k_i}^{\text{sub}})$。// 子轨迹生成与评估。7: 使用公式（2）计算每条子轨迹 $\tau_{k_i}^{\text{sub}}$的优势值 $\widehat{A}_{k_i}$。8: 使用 $\{(\tau_{k_i}^{\text{sub}}, \widehat{A}_{k_i}, (s_k, g_k))\}_{i=1}^{|G|}$更新子轨迹缓冲区 $\mathcal{T}_{\text{sub}}$。// 更新先进先出缓冲区。9: 使用序列级重要性比率（公式（3））对 $\mathcal{T}_{\text{sub}}$ 中的子轨迹微调当前策略，得到新策略$\pi_{\theta_{t+1}}$。输出：在线微调策略 $\pi_{\theta_{T+1}}$。

加入一种主动采样 机制，优先对不确定状态进行优化。我们将在下文详细介绍这些设计选择，并通过消融实验展示其有效性。

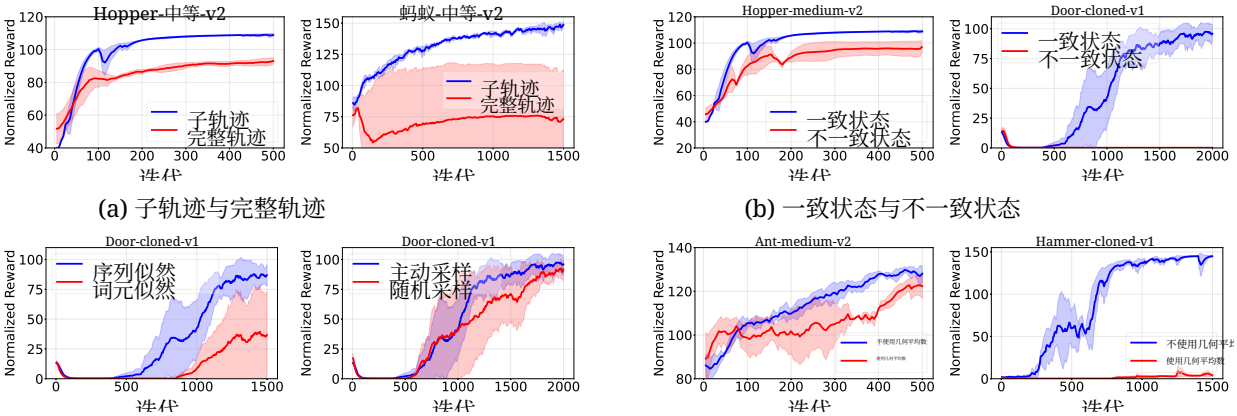
通过子轨迹展开实现更好的信用分配。在面向语言模型的原始形式中，GRPO 会为同一序列中的所有词元分配单个响应级奖励（Shao 等人，2024年；Guo 等人，2025年），因而舍弃了细粒度的信用分配。将其直接应用于经典强化学习环境和任务时，一种简单的方法是汇总一次轨迹展开中的所有逐步奖励，并将这一轨迹级回报均匀分配给每个时间步。然而，这种形式在强化学习环境中表现不佳：如图2a所示，在 Ant-medium-v2 环境中使用完整轨迹进行训练时，模型无法学会任务。这一局限在意料之中，因为强化学习任务，尤其是连续控制任务，比语言建模更需要精准的信用分配。句子中的词元往往具有连贯的相关性，而强化学习中的动作则可能导致截然不同的结果（例如，在迷宫中导航时选择不同的动作）。

为解决这一局限，我们使用子轨迹表述为决策变换器调整 GRPO。首先，从每条完整轨迹中选取 K 重置点 $\{(s_k,\ g_k)\}_{k=1}^{K}$。对于每个重置点（$s_k$，$g_k$），我们使用当前策略 π_{θ_t}生成长度为 L_{traj} 的G 子轨迹 $\{\tau_{k_i}^{\text{sub}}\}_{i=1}^{|G|}$。为了更好地将奖励归因于每条子轨迹，我们还使用期望动作（离散情形下则使用最可能的动作），对每条子轨迹额外执行 L_{eval} 步。子轨迹 $\tau_{k_i}^{\text{sub}}$ 的奖励 $r_{k_i}^{\text{sub}}$ 定义为连续执行 $L_{\text{traj}} + L_{\text{eval}}$ 步所获累积折扣奖励，其中折扣因子 γ 会提高重置点附近所获奖励的权重。对于额外的 L_{eval} 步，我们使用期望动作，以减少随机动作采样造成的方差，从而获得更稳定的子轨迹回报估计。接着，我们计算每组中每条子轨迹的优势，公式如下：

$$\widehat{A}_{k_i} = \frac{r_{k_i}^{\text{sub}} - \text{mean}(\{r_{k_1}^{\text{sub}}, r_{k_2}^{\text{sub}}, \ldots, r_{k_{|G|}}^{\text{sub}}\})}{\text{std}(\{r_{k_1}^{\text{sub}}, r_{k_2}^{\text{sub}}, \ldots, r_{k_{|G|}}^{\text{sub}}\})}. \tag{2}$$

仅长度为 L_{traj} 的子轨迹用于 GRPO 优化，而后续的 L_{eval} 步仅用于评估。参数 L_{traj} 控制信用分配的粒度，而 L_{eval}则决定奖励估计的质量。根据实验结果，我们发现，较小的 L_{traj} 搭配较大的 L_{eval} 效果最佳；有关这些超参数的详细消融实验，请参见 第 4.3 节 。

为确保优化稳定，我们通过在每组生成子轨迹之前，将向量化环境重置到相同状态，来保证状态一致性。这一重置机制对于收敛至关重要，



(a) 子轨迹与完整轨迹　(b) 一致状态与不一致状态

(c) 序列与词元　(d) 主动与随机　（e）移除序列级比率中的几何平均数

图 2：展示 GRPO-DT 中关键设计选择影响的消融研究。我们在多个环境中比较了包含和不包含若干所提组件的 GRPO-DT 变体。（a）与在完整轨迹上进行更新相比，在子轨迹上进行优化能够实现更快的收敛速度和更高的最终性能。（b）从一致状态中采样组，相较于混合不一致状态，能够显著提升学习稳定性。（c）使用序列级似然计算重要性比率，相较于使用逐词元似然，能够产生更可靠的更新。（d）根据不确定性主动选择重置点，相较于随机采样能够加速学习。（e）在子轨迹模式下计算序列级重要性比率时，移除几何平均数（GM）归一化能够提升性能。

如图 2b所示。在无法进行环境重置的情况下，我们使用 TD3（Fujimoto 等人，2018年）训练一个辅助 Q 函数，以在共享状态下评估候选动作（详情见第 3.3.1 节）。这一由 Q 函数引导的变体同样取得了具有竞争力的性能，如第 4.3 节所示。

序列级重要性比率。在标准 GRPO 中，重要性权重是在词元级计算的，反映每一步的似然。当将 GRPO 适配到决策变换器时，我们发现，在序列级计算重要性权重——即针对长度为 L_{traj} 的完整子轨迹进行计算——能够显著提升模型性能。直观而言，序列级重要性比率 $\pi_\theta(\tau_{k_i}^{\text{sub}}|s_k,\ g_k)\ \pi_{\theta_{\text{old}}}(\tau_{k_i}^{\text{sub}}|s_k,\ g_k)$ 与.....更加一致

$\widehat{}$以及优势 A_{k_i}，该优势已在序列级别计算得出。实证验证图 2c进一步支持了这一判断。

我们修改了原始 GRPO 目标函数，以纳入序列级重要性加权。由于子轨迹缓冲区 $\mathcal{T}_{\text{sub}}$ 存储子轨迹 $\tau_{k_i}^{\text{sub}}$及其优势 $\widehat{A}_{k_i}$ 和重置点（s_k，g_k），我们从子轨迹缓冲区中随机采样子轨迹，并最大化以下目标函数：

$$J_{\text{GRPO}}(\theta) = \mathbb{E}_{(\tau_{k_i}^{\text{sub}},\widehat{A}_{k_i},(s_k,g_k))\sim\text{unif}(\mathcal{T}_{\text{sub}})} \min\left(\frac{\pi_\theta(\tau_{k_i}^{\text{sub}}\mid s_k,g_k)}{\pi_{\theta_{\text{old}}}(\tau_{k_i}^{\text{sub}}\mid s_k,g_k)}\widehat{A}_{k_i},\ \text{clip}\left(\frac{\pi_\theta(\tau_{k_i}^{\text{sub}}\mid s_k,g_k)}{\pi_{\theta_{\text{old}}}(\tau_{k_i}^{\text{sub}}\mid s_k,g_k)},1-\varepsilon,1+\varepsilon\right)\widehat{A}_{k_i}\right) - \beta D_{\text{KL}}(\pi_\theta\,\|\,\pi_{\theta_{\text{ref}}}). \quad (3)$$

