

参赛学生姓名:

陈雨涵、林子博、阮邦城

中学:

西南大学附属中学校

省份:

重庆

国家/地区:

中国

指导老师姓名:

邓白龙、来泽

指导老师单位:

西南大学附属中学校

论文题目:

一种基于判别器同步优化的偏好模仿学习方法——基于雅达利 2600 游戏

一种基于判别器同步优化的偏好模仿学习方法

——基于雅达利 2600 游戏

陈雨涵 林子博 阮邦城

摘要

强化学习的核心难题之一是奖励函数的设计。目前主流的偏好强化学习方法遵循两阶段流程：先从人类偏好中拟合一个奖励函数，再以此训练策略。这一中间步骤在复杂环境中可能引入偏差——当偏好数据有限时，拟合出的奖励函数在分布外状态上可能给出错误的引导。为此，本文提出偏好模仿学习（PreIL），不单独建模奖励函数，而是让判别器作为奖励信号的直接来源，在策略训练过程中同步更新，使奖励信号始终适应当前的策略分布。

本文对 Bradley-Terry 模型提出了三个方面的质疑：它将轨迹奖励视为逐步加和，丢失了动作的时序信息；从有限的偏好对中恢复每步精确奖励值，本身是一个不适定问题；其逐时刻加总的假设也与人类整体性的判断方式不符。基于这些观察，本文用全局时序品质假设（GTQ）概括人类对轨迹的整体评价特征，并归纳出三条基本性质作为后续分析的基础。

受生成对抗模仿学习（GAIL）的启发，本文设计了一个基于 LSTM 的时序判别器，用来区分更优轨迹和较差轨迹。判别器的输出可以直接作为奖励信号，指导强化学习智能体提升策略。本文进一步将 n 步 Q 学习与偏好模仿学习相结合，通过轨迹子序列的判别器输出来构造学习目标，取得了更好的效果。

在 6 个 Atari 游戏上，PreIL 的平均人类标准化得分达到 45.6%，而基于奖励恢复的 PrefDQN 和 PrefNSDQN 仅为 -3.1% 和 -12.1%。对比实验进一步表明，这一性能优势主要源于模仿学习框架本身，而非 n 步 Q 学习的引入。本文的研究成果表明，在偏好数据有限的

情况下，放弃显式奖励建模、直接在策略层面利用偏好信息，可能是比传统两阶段范式更高效的选择。

关键词：强化学习；偏好学习；模仿学习；判别器奖励；时序判别器；Atari 游戏

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Abstract

Designing a proper reward function remains a core challenge in reinforcement learning. Mainstream preference-based reinforcement learning methods follow a two-stage procedure: first fit a reward function from human preferences, then train a policy using it. This intermediate step can introduce bias in complex environments—when preference data is limited, the fitted reward function may give misleading guidance on out-of-distribution states. To address this, we propose Preference-based Imitation Learning (PreIL), which does not model the reward function separately. Instead, a discriminator serves as the direct source of reward signals and is updated synchronously during policy training, ensuring that the reward signal always adapts to the current policy distribution.

We question the Bradley-Terry model from three aspects: it decomposes trajectory reward into a step-wise sum, thereby losing temporal ordering information; recovering precise per-step reward values from limited preference pairs is fundamentally ill-posed; and its assumption of step-wise summation conflicts with the holistic nature of human judgment. Based on these observations, we introduce the concept of Global Temporal Quality (GTQ) to characterize humans' overall evaluation of trajectories, and summarize three basic properties as the foundation for subsequent analysis.

Inspired by Generative Adversarial Imitation Learning (GAIL), we design an LSTM-based temporal discriminator to distinguish preferred trajectories from dispreferred ones. The discriminator output directly serves as a dense reward signal to guide policy improvement. We further integrate n-step Q-learning with preference imitation learning, constructing learning targets from discriminator outputs over trajectory subsequences, which yields improved performance.

Experimental results on six Atari games show that PreIL achieves an average HNS of 45.6%, compared to -3.1% for PrefDQN and -12.1% for PrefNSDQN. Ablation studies indicate that the performance gain originates primarily from the imitation learning framework rather than n-step Q-learning. This work offers an alternative perspective for preference-based reinforcement learning.