这一序列级目标函数带来了更稳定、样本效率更高的优化效果，与此同时，Zheng 等人(2025年在语言建模领域的研究结果一致。尽管Zheng 等人(2025年) 提议对序列级重要性比率取几何平均数，但我们发现，在相对较短的轨迹展开长度下，将其与子轨迹优化结合时，去除几何归一化（如公式（3）所示）能够带来更优性能（图 2e）。我们推测，这种改进源于：不进行几何归一化的学习（i）通过裁剪更快地抑制过时的子轨迹；（ii）允许对近似同策略的子轨迹进行更激进的更新。

在最大化 t 的同时　GRPO目标函数在 公式（3）中，我们施加熵约束 $H(\pi_\theta(\tau_{k_i}^{\text{sub}}\mid s_k,g_k)) \geq$　　ρ

以鼓励探索。这里，$H(\pi_\theta(\tau_{k_i}^{\text{sub}} \mid s_k, g_k))$ 表示在子轨迹 $\tau_{k_i}^{\text{sub}}$ 上计算的平均策略熵，ρ 是预先设定的下界。按照 Zheng et al. (2022) 的方法，我们通过原始-对偶形式 $\min_{\kappa \geq 0} \max_\theta J(\theta) + \kappa\big(H(\pi_\theta(\tau_{k_i}^{\text{sub}} \mid s_k, g_k)) - \rho\big)$放宽这一硬约束，其中 κ 是对偶变量。在实践中，这相当于在公式（3）中增加一个熵项$\kappa H(\pi_\theta(\tau_{k_i}^{\text{sub}} \mid s_k, g_k))$，同时在训练期间自适应地更新 κ，以确保熵约束$H(\pi_\theta(\tau_{k_i}^{\text{sub}} \mid s_k, g_k)) \geq \rho$大致得到满足。

用于状态选择的主动采样。在策略展开期间，我们观察到，某些状态-RTG 对（s_h，g_h）的预测动作分布 $\pi_\theta(\cdot \mid s_h, g_h)$ 方差更高。由于较高的方差意味着更大的预测不确定性，将子轨迹展开的起始点重置到这类不确定状态，可以促进更有效的探索并加快学习。受这一观察启发，我们引入了一种主动采样机制，使子轨迹选择偏向不确定性较高的状态。

具体而言，对于每条轨迹 τ，我们通过汇总动作分布的对角协方差，计算每个步骤 h 的标量不确定性分数。给定 $\pi_\theta(\cdot \mid s_h, g_h) = \mathcal{N}(\mu_\theta, \Sigma_\theta)$，其中 $\Sigma_\theta = \text{diag}(\sigma_h^2, 1, \ldots, \sigma_{h,d}^2)$。我们定义 $\sigma_h^2 := \frac{1}{d}\sum_{i=1}^{d} \sigma_{h,i}^2$，并应用 softmax 变换以获得采样分布 $p_h := \exp(\sigma_h^2) / \sum_{h'=1}^{|\tau|} \exp(\sigma_{h'}^2)$。随后根据 p 对子轨迹重置点进行采样。这种主动选择策略优先在不确定性较高的区域进行学习更新，从而实现更快且更稳定的收敛（图 2d）。

3.3 扩展

3.3.1 Q引导的GRPO-DT

本节介绍*Q*引导的*GRPO*-决策变换器，这是算法1 的一个变体，专为无法重置环境的情形设计。Q引导的 GRPO-决策变换器基本遵循算法1中的伪代码，但第6行的重置操作除外。我们不再从某个重置点显式地展开子轨迹（s_h，g_h），而是训练一个辅助Q函数 Q_{ϕ_t}，以评估从当前策略中采样的动作 $\pi_{\theta_t}(\cdot \mid s_h, g_h)$，用于优势计算。

具体而言，在每个状态-回报目标对（s_h, g_h）下，我们随机采样一组 G 个候选动作 $\{a_{h_i}\}_{i=1}^{G} \sim \pi_{\theta_t}(\cdot \mid s_h, g_h)$。每个采样动作都被视为定义了一个假设子轨迹：该子轨迹从（s_h, g_h）开始，先执行 a_{h_i}，然后遵循策略 π_{θ_t}。根据这一解释，我们使用Q函数评估每个动作的质量，并得到一个标量奖励 $R(a_{h_i}) = Q_{\phi_t}(s_h, a_{h_i})$。随后，我们按照公式（2）使用这些奖励计算归一化优势。Q函数 Q_{ϕ_t} 按照TD3算法进行训练（Fujimoto 等，2018）。

3.3.2 将 PPO 改编为决策变换器

除了将 GRPO 适配到决策变换器之外，我们还将近端策略优化（PPO，Schulman 等，2017）扩展到决策变换器框架，并将其记为 PPO-DT。与直接将 PPO 应用于决策变换器的先前尝试（Yan 等人，2024）不同，我们的方法在在线微调过程中移除了事后回报重标记步骤，如第3.1节所述。这一调整确保了轨迹展开目标与训练目标保持一致，这对于在重要性采样下实现稳定优化至关重要。我们的公式也不同于多智能体强化学习中的 PPO 适配方法；后者完全省略了剩余回报（RTG）条件（Meng等，2023），实际上将决策变换器简化为行为克隆（Ross和Bagnell，2010年；Hussein 等人，2017年）。遵循 Zheng et al. (2022)，我们在从完整轨迹中采样的子轨迹上进行优化，并引入自适应熵项以鼓励探索。由于 PPO 在词元级计算优势值，PPO-DT 相应地采用词元级重要性比率，而非我们的 GRPO-DT 适配方法所使用的序列级比率。训练价值网络时，我们采用广义优势估计（GAE，Schulman 等，2015），将蒙特卡罗估计与时序差分自举相结合，从而使价值函数更新更加平滑、稳定。PPO-DT 的更多实现细节见附录B.2。

4 实验

本节中，我们通过实证评估使用纯强化学习梯度对决策变换器进行在线微调的算法。我们在 第4.1节介绍实验设置，在 第4.2节呈现主要结果，并在 第 4.3 节提供额外分析和消融实验。

4.1 实验设置

环境和数据集。我们在 D4RL（Fu等人，2020）中的三种连续控制和操作环境上评估各方法：(i) MuJoCo（Todorov等人，2012）任务，包括 Hopper、Walker2d 和蚂蚁，具有稠密奖励，并在中等、中等回放和随机数据集上进行评估。(ii) Adroit操作任务（Rajeswaran等人，2017），包括门、锤子和 Pen，在人类和克隆数据集上进行评估。(iii) AntMaze（Fu等人，2020），具有稀疏目标到达奖励（成功时奖励为 1，否则为 0），使用 umaze 和 umaze-diverse 数据集。附录 A 中提供了各环境和数据集的详细说明。

基线。 我们将改编后的 GRPO-决策变换器和 PPO-决策变换器与三种主要基线进行比较：(i) **在线决策变换器**（**ODT**；Chen 等人 (2021年），这是决策变换器的标准在线扩展方法，以监督损失作为微调目标；(ii)**ODT+TD3** (Yan 等人，2024年），这是当前决策变换器在线微调的最先进方法；以及 (iii) **IQL** (Kostrikov等人，2021年），这是一种广泛使用的离线强化学习算法，我们在实验中采用其在线变体。作为参考，我们还报告了未经在线微调的离线预训练**决策变换器**(**决策变换器**）的性能。超参数的详细设置见附录B.1。

指标。 按照 D4RL（Fu等人，2020年），我们报告每种算法的归一化最终奖励，数值越高表示性能越好。所有结果均为三次运行的平均值，并报告标准差。此外，我们还展示了跟踪训练过程中归一化奖励变化的学习曲线。

对于学习曲线，x 轴表示外层学习迭代的次数，即 1中的第 算法1行，以及 Zheng 等人 （2022年）中算法 1的第3行。此选择是在统一不同类别的方法时所作的折中。具体而言，IQL、ODT 和 ODT+TD3 所需的梯度更新次数（因而计算量）几乎比我们的方法多两个数量级；我们的方法包括 GRPO-DT 和 PPO-DT，而我们改编的基于策略梯度的算法所需的环境交互次数则多出数倍至数十倍。这种差异源于 IQL、ODT 和 ODT+TD3 得益于大量经验回放，而策略梯度方法通常需要更多在线交互。

为公平比较渐近性能，我们对所有算法进行的迭代次数多于Yan 等人（2024年）所报告的次数。每次迭代完成梯度更新后进行评估。因此，即使在迭代次数为 0 时，所有方法也已经完成了若干次更新，在此期间它们的行为可能出现分歧并产生不同结果。

4.2 主要结果

表1报告了每种方法在三个随机种子下取平均得到的归一化回报和标准差。总体而言，我们提出的方法 GRPO-决策变换器在大多数任务中取得了最佳性能，创下新的最佳水平。PPO-决策变换器在若干情况下表现出色，但在某些环境中（例如 D-C-v1）表现不佳。ODT+TD3取得了合理的性能，但总体上不及 GRPO-决策变换器。ODT 和 IQL 在大多数基准测试中表现不佳，尤其是在预训练数据质量较低的任务上，例如随机数据集，以及 Adroit 等具有挑战性的环境中。值得注意的是，我们对 ODT+TD3 的实现采用了更长的训练迭代次数（如第4.1节所述），因此取得了优于Yan 等人（2024年）最初报告结果的表现。

使用随机离线数据进行学习。表1的第一部分评估在开始在线探索之前使用随机离线数据集（Fu等人，2020年）进行预训练时的模型性能。这些离线数据集由未经训练的策略收集的轨迹组成，因此几乎不包含有意义的监督信号，

表1：各方法在不同环境和数据集上的平均归一化最终回报比较。最佳结果以**粗体**显示，与最佳结果相差不超过 10% 的结果带下划线。环境和任务缩写如下：Ho = Hopper，Wa = Walker2d，An = 蚂蚁，D = 门，P = Pen，H = 锤子，U = U形迷宫，UD = 多样化U形迷宫；数据集类型：R = 随机，M = 中等，MR =中等回放，C = 克隆，H = 人类。每个条目均以“性能 ±标准差”的形式报告。

环境	数据集	DT	IQL	ODT	ODT+TD3	PPO-决策变换器	GRPO-决策变换器
MuJoCo（随机）	Ho-R-v2	2.13	41.02 ± 13.35	30.43 ± 0.01	83.32 ± 8.46	**106.97 ± 0.96**	99.20 ± 3.80
	Wa-R-v2	4.53	22.75 ± 1.56	10.88 ± 0.34	82.95 ± 18.28	**108.69 ± 8.86**	100.25 ± 33.19
	An-R-v2	31.41	58.69 ± 23.03	19.08 ± 3.97	80.58 ± 7.25	107.45 ± 22.83	**120.69 ± 5.47**
	平均值	12.69	40.82	20.13	82.28	**107.70**	**106.71**
MuJoCo (中等)	Ho-M-v2	46.46	74.19 ± 20.25	98.02 ± 0.63	101.47 ± 2.29	105.65 ± 5.43	**108.81 ± 0.85**
	Ho-MR-v2	38.12	96.97 ± 2.16	87.73 ± 0.59	107.94 ± 2.29	**109.60 ± 1.63**	83.61 ± 20.75
	Wa-M-v2	47.95	103.45 ± 1.37	76.49 ± 0.78	103.27 ± 5.95	109.49 ± 9.04	**158.34 ± 3.75**
	Wa-MR-v2	56.24	103.00 ± 2.65	74.21 ± 2.41	102.80 ± 2.68	117.45 ± 14.79	**137.36 ± 5.64**
	An-M-v2	86.22	118.18 ± 2.42	90.71 ± 0.03	131.56 ± 0.41	139.84 ± 0.95	**147.51 ± 2.44**
	An-MR-v2	84.30	117.51 ± 0.82	83.63 ± 0.87	120.01 ± 2.94	117.95 ± 2.54	**142.05 ± 3.32**
	平均值	59.88	102.21	85.13	111.18	116.66	**129.61**
Adroit	D-C-v1	0.14	46.72 ± 0.30	1.26 ± 1.02	79.98 ± 5.62	0.19 ± 0.00	**96.41 ± 7.59**
	D-H-v1	3.28	11.27 ± 0.44	8.76 ± 3.87	79.73 ± 4.37	**94.12 ± 3.99**	89.33 ± 10.12
	P-C-v1	43.31	62.11 ± 13.25	16.24 ± 5.12	109.86 ± 6.27	27.14 ± 0.24	**111.15 ± 2.61**
	P-H-v1	33.10	24.94 ± 1.48	19.84 ± 7.42	77.18 ± 7.42	9.92 ± 5.00	**85.11 ± 6.08**
	H-C-v1	0.65	4.87 ± 3.10	1.32 ± 0.06	119.95 ± 2.45	130.60 ± 2.81	**140.45 ± 1.93**
	H-H-v1	1.08	1.04 ± 1.56	0.91 ± 0.22	120.93 ± 2.18	129.23 ± 2.18	**132.64 ± 12.56**
	平均值	13.59	25.15	8.06	97.93	65.2	**109.18**
AntMaze	U-v2	50.00	91.21 ± 2.14	89.27 ± 3.73	**99.64 ± 0.20**	0.00 ± 0.00	96.07 ± 0.53
	UD-v2	65.00	0.00 ± 0.00	63.81 ± 1.64	**99.42 ± 0.43**	47.00 ± 4.00	97.70 ± 2.67
	平均值	57.5	45.61	76.54	**99.53**	23.50	96.89

可作为离线到在线适应以及模型对较差预训练质量的鲁棒性的标准压力测试（Yan 等人，2024年）。

在这一具有挑战性的设置下，主要依赖监督目标的方法在在线微调期间难以取得进展：ODT 未能取得实质性进展。IQL 虽然利用了时序差分学习，却受到价值估计不准确和策略改进有限的影响。ODT+TD3 通过在监督目标中加入强化学习梯度，取得了中等水平的表现，但仍明显弱于完全依赖强化学习梯度的 GRPO-DT 和 PPO-DT。通过直接优化奖励驱动的强化学习信号，我们的方法有效地弥补了糟糕的初始化，并在这一低质量预训练阶段取得了最佳的整体表现。

使用中等质量的离线数据进行训练。表1表1的最后三个数据块报告了在在线探索之前使用中等质量预训练数据获得的结果，这代表了现实中的离线到在线适应场景。在**MuJoCo** 任务中，我们改进后的方法 GRPO-DT 和 PPO-DT 取得了最高的整体回报。ODT+TD3 和 IQL 仍具有竞争力，而 ODT 的表现尚可，但逊于前两者。

在更具挑战性的**Adroit** 环境中，状态和动作空间的维度都相对较高，策略在微调过程中容易退化或崩溃。在这些条件下，ODT 和 IQL 未能超越预训练性能，而我们改进后的 GRPO-DT 始终表现出色，在 6 个数据集中的 5 个上取得最佳表现。ODT+TD3 在部分 Adroit 任务上仍具竞争力，但在所有数据集上都不及我们的 GRPO-DT。PPO-DT 在某些任务上表现出色，但在其他任务上的提升有限。

最后，在具有稀疏目标到达奖励的**AntMaze** 环境中（成功时奖励为 1，否则为 0），ODT+TD3 表现最佳，而我们的 GRPO-DT 紧随其后，回报仅比最佳分数低 4%。在这种稀疏奖励设置下，所有其他方法都未能取得实质性进展。

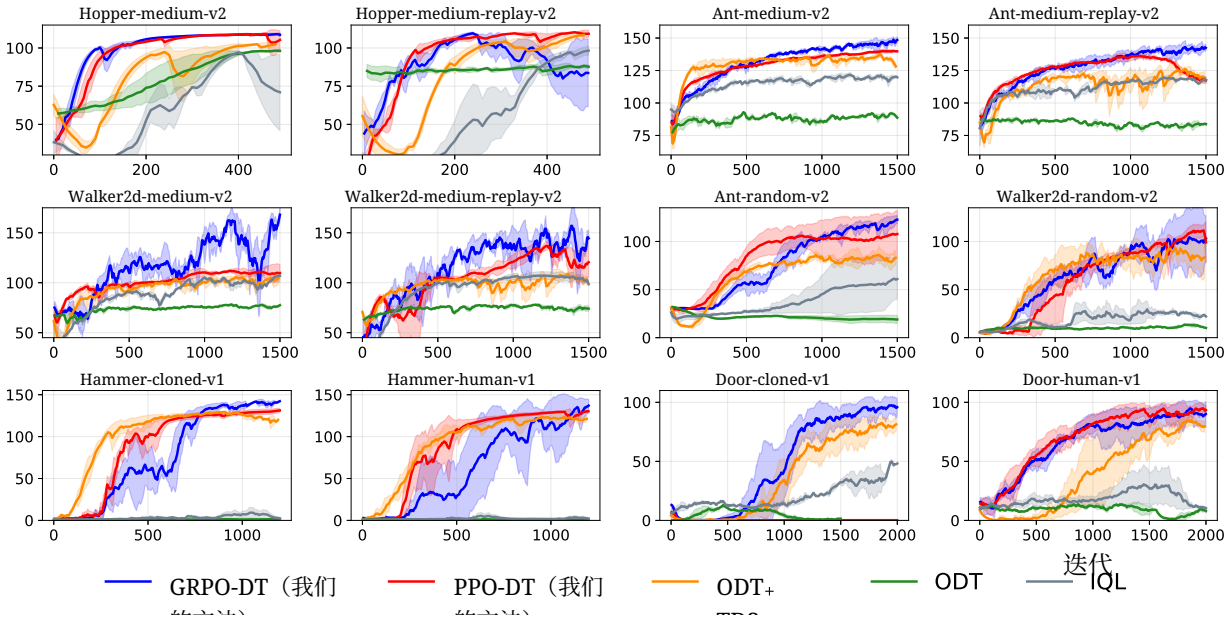


图3：不同强化学习环境中的性能对比。我们提出的 GRPO-DT 方法在大多数任务中取得了最佳性能。PPO-DT 在大多数情况下表现出色，但在某些环境中未能奏效。ODT+TD3 整体表现尚可，但通常不及 GRPO-DT。ODT 和 IQL 在大多数环境中的表现始终不佳。