Keywords: reinforcement learning; preference learning; imitation learning; discriminator-based reward; temporal discriminator; Atari games

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

目录

摘要	2
Abstract	4
第 1 章 引言	8
第 2 章 相关工作	10
2.1. 模仿学习	10
2.2. 偏好学习	10
第 3 章 背景与预备知识	11
3.1. 马尔可夫决策过程	11
3.2. 偏好强化学习与 Bradley-Terry 模型	11
3.3. 模仿学习与占用度量	12
第 4 章 研究方法	13
4.1. 全局时序品质假设的基本性质	13
4.2. Bradley-Terry 模型的三个问题	13
4.3. PreIL 框架	14
4.4. 时序判别器设计	15
4.5. n 步 Q 学习扩展	15
4.6. PreIL 算法流程	16
第 5 章 实验	18
5.1. 实验设置	18
5.2. 主实验结果	19
5.3. 参数敏感性分析	21
5.4. 奖励恢复失败的可视化分析	23
第 6 章 讨论与结论	24

第 7 章 参考文献	25
附 录	27
一、实验超参数设置	27
二、实验环境	27
致 谢	29

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

第 1 章 引言

深度强化学习已经在 Atari 游戏[1]、机器人操控[2]、自动驾驶[3]等场景中取得了令人瞩目的成果，其核心机制是让智能体通过与环境交互、依据奖励信号来学习决策策略。但奖励函数本身的设计往往构成瓶颈，如果奖励设置不当，智能体可能找到奖励函数的漏洞而非完成预期任务，这种现象被称为规格博弈（specification gaming）[4]。如何减少对人工设计奖励函数的依赖，是强化学习走向实际应用必须面对的问题。

为了解决奖励设计这个问题，现有的研究主要探索了两条路径。

1. 模仿学习（Imitation Learning, IL），通过学习专家的演示来获得策略，不需要设计奖励函数[5]。但是模仿学习需要专家演示，而很多时候获取高质量的专家演示本身就很困难。

2. 从人类反馈中学习（Learning from Human Feedback, LHF），其中基于偏好的强化学习（Preference-based Reinforcement Learning, PbRL）只需要人类比较两段轨迹哪一段更好，大大降低了标注难度[6]。

目前的偏好强化学习方法大多分为两个阶段：

1. 第一阶段，用 Bradley-Terry（BT）模型[7]从偏好对中学习一个奖励函数；
2. 第二阶段，用这个奖励函数训练强化学习算法。这种两阶段方法在 DeepMind 的工作[6]中被提出，后来被很多研究采用[8,9]。

但是本文在学习过程中发现，这种先恢复奖励再优化策略的做法可能有问题。当人类比较两段完整轨迹的好坏时，是对轨迹的整体印象做出判断，并不是先给每一步打个分再加起来。如果硬要把整体的偏好拆解成每一步的奖励，可能会丢失很多重要信息，而且恢复出来的奖励不一定可靠。

本文对 BT 模型的质疑集中在三个方面：

1. 时序信息的丢失：BT 模型将轨迹奖励视为逐步加和，交换任意两个时刻的奖励值不改变总奖励，但动作的先后顺序往往决定轨迹成败——打砖块游戏中先击中侧边砖块和先击中底部砖块，效果截然不同。

2. 奖励恢复的不适定性：偏好比较仅提供轨迹级别的相对排序，却要据此推断每个状态-动作对的精确数值，满足同一组偏好数据的奖励函数有无穷多个，它们在分布外状态上的取值可能完全不同。

3. 认知偏差：人判断轨迹好坏依赖整体印象和模式识别，BT 模型逐时刻加总的假设与这一直觉不符。

基于这些观察，本文想到：能不能绕开显式奖励恢复这一步，直接从偏好中学策略呢？本文受到生成对抗模仿学习（GAIL）[5]的启发，提出了偏好模仿学习（Preference-based Imitation Learning, PreIL）方法。具体来说，本文训练一个基于 LSTM 的判别器，让它判断一段轨迹是更好还是更差。判别器的输出可以直接作为奖励信号，指导智能体提升策略。