相较于 ODT 的计算与实践优势+TD3。除了取得更优的性能（如表1 和图 3所示）外，我们改编的 GRPO-DT 和 PPO-DT 相较于 ODT 也具备显著的计算与实践优势+TD3。

(i) 更少的梯度更新。 子轨迹展开能够实现更细粒度的信用分配和更准确的梯度估计，使我们的方法仅需少量更新就能有效学习。例如，我们的方法每次迭代仅执行 8×256 次梯度更新，而 ODT+TD3（以及 ODT）则需要约 256×300次更新，计算成本因此大幅降低。

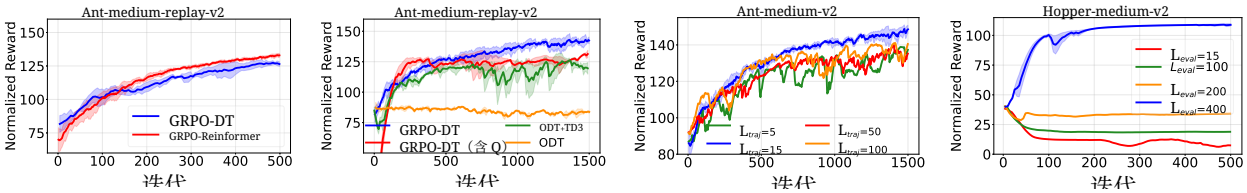
(ii) **与预训练的DT风格模型无缝兼容**。 我们的方法无需大幅修改，即可直接微调任何预训练的DT风格模型（其他模型的实验见 第 4.3 节 ），而 ODT+TD3 则需要 修改离线预训练目标，以整合强化学习梯度并联合训练辅助Q函数——这会显著降低其灵活性和可扩展性。

(iii) 实现更简单，稳定性更高。 与依赖辅助评论家网络的 ODT+TD3 不同，我们的方法无需引入额外网络。这种无评论家设计简化了实现，提高了训练稳定性，并有助于在不同环境和预训练模型中实现可复现性。

4.3 补充分析与消融实验

本节提供补充分析和消融研究。支持 算法1 中关键设计选择的实证证据见 图1 和 图2，其中包括移除事后回报重标记、使用子轨迹展开、使用一致状态进行轨迹展开、采用序列级重要性比率、采用主动采样进行状态选择，以及移除序列级重要性比率的几何平均归一化所产生的影响。

超越决策变换器的在线微调。为评估我们的 GRPO-DT 在标准决策变换器之外的适用性，我们在*Reinformer* 模型（Zhuang 等人，2024年）上对其进行评估。Reinformer 将



(a) Reinformer 上的 GRPO　(b) Q引导的GRPO-DT　(c) 关于 L_{traj} 的消融实验　(d) 关于 L_{eval} 的消融实验

图4：补充分析与消融实验。(a) GRPO-DT在不同决策变换器模型上的表现，包括标准DT和Reinformer，表明GRPO-DT不仅适用于标准DT。(b) 在无法进行环境重置时，对比基于重置的GRPO-DT与Q引导的GRPO-DT；即使不进行重置，Q引导的GRPO-DT仍能保持优异表现。(c) 子轨迹长度消融实验 L_{traj}：子轨迹过短或过长都会降低性能，凸显了选择合适子轨迹长度的重要性。(d) 子轨迹评估长度消融实验 L_{eval}：评估步数不足会导致训练不稳定或模型崩溃，而较大的 L_{eval} 则能带来更可靠的性能。

回报最大化融入监督式序列建模，并利用期望分位数回归预测更高的回报目标，进而指导动作选择。如图 4a所示，我们的 GRPO-DT 在 Reinformer 上也表现出色，这表明我们提出的 GRPO-DT 能够泛化至标准决策变换器之外。

Q引导的GRPO-DT无需重置。在无法进行环境重置的场景中，我们使用TD3（Fujimoto 等人，2018年）训练辅助Q函数，并使用Q引导的优势值应用GRPO-DT（详见第3.3.1节）。此变体还降低了样本复杂度，因为不再需要生成和评估子轨迹（算法1中的第16行）。如图4b所示，即使不进行环境重置，Q引导版本的GRPO-DT仍能保持优异表现。

子轨迹长度的消融实验。 在我们的方法中，每条子轨迹都是优势计算和信用分配的基本单位，因此其长度是 L_{traj} 一个关键的超参数。 图 4c 中的实证结果表明，增加子轨迹长度会使训练不稳定并降低性能。相反，过短的子轨迹也可能导致次优结果。这可能是因为从相似状态分布中采样的短片段过于同质化，所提供的多样性有限，且为强化学习微调提供的学习信号较弱。

子轨迹评估长度的消融实验。 对于每条子轨迹，我们将轨迹展开延长 L_{eval}额外的评估步数以计算奖励。在我们的实验中， L_{eval} 根据环境的不同，其范围为 30 到 400（见表 6）。我们对评估长度 L_{eval} 进行消融实验，以研究其影响。如图 4d所示，增加 L_{eval} 能够更可靠地估计子轨迹质量，从而持续提升模型性能。

5 相关工作

用于强化学习的 Transformer。随着变换器架构（Vaswani 等人，2017年）成为深度学习中的主导范式——尤其是在语言领域（Radford 等人，2018年；Brown 等人，2020年）和视觉领域（Dosovitskiy 等人，2021年）——越来越多的研究开始探索基于 Transformer 的强化学习方法。在这一范式中，以决策变换器（Chen 等人，2021年）为代表，强化学习被表述为一个序列建模问题：模型以过去的状态、动作和待实现回报为条件，自回归地预测未来动作。借助 Transformer 强大的序列建模能力，决策变换器风格的方法在一系列基准测试中展现出有竞争力或最先进的性能（Janner 等人，2021年；Wang 等人，2022年；Yamagata 等人，2023年；Zhuang等人，2024年）。这些方法最初是为离线强化学习开发的，近年来已有多项研究将其拓展至在线环境——这通常称为离线到在线强化学习（Lee 等人，2022年；Yu 和 Zhang，2023年；Nakamoto 等人，2023年）。然而，现有的在线

决策变换器的微调方法要么完全依赖监督式序列建模目标（Zheng 等人，2022年；Xie 等人，2023年），要么继续优先考虑监督损失，只赋予强化学习梯度较小的权重（Yan 等人，2024年）。相比之下，我们的工作通过提出首个使用纯强化学习梯度对决策变换器进行在线微调的方法，补充了这一研究方向，从而无需依赖监督目标即可进行有原则的策略优化。

用于 Transformer 的强化学习。强化学习也已成为对齐和增强大语言模型（Ouyang 等人，2022年；Bai 等人，2022年；Lee 等人，2023年）以及多模态模型（Liu 等人，2025b；Shen 等人，2025年）的有效方法。人们探索了范围广泛的算法，从 PPO 及其变体等在线策略梯度方法（Ouyang 等人，2022年；Luong 等人，2024年；Kazemnejad 等人，2025年），到基于 IQL 及其变体的离线强化学习方法（Snell 等人，2022年；Qi 等人，2024年），以及 DPO 及其扩展等基于偏好的策略优化方法（Rafailov 等人，2023年；Ethayarajh 等人，2024年）。近年来，GRPO 及其变体受到广泛关注（Shao 等，2024；Guo 等，2025；Liu 等人，2025a；Yu 等人，2025年），并已成功应用于大语言模型和多模态模型，以提升推理性能。在本研究中，我们将 GRPO 和 PPO 扩展至经典强化学习环境中决策变换器的在线微调。此外，我们还提出了 GRPO 的一种子轨迹变体，相较于标准 GRPO，它能显著改善信用分配；这一变体或许也适用于本文所考察设定之外的场景，具有独立研究价值。

其他相关工作。 在撰写本文期间，我们注意到一项关于序列级重要性比率的同期研究，作者为 Zheng 等人 (2025年)，该研究于2025年7月在线发表。尽管我们的方法共享相同的核心思想，但我们的序列级重要性比率技术是独立开发的；我们工作的初始版本于2025年9月公开发布。我们的方法在一项关键设计选择上不同于 Zheng 等人 (2025年)。具体而言，Zheng 等人 (2025年) 提议对序列级重要性比率取几何平均数，而我们在进行子轨迹级优化时不进行这种几何归一化。实验上，我们发现，结合相对较短的轨迹展开长度时，这一改动能够带来更优的性能（图 2e）。我们推测，这种提升源于不进行几何归一化的学习方式能够：（i）通过裁剪更快地抑制过时的子轨迹；（ii）对近似同策略的子轨迹进行更激进的更新。

我们的主动状态采样技术受到经典主动学习（Settles，2009）的启发。在经典主动学习中，基于不确定性的方法已知能够将样本复杂度相较于标准监督学习呈指数级降低（Hanneke，2014；Puchkin 和 Zhivotovskiy，2021；Zhu 和 Nowak，2022a,b），并且在深度神经网络上的实证表现出色（Ash 等，2020；Saran 等，2023；Zhang 等，2024）。近年来，主动学习的核心原则已应用于强化学习（Yin 等，2023）、大语言模型（Bhatt 等，2024；Wang 等，2025）和多模态模型（Zhang 和 Zhu，2025）。

6 结论

在本研究中，我们系统地研究了使用纯强化学习梯度在线微调决策变换器。我们发现，事后回报重标记是阻碍基于重要性采样的策略梯度方法有效应用于在线决策变换器微调的关键障碍。基于这一发现，我们提出了一种有原则的解决方案：将 GRPO 适配到决策变换器框架中。我们的方法结合了子轨迹优化以改进信用分配、序列级重要性比率以提升稳定性和效率，以及主动状态采样以加强探索，从而实现对预训练决策变换器进行纯强化学习在线微调。我们还进一步证明，PPO 也可以适配于在线决策变换器微调。大量实验表明，我们的方法优于现有在线决策变换器基线，并在多个基准测试中取得了新的最先进性能。总体而言，我们的发现凸显了纯强化学习优化用于决策变换器的可行性和有效性，并为通过强化学习驱动的微调来扩展基于决策变换器的智能体开辟了新途径。

参考文献

Josh Achiam、Steven Adler、Sandhini Agarwal、Lama Ahmad、Ilge Akkaya、Florencia Leoni Aleman、Diogo Almeida、Janko Altenschmidt、Sam Altman、Shyamal Anadkat 等。GPT-4 技术报告。*arXiv* 预印本 *arXiv:2303.08774*，2023年。Marcin Andrychowicz、Filip Wolski、Alex Ray、Jonas Schneider、Rachel Fong、Peter Welinder、Bob McGrew、Josh Tobin、OpenAI、Pieter Abbeel 和 Wojciech Zaremba。后见经验回放。神经信息处理系统进展，第30卷，2017年。Jordan T. Ash、Chicheng Zhang、Akshay Krishnamurthy、John Langford 和 Alekh Agarwal。通过多样化不确定梯度下界进行深度批量主动学习。载于国际学习表征会议，2020年。

Jimmy Lei Ba、Jamie Ryan Kiros 和 Geoffrey EHinton。层归一化。*arXiv* 预印本 *arXiv:1607.06450*，2016年。

Yuntao Bai、Saurav Kadavath、Sandipan Kundu、Amanda Askell、Jackson Kernion、Andy Jones、Anna Chen、Anna Goldie、Azalia Mirhoseini、Cameron McKinnon 等。宪法式人工智能：通过人工智能反馈实现无害性。*arXiv* 预印本 *arXiv:2212.08073*，2022年。

Gantavya Bhatt、Yifang Chen、Arnav M Das、Jifan Zhang、Sang T Truong、Stephen Mussmann、Yinglun Zhu、Jeffrey Bilmes、Simon S Du、Kevin Jamieson 等。用于大语言模型标签高效监督微调的实验设计框架。计算语言学协会论文集，2024年。

Tom Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared D Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell 等。语言模型是少样本学习器。神经信息处理系统进展，33:1877–1901，2020年。

Lili Chen、Kevin Lu、Aravind Rajeswaran、Kimin Lee、Aditya Grover、Misha Laskin、Pieter Abbeel、Aravind Srinivas 和 Igor Mordatch。决策变换器：通过序列建模实现强化学习。神经信息处理系统进展，34：15084–15097，2021年。

Gheorghe Comanici、Eric Bieber、Mike Schaekermann、Ice Pasupat、Noveen Sachdeva、Inderjit Dhillon、Marcel Blistein、Ori Ram、Dan Zhang、Evan Rosen 等。Gemini 2.5：凭借高级推理、多模态、长上下文和新一代智能体能力拓展前沿。*arXiv* 预印本 *arXiv:2507.06261*，2025年。

Alexey Dosovitskiy、Lucas Beyer、Alexander Kolesnikov、Dirk Weissenborn、Xiaohua Zhai、Thomas Unterthiner、Mostafa Dehghani、Matthias Minderer、Georg Heigold、Sylvain Gelly、Jakob Uszkoreit 和 Neil Houlsby。一张图像胜过 16×16 个词：用于大规模图像识别的 Transformer。发表于国际学习表征会议，2021年。

Stefan Elfwing、Eiji Uchibe 和 Kenji Doya。用于强化学习中神经网络函数逼近的 Sigmoid加权线性单元。神经网络，107：3–11，2018年。

Kawin Ethayarajh、Winnie Xu、Niklas Muennighoff、Dan Jurafsky 和 Douwe Kiela。KTO：将模型对齐视为前景理论优化。*arXiv* 预印本 *arXiv:2402.01306*，2024年。

Justin Fu、Aviral Kumar、Ofir Nachum、George Tucker 和 Sergey Levine。D4RL：深度数据驱动强化学习数据集。*arXiv* 预印本 *arXiv:2004.07219*，2020年。

Scott Fujimoto 和 Shixiang Shane Gu。离线强化学习的极简方法。神经信息处理系统进展，34：20132–20145，2021年。

Scott Fujimoto、Herke Hoof 和 David Meger。《解决演员-评论家方法中的函数逼近误差》。载于国际机器学习会议，第 1587–1596 页。PMLR，2018年。

Dibya Ghosh、Abhishek Gupta、Ashwin Reddy、Justin Fu、Coline Devin、Benjamin Eysenbach 和 Sergey Levine。《通过迭代监督学习实现目标》。arXiv 预印本 arXiv:1912.06088，2019年。Daya Guo、Dejian Yang、Haowei Zhang、Junxiao Song、Ruoyu Zhang、Runxin Xu、Qihao Zhu、Shirong Ma、Peiyi Wang、Xiao Bi 等。《DeepSeek-R1：通过强化学习激发大语言模型的推理能力》。arXiv 预印本 arXiv:2501.12948，2025年。Steve Hanneke。《主动学习理论》。Foundations and Trends in Machine Learning，7(2-3)，2014年。Shengyi Huang、Rousslan Fernand Julien Dossa、Chang Ye、Jeff Braga、Dipam Chakraborty、Kinal Mehta 和 Jo˜ao G.M. Ara´ujo。《CleanRL：高质量的深度强化学习算法单文件实现》。Journal of Machine Learning Research，23(274)：1–18，2022年。网址：http://jmlr.org/papers/ v23/21-1342.html。Ahmed Hussein、Mohamed Medhat Gaber、Eyad Elyan 和 Chrisina Jayne。《模仿学习：学习方法综述》。ACM Computing Surveys (CSUR)，50(2)：1–35，2017年。Michael Janner、Qiyang Li 和 Sergey Levine。《将离线强化学习视为一个大型序列建模问题》。神经信息处理系统进展，34：1273–1286，2021年。Amirhossein Kazemnejad、Milad Aghajohari、Eva Portelance、Alessandro Sordoni、Siva Reddy、Aaron Courville 和 Nicolas Le Roux。《VinePPO：改进大语言模型强化学习训练中的信用分配》。载于第四十二届国际机器学习会议，2025年。Ilya Kostrikov、Ashvin Nair 和 Sergey Levine。《基于隐式Q学习的离线强化学习》。arXiv 预印本 arXiv:2110.06169，2021年。Harrison Lee、Samrat Phatale、Hassan Mansoor、Thomas Mesnard、Johan Ferret、Kellie Lu、Colton Bishop、Ethan Hall、Victor Carbune、Abhinav Rastogi 等。《RLAIF 与 RLHF：利用人工智能反馈扩展基于人类反馈的强化学习》。arXiv 预印本 arXiv:2309.00267，2023年。Seunghyun Lee、Younggyo Seo、Kimin Lee、Pieter Abbeel 和 Jinwoo Shin。《通过平衡回放和悲观Q集成实现离线到在线强化学习》。载于机器人学习会议，第 1702–1712 页。PMLR，2022年。Zichen Liu、Changyu Chen、Wenjun Li、Penghui Qi、Tianyu Pang、Chao Du、Wee Sun Lee 和 Min Lin。《理解类R1-Zero训练：批判性视角》。arXiv 预印本 arXiv:2503.20783，2025a。Ziyu Liu、Zeyi Sun、Yuhang Zang、Xiaoyi Dong、Yuhang Cao、Haodong Duan、Dahua Lin 和 Jiaqi Wang。《Visual-RFT：视觉强化微调》。arXiv 预印本 arXiv:2503.01785，2025b。Ilya Loshchilov 和 Frank Hutter。《解耦权重衰减正则化》。arXiv 预印本 arXiv:1711.05101，2017 年。Trung Quoc Luong 、 Xinbo Zhang 、 Zhanming Jie 、 Peng Sun 、 Xiaoran Jin 和 Hang Li。《ReFT：通过强化微调进行推理》。arXiv 预印本 arXiv:2401.08967，2024年。Linghui Meng、Muning Wen、Chenyang Le、Xiyun Li、Dengpeng Xing、Weinan Zhang、Ying Wen、Haifeng Zhang、Jun Wang、Yaodong Yang 等。《离线预训练多智能体决策变换器》。Machine Intelligence Research，20(2)：233–248，2023年。Zak Mhammedi、Dylan J Foster 和 Alexander Rakhlin。《重置在在线强化学习中的作用》。神经信息处理系统进展，37：12334–12407，2024年。Mitsuhiko Nakamoto、Simon Zhai、Anikait Singh、Max Sobol Mark、Yi Ma、Chelsea Finn、Aviral Kumar 和 Sergey Levine。《Cal-QL：用于高效在线微调的校准离线强化学习预训练》。神经信息处理系统进展，36：62244–62269，2023年。Long Ouyang、Jeffrey Wu、Xu Jiang、Diogo Almeida、Carroll Wainwright、Pamela Mishkin、Chong Zhang、Sandhini Agarwal、Katarina Slama、Alex Ray 等。《训练语言模型遵循人类反馈指令》。神经信息处理系统进展，35：27730–27744，2022年。