本文的核心贡献体现在三个层面：

1. 理论层面，归纳出全局时序品质假设（GTQ）的三条基本性质，并分析了 Bradley-Terry 模型在处理人类偏好时存在的三个问题。

2. 方法层面，提出了偏好模仿学习（PreIL）方法，通过时序判别器直接从偏好对中学习策略，不单独建模奖励函数。

3. 实验层面，6 个 Atari 游戏上通过实验验证了 PreIL 的有效性，并通过参数分析和可视化分析探讨了算法的工作原理。

第 2 章 相关工作

2.1. 模仿学习

模仿学习绕开了奖励函数设计，直接从专家演示中学习策略。最直接的做法是行为克隆[10]——将专家的状态-动作对当作监督信号训练映射函数，但这种方法存在分布偏移隐患：智能体一旦进入训练数据未覆盖的状态，误差便会累积放大。另一类方法转而推断专家背后的奖励函数，再以此驱动策略学习，最大熵 IRL[11]、GCL[12]、AIRL[13]分别从不同角度推进了这一路线。GAIL[5]则开辟了第三种思路——用对抗训练替代显式的奖励恢复，判别器的输出直接作为奖励信号。

GAIL[5]借用了 GAN[14]的对抗框架：判别器试图分辨输入来自专家还是当前策略，策略则努力让自身轨迹骗过判别器，双方在博弈中共同进化。GAIL 的判别器输出可直接作为奖励信号，免去了显式恢复奖励函数的步骤。后续研究[15]将 BC、IRL、GAIL 等方法统一到了散度最小化的理论框架下。本文的 PreIL 与 GAIL 在框架上有共通之处，但输入信号不同——GAIL 依赖专家演示，PreIL 使用的是人类偏好比较，由此产生了更松的约束和不同的训练动态。

2.2. 偏好学习

偏好学习的输入是成对的比较结果而非绝对评分，标注成本显著低于逐时刻打分。Akrouer 等人[16]最早尝试直接从全轨迹偏好中学习策略，Biyik 等人[17]进一步用主动采样减少所需的反馈次数。偏好建模的核心假设可追溯到两个经典模型：BT 模型[7]和 Thurstone 模型[18]，二者均假设每个选项存在潜在价值，比较概率由价值差决定，区别在于概率分布的形式——BT 用 Logistic 函数，Thurstone 用正态分布。

在强化学习领域，Christiano 等人[6]提出了 Deep RL from Human Preferences 框架，确立了奖励学习加策略优化的两阶段范式，并在 Atari 和 MuJoCo 环境上验证了效果。后续有很多工作在这个基础上改进，比如 PEBBLE[8]用不确定性采样来减少需要的偏好数量。这些方法大多都要先恢复奖励函数。本文的不同之处在于，本文尝试绕开显式奖励恢复，直接从偏好中学习策略。

第3章 背景与预备知识

3.1. 马尔可夫决策过程

本研究的形式化框架是马尔可夫决策过程（MDP），由状态空间 S 、动作空间 A 、转移概率 P 、奖励函数 r 和折扣因子 $\gamma \in [0,1)$ 五个要素构成。其中 r 是本文讨论的核心变量：在标准设定中 r 由环境给定，但在偏好学习场景下 r 需要从人类反馈中推断。

策略 π 定义了在每个状态下选择各个动作的概率。策略 π 的期望折扣回报是：

$$J(\pi) = E \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] \quad (1)$$

强化学习的目标是找到最优策略 π^* 来最大化期望折扣回报。在本文的研究中，环境不提供显式的奖励函数 r ，奖励需要从人类偏好中学习。

3.2. 偏好强化学习与 Bradley-Terry 模型

偏好强化学习力图以最少的人类反馈获取符合期望的策略。其标准流程为：收集轨迹片段 $\rightarrow$ 由人类标注偏好 $\rightarrow$ 拟合奖励函数 $\rightarrow$ 训练策略，然后交替迭代。值得注意的是，奖励函数拟合与策略优化这两个阶段是解耦的——奖励函数在拟合后即固定，不再随策略更新而变化，这一特点既是其模块化的优势，也可能成为性能瓶颈的来源。

具体来说，智能体和环境交互，收集轨迹片段。给定两段轨迹片段 σ_i 和 σ_j ，如果人类觉得 σ_i 更好，就记为 $\sigma_i > \sigma_j$ 。设 π_θ 是参数化的策略， r_ψ 是参数化的奖励函数。算法交替执行两个步骤：（i）智能体学习——策略 π_θ 和环境交互，用学到的奖励函数 r_ψ 来更新策略；（ii）奖励学习——根据人类反馈更新奖励函数 r_ψ 。

3.2.1. Bradley-Terry 模型

Bradley-Terry（BT）模型是偏好学习中最常用的模型。它假设每个选项有一个潜在的价值，偏好概率由价值的相对大小决定。对于两段轨迹 σ_i 和 σ_j ，BT 模型给出 σ_i 更好的概率是：

$$P(\sigma_i > \sigma_j) = \frac{\exp\left(\sum_{(s_t, a_t) \in \sigma_i} \hat{r}_\psi(s_t, a_t)\right)}{\exp\left(\sum_{(s_t, a_t) \in \sigma_i} \hat{r}_\psi(s_t, a_t)\right) + \exp\left(\sum_{(s_t, a_t) \in \sigma_j} \hat{r}_\psi(s_t, a_t)\right)} \quad (2)$$

其中 $\hat{r}_\psi(s, a)$ 是参数化的奖励函数。通过最大化偏好数据的似然，可以学到奖励函数的参数 ψ 。对应的损失函数是：

$$\mathcal{L}(\hat{r}_\psi) = - \sum_{(\sigma_i, \sigma_j, u) \in D} [u_i \log P(\sigma_i > \sigma_j) + u_j \log P(\sigma_j > \sigma_i)] \quad (3)$$

其中 $u=(u_i, u_j)$ 是偏好标签，如果 σ_i 更好则 $u=(1,0)$ ，如果 σ_j 更好则 $u=(0,1)$ 。

3.3. 模仿学习与占用度量

当状态空间维度升高、偏好数据有限时，奖励恢复的误差会显著放大。模仿学习提供了一条规避思路：不显式恢复奖励，而是直接从数据到策略。其中 GAIL[5]的方案尤为巧妙——它用 GAN[14]的对抗框架替代了显式奖励建模，判别器的输出本身即充当奖励。这一设计启发了本文的核心想法：既然判别器可以区分"专家与策略"，那它同样可以区分"人偏好的与人不偏好的"。

占用度量（occupancy measure）是模仿学习中的一个重要概念。对于策略 π ，占用度量 $\rho_\pi(s, a)$ 定义为在策略 π 下，状态-动作对 (s, a) 被访问的折扣频率：

$$\rho_\pi(s, a) = \pi(a|s) \sum_{t=0}^{\infty} \gamma^t P(s_t = s | \pi) \quad (4)$$

占用度量和策略是一一对应的。强化学习的目标可以用占用度量重新表示为：

$$\eta(\pi) = E_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] = \sum_{s, a} \rho_\pi(s, a) \cdot r(s, a) \quad (5)$$

GAIL 的理论基础是占用度量匹配：当两个策略的占用度量相等时，它们产生的状态-动作分布一致，行为也就等价。GAIL 用判别器来衡量这种分布差异——判别器越难分辨输入来自哪一方，说明两个分布越接近。在对抗训练的均衡点，策略生成的轨迹与专家轨迹在分布层面重合，此时判别器的输出便退化为均匀分布，但训练过程中判别器的输出可作为密集奖励信号引导策略改进。

这些工作启发了本文：在偏好学习中，是不是也可以用类似的方式，绕开显式奖励恢复，直接通过判别器从偏好对中学习策略？这就是本文提出 PreIL 方法的想法来源。

第 4 章 研究方法

4.1. 全局时序品质假设的基本性质

在研究偏好学习的过程中，本文注意到人类对轨迹好坏的判断，是一种整体的、直觉的评价，而不是逐时刻计算出来的。根据此判断，本文使用全局时序品质假设（Global Temporal Quality, GTQ）这个概念，来表示一条完整轨迹的好坏程度。人类比较两条轨迹，本质上就是在比较它们的 GTQ。

通过观察和思考，本文总结了 GTQ 的三条基本性质：

性质 1（非可加性）：轨迹的整体品质不等于各时刻品质的简单加和。整体和部分不一样，轨迹的时间结构和变化过程对整体品质很重要。

性质 2（时序依赖性）：整体品质和动作的顺序有关。交换两个时间步的动作，整体评价可能会改变，因为动作的先后顺序本身就是品质的一部分。

性质 3（质性可判别性）：人类能判断哪个更好，但说不出具体好多少。偏好比较只能给出顺序，不能给出精确的数值，因为人类的判断更多是定性的而不是定量的。

这三条性质是本文分析现有方法问题的出发点。

4.2. Bradley-Terry 模型的三个问题

基于 GTQ 的三条性质，本文可以看到 Bradley-Terry 模型在处理人类偏好时存在三个问题。

1. 时序信息丢失。BT 模型把轨迹的总奖励看成每一步奖励的加和，这意味着交换任意两个时刻的奖励，总奖励不变，偏好判断也不变。但实际上，动作的顺序往往决定了轨迹的成败，这和 GTQ 的时序依赖性（性质 2）是矛盾的。