Nikita Puchkin 和 Nikita Zhivotovskiy。通过弃答实现非特定主动学习的指数级节省。载于学习理论会议，第 3806–3832 页。PMLR，2021年。Jianing Qi、Hao Tang 和 Zhigang Zhu。Verifierq：利用基于 Q-learning 的验证器增强大语言模型测试时计算。*arXiv* 预印本 *arXiv:2410.08048*，2024年。Alec Radford、Karthik Narasimhan、Tim Salimans、Ilya Sutskever 等。通过生成式预训练改进语言理解。2018年。Rafael Rafailov、Archit Sharma、Eric Mitchell、Christopher D Manning、Stefano Ermon 和 Chelsea Finn。直接偏好优化：你的语言模型其实是一个奖励模型。神经信息处理系统进展，36：53728–53741，2023年。Aravind Rajeswaran、Vikash Kumar、Abhishek Gupta、Giulia Vezzani、John Schulman、Emanuel Todorov 和 Sergey Levine。通过深度强化学习和示范学习复杂的灵巧操作。*arXiv* 预印本 *arXiv:1709.10087*，2017年。St´ephane Ross 和 Drew Bagnell。用于模仿学习的高效归约。载于第十三届国际人工智能与统计学会议论文集，第 661–668 页。JMLR研讨会与会议论文集，2010年。

Akanksha Saran、Safoora Yousefi、Akshay Krishnamurthy、John Langford 和 Jordan T Ash。《使用深度神经网络的流式主动学习》。载于国际机器学习会议，第 30005–30021 页。PMLR，2023年。

John Schulman、Philipp Moritz、Sergey Levine、Michael Jordan 和 Pieter Abbeel。《使用广义优势估计进行高维连续控制》。*arXiv* 预印本 *arXiv:1506.02438*，2015年。

John Schulman、Filip Wolski、Prafulla Dhariwal、Alec Radford 和 Oleg Klimov。《近端策略优化算法》，2017年。

Burr Settles。《主动学习文献综述》，2009年。

Zhihong Shao、Peiyi Wang、Qihao Zhu、Runxin Xu、Junxiao Song、Xiao Bi、Haowei Zhang、Mingchuan Zhang、YK Li、Yang Wu 等。《Deepseekmath：突破开放式语言模型数学推理能力的极限》。*arXiv* 预印本 *arXiv:2402.03300*，2024年。

Haozhan Shen、Peng Liu、Jingcheng Li、Chunxin Fang、Yibo Ma、Jiajia Liao、Qiaoli Shen、Zilun Zhang、Kangjia Zhao、Qianqian Zhang等。VLM-R1：一种稳定且可泛化的R1风格大型视觉语言模型。*arXiv*预印本 *arXiv:2504.07615*，2025年。

Charlie Snell、Ilya Kostrikov、Yi Su、Mengjiao Yang和Sergey Levine。利用隐式语言Q学习实现自然语言生成的离线强化学习。*arXiv*预印本 *arXiv:2206.11871*，2022年。

Qwen团队。QwQ-32B：拥抱强化学习的力量，2025年。

Emanuel Todorov、Tom Erez和Yuval Tassa。MuJoCo：一种用于基于模型的控制的物理引擎。载于*2012*年*IEEE/RSJ*国际智能机器人与系统会议，第5026–5033页。IEEE，2012年。

Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、 ˙ Lukasz Kaiser 和 Illia Polosukhin。《Attention is all you need》。神经信息处理系统进展，30，2017年。

Kerong Wang、Hanye Zhao、Xufang Luo、Kan Ren、Weinan Zhang 和 Dongsheng Li。《用于离线强化学习的自举 Transformer》。神经信息处理系统进展，35：34748–34761，2022年。

Shenzhi Wang、Le Yu、Chang Gao、Chujie Zheng、Shixuan Liu、Rui Lu、Kai Dang、Xionghui Chen、Jianxin Yang、Zhenru Zhang 等。超越80/20法则：高熵少数派词元推动大语言模型推理的有效强化学习。*arXiv*预印本 *arXiv:2506.01939*，2025年。

谢志辉、林子川、叶德恒、傅强、韦扬和李帅。面向决策变换器的未来条件无监督预训练。载于国际机器学习会议，第38187–38203页。PMLR，2023年。山形拓、艾哈迈德·哈利勒和劳尔·桑托斯-罗德里格斯。Q学习决策变换器：利用动态规划进行离线强化学习中的条件序列建模。载于国际机器学习会议，第38989–39007页。PMLR，2023年。严凯、亚历克斯·施温和王宇雄。强化学习梯度作为在线微调决策变换器的维生素。神经信息处理系统进展，37:38590–38628，2024年。尹东、斯里达尔·蒂亚加拉詹、内韦娜·拉齐奇、尼韦德·拉贾拉曼、郝博涛和恰巴·塞佩什瓦里。通过局部规划实现样本高效深度强化学习。*arXiv*预印本 *arXiv:2301.12579*，2023年。尤洋、李静、萨尚克·雷迪、乔纳森·许、桑吉夫·库马尔、斯里纳德·博贾纳帕利、宋晓丹、詹姆斯·德梅尔、库尔特·克伊策和谢卓睿。深度学习的大批量优化：76分钟训练BERT。*arXiv*预印本 *arXiv:1904.00962*，2019年。

余启莹、张正、朱若飞、袁宇峰、左晓晨、岳宇、戴维南、范甜甜、刘高鸿、刘灵均等。DAPO：大规模开源大语言模型强化学习系统。*arXiv*预印本 *arXiv:2503.14476*，2025年。

俞子顺和张新华。面向离线到在线强化学习的演员-评论家对齐。载于国际机器学习会议，第40452–40474页。PMLR，2023年。

张建成和Yinglun Zhu。迈向多模态主动学习：利用有限配对数据进行高效学习。*arXiv*预印本 *arXiv:2510.03247*，2025年。

Jifan Zhang、Yifang Chen、格雷戈里·卡纳尔、阿纳夫·莫汉蒂·达斯、Gantavya Bhatt、Stephen Mussmann、Yinglun Zhu、杰夫·比尔梅斯、杜少雷、Kevin Jamieson等人。Labelbench：自适应标签高效学习的综合基准测试框架。以数据为中心的机器学习研究期刊，2024年。Chujie Zheng、Shixuan Liu、李明泽、陈雄辉、于博文、Chang Gao、Kai Dang、刘玉琼、门瑞、杨安等人。组序列策略优化。*arXiv*预印本 *arXiv:2507.18071*，2025年。郑钦清、张艾米和Aditya Grover。在线决策变换器。载于国际机器学习会议，第27042–27059页。PMLR，2022年。Yinglun Zhu和罗伯特·诺瓦克。神经网络主动学习：来自非参数统计的见解。神经信息处理系统进展，35：142–155，2022a。Yinglun Zhu和罗伯特·诺瓦克。带弃权的高效主动学习。神经信息处理系统进展，35：35379–35391，2022b。庄子峰、彭登云、刘锦欣、张子琪和王东林。Reinformer：离线强化学习的最大回报序列建模。*arXiv*预印本 *arXiv:2405.08740*，2024年。

表 2： MuJoCo 数据集大小（按总步数计）及归一化最终奖励统计 .

数据集	Size	归一化奖励
Hopper-medium-v2	999 906	44.32 ± 12.27
Hopper-medium-replay-v2	402 000	14.98 ± 16.32
Hopper-random-v2	999 906	1.19 ± 1.16
Walker2d-medium-v2	999 995	62.09 ± 23.83
Walker2d-medium-replay-v2	302 000	14.84 ± 19.48
Walker2d-random-v2	999 997	0.01 ± 0.09
Ant-medium-v2	999 946	80。30 ± 35.82
Ant-medium-replay-v2	302 000	30。95 ± 31.66
Ant-random-v2	999 930	6.36 ± 10.07

A 环境与数据集详情

我们研究了D4RL（Fu等人，2020年）中的三个连续控制与操作环境：*MuJoCo*、*Adroit*和*AntMaze*。我们总共针对8项不同任务进行了实验，涵盖17个离线数据质量各异的数据集。下面将详细介绍每种环境、任务和数据集。

MuJoCo 环境。第一个环境是MuJoCo (Todorov等人，2012年），包括具有稠密奖励信号的运动任务*Hopper*、*Walker2d*和蚂蚁。我们使用中等、中等回放和随机数据集评估这些任务，其中中等数据集包含由在中等水平表现时提前停止的策略生成的轨迹，中等回放数据集包含在中等策略训练过程中采样的轨迹，而随机数据集包含由随机策略收集的轨迹。各离线数据集的大小和归一化奖励统计数据见表 2。下面将讨论各项任务。

- **Hopper**。Hopper 是一项单腿运动任务，智能体控制三个关节，使机器人向前跳跃并保持稳定性。动作空间是一个三维连续向量，对应于施加在各关节上的力矩，每个力矩都限制在 $[-1, 1]$ 范围内。观测空间有 11 个维度，包含位置和速度信息。在每个时间步，奖励由健康奖励、前进进度奖励，以及与动作平方幅度成正比的控制成本惩罚组成。

- **Walker2d.** Walker2d 是一项二维双足行走机器人任务，智能体通过控制六个关节，使机器人稳定地向前行走。动作空间是一个 6 维连续向量，对应施加在铰链关节上的力矩，每个力矩的范围均为 $[-1, 1]$。观测空间有 17 个维度，由位置和速度信息组成。在每个时间步，奖励由健康奖励、前进进度奖励，以及与动作幅度平方成正比的控制成本惩罚共同构成。

- **蚂蚁.** 蚂蚁是一项三维移动任务，智能体控制一个拥有 8 个关节的四足机器人，在保持平衡的同时向前移动。动作空间是一个 8 维连续向量，对应施加在铰链关节上的力矩，每个力矩的范围均为 $[-1, 1]$。观测空间有 105 个维度，由位置、速度和外部接触力信息组成。在每个时间步，奖励由健康奖励、前进进度奖励、与动作幅度平方成正比的控制成本惩罚，以及与接触力幅度平方成正比的外部接触力惩罚共同构成。

Adroit环境。 第二个环境是 **Adroit**，包含具有挑战性的操作任务 门、锤子和 *Pen*。我们使用 人类 和 克隆 数据集对这些任务进行评估，其中 人类 数据集包含少量人类示范，而 克隆 数据集包含从行为克隆策略和人类示范中收集的轨迹混合数据。每个离线数据集的数据集大小和归一化奖励统计信息见表3。下面将分别讨论各项任务。

表3： Adroit数据集的大小（以总步数计）及归一化最终奖励统计。

数据集	Size	归一化奖励
Pen-cloned-v1	499 886	108.63 ± 122.43
Pen-human-v1	4800	202.69 ± 154.48
Hammer-cloned-v1	999 872	8.11 ± 23.35
Hammer-human-v1	10 948	23.80 ± 33.86
Door-cloned-v1	999 939	12.29 ± 18.35
Door-human-v1	6504	28.35 ± 13.88

表4： AntMaze 数据集大小（按总步数计）及归一化最终奖励统计 .

数据集	Size	归一化奖励
Antmaze-umaze-v2	998 573	86.14 ± 34.55
Antmaze-umaze-diverse-v2	999 000	3.48 ± 18.32

• **门**。门任务涉及一个学习解锁并打开门的机器人手臂系统。动作空间是一个28维连续向量，表示关节角位置（缩放至 $[-1, 1]$）。39维观测空间包含手指关节的角位置、手掌姿态，以及门闩和门的状态信息。稠密奖励将距离、铰链对齐和速度惩罚结合起来，并根据门铰链位移给予正奖励。

• **锤子**。锤子任务要求机器人手臂系统拿起锤子并将钉子钉入木板。动作空间是一个 26 维连续向量，表示关节角位置（缩放至 $[-1, 1]$）。46 维观测空间包含手指关节的角位置、手掌姿态、锤子和钉子的姿态，以及作用于钉子的外力等信息。稠密奖励结合了距离惩罚和速度惩罚，并对举起锤子和钉入钉子给予正向奖励。

• **Pen**。钢笔任务要求机器人手臂系统操控钢笔，使其达到目标朝向。动作空间是一个 24 维连续向量，表示关节角位置（缩放至$[-1, 1]$）。45 维观测空间包含手指关节的角位置、手掌姿态，以及真实钢笔和目标的姿态等信息。稠密奖励结合了距离惩罚和朝向惩罚，并对钢笔掉落给予惩罚，同时对精确对齐和稳定控制给予奖励。

AntMaze环境。第三个环境是 **AntMaze**（Fu等人，2020年），其中包含在稀疏目标到达奖励的迷宫中进行的导航任务。我们在 *U*形迷宫 和 多样化*U*形迷宫 任务上进行评估，前者具有固定的起点和终点，后者的起点和终点则是随机的。AntMaze环境中的U形迷宫环境将一台蚂蚁四足机器人置于U形迷宫中。动作空间是一个8维连续向量，对应于施加在铰链关节上的力矩，每个力矩的范围均不超过$[-1, 1]$。观测空间是一个目标感知字典，包含一个27维的“观测”向量（蚂蚁身体部位的位置和速度）、一个2维期望目标向量（期望位置的坐标）以及一个2维已达成目标向量（当前位置的坐标）。奖励是稀疏的：蚂蚁到达目标位置时奖励值为1，否则为0。两个离线数据集的数据集大小和归一化奖励统计见表4。[1]

[1]遵循 ODT+TD3 的（Yan 等人，2024年）做法，我们从数据集中移除了所有 1步轨迹。

表 5：所有环境通用的架构和训练超参数。策略网络采用 Transformer 参数化，价值网络采用多层感知机参数化。

策略网络（Transformer）的超参数	价值
层数	4
注意力头数	4
嵌入维度	512
激活函数	SiLU（Elfwing 等人，2018年）
优化器	LAMB（You 等人，2019年）
丢弃	预训练时为 0.1，微调时禁用
梯度范数裁剪	0.5
权重衰减	1×10^{-4}
KL 系数 β	1×10^{-3}
目标熵 ρ	$-$维度($\mathcal{A}$)
价值网络超参数（多层感知机）	**价值**
层数	2
嵌入维度	MuJoCo 为 256，其他为 512
激活函数	SiLU（Elfwing 等人，2018年）
优化器	AdamW（Loshchilov 和 Hutter，2017年）

B 实验细节

B.1 超参数

本节介绍我们实验中使用的超参数。对于ODT+TD3和ODT基线，我们使用 Yan 等人 (2024年）提供的代码库和默认超参数。对于IQL基线，我们基本遵循 Yan 等人 (2024年）的实现，但将预训练步数设为与我们的方法及其他基线相同，以确保比较公平。

表 5 总结了我们的算法在所有环境中使用的通用架构和训练超参数，这些设置在很大程度上遵循Yan 等人（2024年）的做法。遵循Zheng 等人（2022年）和Yan 等人（2024年），我们不使用位置嵌入。以下提供了算法特定的其他细节。

对于 GRPO-决策变换器，我们在 MuJoCo 和 AntMaze 中每次迭代收集 1 条完整轨迹，在 Adroit 中每次迭代收集 5 条完整轨迹。完整轨迹回放缓冲区最多存储 32 条轨迹。重置时，我们从缓冲区中采样 16 条轨迹，每条轨迹选取 4 个重置点，并构建大小为 8 的组，从而每次迭代产生 512 条子轨迹。子轨迹缓冲区最多存储 2048 条子轨迹。在构建子轨迹组时，我们会丢弃原始奖励与组平均奖励相差不超过加性边际Δ_r 的子轨迹，以提供更强的优化信号。GRPO-决策变换器的环境特定超参数见表 6。

对于Q引导的GRPO-决策变换器，我们在Ant-medium-replay-v2（图 4b）上进行实验，并基本遵循Yan 等人（2024年）训练Q函数以及使用我们的GRPO-决策变换器训练策略网络的做法。为提高训练稳定性，我们将策略学习率改为$\mathrm{lr}_{\mathrm{policy}} = 1 \times 10^{-5}$，并将初始熵对偶变量设为 $\kappa_{\mathrm{initial}} = 1 \times 10^{-2}$。

对于PPO-决策变换器，在MuJoCo和AntMaze中，我们每次迭代收集8条完整轨迹；在Adroit中，每次迭代收集8或16条完整轨迹（见表 7）。完整轨迹回放缓冲区的容量为每次迭代所收集轨迹数量的4倍。按照ODT+TD3的做法，我们对Adroit和AntMaze中的价值网络应用LayerNorm（Ba 等人，2016），以提高训练稳定性。PPO-决策变换器的环境特定超参数见表 7。