2. **分布外外推不可靠**。偏好比较只能给出轨迹级别的相对顺序，而 BT 模型要恢复每一步的精确奖励值。从有限的偏好对中恢复出所有状态-动作对的奖励，这个问题是不稳定的——有很多不同的奖励函数都能拟合同样的偏好数据，但它们在没见过的状态上可能给出完全不同的奖励。这和 GTQ 的质性可判别性（性质 3）有关。

3. **与人类认知方式不匹配**。BT 模型假设人类是逐时刻计算奖励然后加总来做比较的，但实际上人类更多是依靠整体印象和模式识别来判断好坏，不是做精确的数值计算。BT 模型的数学假设和人类实际的思考方式有差距。

这三个问题让本文觉得，奖励恢复这条路可能不是最优的，应该试试直接从偏好中学策略的方法。

4.3. PreIL 框架

本文提出偏好模仿学习（Preference-based Imitation Learning, PreIL）方法。它的核心思想是：绕开显式奖励恢复，直接从人类偏好中学习策略。

PreIL 借鉴了 GAIL 的思路，用占用度量匹配的方式来理解偏好学习：本文要找一个策略 π ，使得它产生的轨迹分布尽可能接近更优轨迹的分布，同时尽可能远离较差轨迹的分布。这和模仿学习很像，只是把专家演示换成了偏好对中的优/劣轨迹。

具体来说，本文训练一个判别器 $D(s, a; \phi)$ ，它的目标是区分来自更优集合的状态-动作对和来自较差集合的状态-动作对。判别器的损失函数是：

$$\mathcal{L}(\phi) = E_{(s,a) \sim \sigma_N \cup \tau} [\log D(s,a;\phi)] + E_{(s,a) \sim \sigma_P} [\log (1 - D(s,a;\phi))] \quad (6)$$

其中 σ_P 是偏好对中更优的轨迹片段， σ_N 是较差的轨迹片段， τ 是当前策略产生的轨迹。判别器的输出 $D(s, a; \phi)$ 可以直接作为奖励信号，来指导强化学习智能体提升策略。

用判别器输出 $D(s, a; \phi)$ 代替环境奖励 r ，策略的损失函数为：

$$\mathcal{L}(\theta) = E \left[\left(D(s_t, a_t; \phi) + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta') - Q(s_t, a_t; \theta) \right)^2 \right] \quad (7)$$

其中 θ' 是目标网络的参数。

4.4. 时序判别器设计

判别器是 PreIL 的关键部分，它的作用是评估一段轨迹的品质。因为轨迹数据有时间顺序，本文选择用 LSTM 作为判别器的主要结构，这样可以更好地捕捉时序特征。

判别器的结构设计如下：

1. 嵌入层：把每一步的状态和动作转换成低维的特征向量。对于 Atari 游戏的 RAM 输入，用全连接层提取特征；对于离散动作，用嵌入层得到动作表示，然后把两者拼在一起。
2. LSTM 层：把特征向量序列输入 LSTM，利用 LSTM[20]的门控机制来捕捉时间上的依赖关系。取 LSTM 最后一步的隐藏状态作为整个轨迹片段的表示。
3. 输出层：把 LSTM 的隐藏状态通过全连接层映射成一个数值，再经过 Sigmoid 激活函数得到 0 到 1 之间的概率，表示轨迹是更优的概率。
4. 与逐状态-动作对的判别器相比，时序判别器能更好地捕捉轨迹的整体品质，因为它考虑了动作之间的时间关系。这和 GTQ 的时序依赖性（性质 2）是一致的。

4.5. n 步 Q 学习扩展

Q 学习[21]是一种经典的值函数方法，为了更直接地利用偏好信息，本文把 PreIL 扩展到轨迹子序列上，而不是单个状态-动作对。本文把判别器从 $D(s, a; \phi)$ 改成 $D(\sigma_d; \phi)$ ，其中 σ_d 是包含 n 个连续状态-动作对的轨迹子序列： $\sigma_d = [(s_t, a_t), \dots, (s_{t+n-1}, a_{t+n-1})]$ 。