B.2 PPO-DT 的实现细节

本节介绍我们对 PPO-DT（算法 2）的实现，它将经典的近端策略优化（PPO，Schulman等人，2017年）应用于决策变换器（Chen等人，2021年）。

表 6：GRPO-DT 的环境特定超参数。 n_{batch} 表示训练批次大小。 c_{offline}和 c_{online} 分别表示离线阶段和在线阶段所使用的上下文长度。 g_{online} 表示在线探索的初始剩余回报。 L_{traj} 表示子轨迹展开长度， L_{eval} 表示额外评估步数。 γ 是用于计算子轨迹奖励的折扣因子。 ε表示 GRPO 裁剪超参数， Δ_r 表示用于子轨迹筛选的加性边际。$\text{lr}_{\text{policy}}$ 表示策略网络的学习率， κ_{initial} 表示在线微调的熵对偶变量初始值，该值会在训练过程中自适应更新。

环境	n_{batch}	c_{offline}	c_{online}	g_{online}	L_{traj}	L_{eval}	γ	ε	Δ_r	学习率policy	κ_{initial}
Ho-M/MR-v2	256	20	1	7200	15	400	0.995	0.2	2.0	5e-5	0.20
Ho-R-v2	256	20	1	7200	15	400	0.995	0.2	2.0	5e-5	0.20
Wa-M/MR-v2	256	20	1	10000	15	400	0.995	0.3	2.0	5e-5	0.04
Wa-R-v2	256	20	1	10000	15	400	0.995	0.3	2.0	5e-5	0.20
An-M/MR-v2	256	20	1	12000	15	200	0.995	0.3	2.0	5e-5	0.04
An-R-v2	256	20	1	12000	15	200	0.995	0.3	2.0	5e-5	0.20
D-C-v1	512	5	1	3000	10	100	0.99	0.3	0.5	3e-5	0.10
D-H-v1	512	5	1	3000	10	100	0.99	0.3	0.4	3e-5	0.04
P-C-v1	512	5	1	6000	3	30	0.99	0.3	0	3e-5	0.02
P-H-v1	512	5	1	6000	3	30	0.99	0.3	0	3e-5	0.02
H-C-v1	512	5	5	4000	10	100	0.99	0.3	0	3e-5	0.05
H-H-v1	512	5	5	4000	10	100	0.99	0.3	0.8	3e-5	0.05
U-v2	256	5	1	2	10	200	1.0	0.2	0	5e-5	0.05
UD-v2	256	1	5	2	10	200	1.0	0.2	0	5e-5	0.05

在每次迭代中，我们采样 K_{PPO} 条完整轨迹。对于每条轨迹 τ 以及每个时间步 h，我们通过广义优势估计（GAE，Schulman 等人，2015年）计算优势值：

$$\widehat{A}_h = \sum_{\ell=0}^{T-h-1} (\gamma\lambda)^{\ell} \Big(r_{h+\ell} + \gamma V_{\phi_t}(s_{h+\ell+1}) - V_{\phi_t}(s_{h+\ell}) \Big), \tag{4}$$

其中 V_{ϕ_t} 表示学习得到的价值函数， $\gamma \in [0,1]$ 表示折扣因子， $\lambda \in [0,1]$ 则控制 GAE 的偏差-方差权衡。为更新策略网络，我们首先随机采样长度为 $\tau_k^{\text{sub}} = (s_{k_1},\ a_{k_1},\ g_{k_1},\ \cdots,\ s_{k_L},\ a_{k_L},\ g_{k_L})$ 的子轨迹，以构成批次 $L = c_{\text{train}}$；遵循Huang 等人 (2022年的做法，我们进一步对整个批次中的优势值进行归一化。随后，通过最大化以下目标函数来更新参数：

$$J_{\text{PPO}}(\theta) = \frac{1}{|\mathcal{B}_{\text{sub}}||\tau_k^{\text{sub}}|} \sum_{\tau_k^{\text{sub}} \in \mathcal{B}_{\text{sub}}} \sum_{i=1}^{|\tau_k^{\text{sub}}|} \min\left(\frac{\pi_\theta(a_{k_i} \mid s_{k_i}, g_{k_i})}{\pi_{\theta_{\text{old}}}(a_{k_i} \mid s_{k_i}, g_{k_i})} \widehat{A}_{k_i},\ \text{clip}\left(\frac{\pi_\theta(a_{k_i} \mid s_{k_i}, g_{k_i})}{\pi_{\theta_{\text{old}}}(a_{k_i} \mid s_{k_i}, g_{k_i})}, 1-\varepsilon, 1+\varepsilon \right) \widehat{A}_{k_i} \right) - \beta D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\theta_{\text{ref}}}). \tag{5}$$

与 GRPO-决策变换器类似，我们在 式（5）中加入额外的熵项 $\kappa H(\pi_\theta(\tau_k^{\text{sub}} \mid s_k,\ g_k))$，并在训练过程中自适应地更新 κ，以确保熵约束 $H(\pi_\theta(\tau_k^{\text{sub}} \mid s_k,\ g_k)) \geq \rho$得到近似满足。

通过最小化采样子轨迹中状态对应的价值目标 $V_{\text{target}(s_h)} = \widehat{A}_h + V_{\phi_t}(s_h)$ 的平方损失来更新价值函数：

$$\mathcal{L}_{\text{value}} = \mathbb{E}_{s_h}\left[(V_\phi(s_h) - V_{\text{target}}(s_h))^2 \right]. \tag{6}$$

表 7：PPO-DT 的环境特定超参数。 n_{batch} 表示训练批次大小。 c_{offline}和 c_{online} 分别表示离线阶段和在线阶段使用的上下文长度。 g_{online} 表示在线探索的初始剩余回报。 K_{PPO} 表示每次迭代在线进行完整轨迹滚动采样的次数。$\text{lr}_{\text{policy}}$ 和lr_{value} 分别表示策略网络和价值网络的学习率。 κ_{initial}表示在线微调时熵对偶变量的初始值，该值会在训练过程中自适应更新。在所有环境中，我们将 PPO 裁剪超参数设为 $\varepsilon = 0.2$ ，将 GAE 超参数（γ、 λ）设为 $=$ (0.99、0.95)。

环境	n_{batch}	c_{offline}	c_{online}	g_{online}	K_{PPO}	$\text{lr}_{\text{policy}}$	lr_{value}	κ_{initial}
Ho-M/MR-v2	256	20	1	7200	8	5e-5	1e-3	0.02
Ho-R-v2	256	20	1	7200	8	5e-5	1e-3	0.04
Wa-M/MR-v2	256	20	1	10000	8	5e-5	1e-3	0.02
Wa-R-v2	256	20	1	10000	8	5e-5	1e-3	0.20
An-M/MR-v2	256	20	1	12000	8	5e-5	1e-3	0.02
An-R-v2	256	20	1	12000	8	5e-5	1e-3	0.02
D-C-v1	512	5	1	3000	16	3e-5	2e-4	0.002
D-H-v1	512	5	1	3000	16	3e-5	2e-4	0.002
P-C-v1	512	5	1	6000	8	3e-5	2e-4	0.04
P-H-v1	512	5	1	6000	8	3e-5	2e-4	0.04
H-C-v1	512	5	5	4000	16	3e-5	2e-4	0.005
H-H-v1	512	5	5	4000	16	3e-5	2e-4	0.005
U-v2	256	5	1	2	8	5e-5	1e-3	0.02
UD-v2	256	1	5	2	8	5e-5	1e-3	0.02

算法 2 使用 PPO 在线微调决策变换器（PPO-DT）输入：预训练策略 π_{θ_1}、价值网络 V_{ϕ_1}、完整轨迹缓冲区 $\mathcal{T}_{\text{replay}}$、迭代次数 T、初始剩余回报 g_{online}、每次迭代收集的轨迹数量 n_{PPO}、子轨迹长度 CL_{train}。1：对于迭代 $t = 1, \cdots, T$，执行 2：使用当前策略 $\pi_{\theta_t}(\cdot \mid s_1, g_{\text{online}})$ 执行 K_{PPO} 条轨迹，以初始（随机化）状态 s_1 和回报目标 g_{online} 为条件；更新 $\mathcal{T}_{\text{replay}}$。// 收集完整轨迹；更新先进先出缓冲区。3：对于每条轨迹 $\tau \in \mathcal{T}_{\text{replay}}$，根据广义优势估计（公式（4））计算每一步的优势。4：从 $\mathcal{T}_{\text{replay}}$ 中按照分布 $p(\tau) = \frac{|\tau|}{\sum_{\tau \in \mathcal{T}}}$ 重放$|\tau|$，采样一批 $\mathcal{B}$ 条完整轨迹。5：对于每条轨迹 $\tau \in \mathcal{B}$，随机采样一条长度为 $L = c_{\text{train}}$ 的子轨迹 τ^{sub}，形成子轨迹批次 $\mathcal{B}_{\text{sub}}$。对批次 $\mathcal{B}_{\text{sub}}$ 中的优势值进行归一化。6：根据公式~~（5）~~微调当前策略，得到更新后的 $\pi_{\theta_{t+1}}$；根据公式（6）微调价值网络，得到更新后的 $V_{\phi_{t+1}}$。输出：在线微调策略 $\pi_{\theta_{T+1}}$。