对应的判别器损失函数变成：

$$\mathcal{A}(\phi) = E_{\sigma_d \sim \sigma_N \cup \tau} [\log D(\sigma_d; \phi)] + E_{\sigma_d \sim \sigma_P} [\log (1 - D(\sigma_d; \phi))] \quad (8)$$

4.5.1. n 步 Q 学习的形式化

n 步 Q 学习[22]是 Q 学习的扩展，它用多步的真实奖励来计算学习目标，在偏差和方差之间有更好的平衡。标准的 n 步 Bellman 方程是：

$$Q(s_t, a_t) = E \left[\sum_{j=0}^{n-1} \gamma^j r(s_{t+j}, a_{t+j}) + \gamma^n \max_{a_{t+n}} Q(s_{t+n}, a_{t+n}) \right] \quad (9)$$

PreIL 中，轨迹子序列的奖励不是来自环境，而是来自判别器 $D(\sigma_d)$ 。从占用度量的角度看，判别器的输出就是对轨迹子序列整体品质的评估，所以可以直接用来近似 n 步回报。

把判别器输出代入 n 步 Bellman 方程，得到：

$$Q(s_t, a_t) = E \left[D(\sigma_d) + \gamma \max_{a_{t+n}} Q(s_{t+n}, a_{t+n}) \right] \quad (10)$$

与标准 n 步 Bellman 方程不同，这里的奖励来自判别器 $D(\sigma_d)$ 而非环境，判别器对整个子序列的评估已包含了 n 步的奖励信息。

对应的损失函数是：

$$\mathcal{L}(\theta) = E \left[\left(D(\sigma_d) + \gamma \max_{a_{t+n}} Q(s_{t+n}, a_{t+n}; \theta) - Q(s_t, a_t; \theta) \right)^2 \right] \quad (11)$$

4.6. PreIL 算法流程

PreIL 的算法流程如下。算法主要有三部分：收集轨迹存到经验回放[23]缓冲区、用 n 步 DQN 优化 Q 函数、定期查询人类偏好并更新判别器。

Algorithm1 Preference-based Imitation Learning

Input: 每次更新间隔步数: k ; n 步 DQN 步数: n ; 偏好查询间隔步数: j

```
1  初始化: 经验回放记忆  $M$ , 偏好数据库  $\Lambda$ , 策略网络参数  $\theta$ , 目标网络参数  $\theta'$ , 判别器参数  $\phi$ 
2  for 每次迭代 do
3    // 阶段一: 收集轨迹
4    for  $t = 1, \dots, k$  do
5      以概率  $\epsilon$  选随机动作, 否则选  $a_t = \arg \max_a Q(s_t, a; \theta)$ 
6      执行动作  $a_t$ , 得到下一状态  $s_{t+1}$ 
7      将  $(s_t, a_t, s_{t+1})$  存入  $M$ 
8    end for
9    // 阶段二:  $n$  步 DQN 优化策略
10   从  $M$  中采样轨迹子序列  $\sigma_b = [(s_b, a_b), \dots, (s_{b+n-1}, a_{b+n-1})]$ 
11   用判别器评估奖励:  $r_b = D(\sigma_b; \phi)$ 
12   计算 TD 目标:  $y_b = r_b + (1 - \text{done}) \cdot \gamma \cdot \max_a Q(s_{b+n}, a; \theta')$ 
13   更新策略网络, 最小化损失:  $L(\theta) = (y_b - Q(s_b, a_b; \theta))^2$ 
14   // 阶段三: 查询偏好并更新判别器
15   if 迭代次数是  $j$  的倍数 then
16     从  $M$  中采样轨迹片段对, 向人类查询偏好  $u$ 
17      $\Lambda \leftarrow \Lambda \cup \{(\sigma_1, \sigma_2, u)\}$ 
18     从  $\Lambda$  中取偏好片段  $\sigma_P$  和非偏好片段  $\sigma_N$ 
19     更新判别器, 最小化损失:  $L(\phi) = E_{\{\sigma \sim \sigma_N\}} [\log D(\sigma; \phi)] + E_{\{\sigma \sim \sigma_P\}} [\log(1 - D(\sigma; \phi))]$ 
20   end if
21 end for
```

和传统的 n 步 DQN 不同, 在 PreIL 中, 轨迹子序列的奖励是由判别器评估的, 而判别器会随着新的人类反馈不断更新。所以每次从回放记忆中采样轨迹子序列时, 都要用当前的判别器重新计算奖励值。这是 PreIL 和标准 n 步 DQN 的一个重要区别。

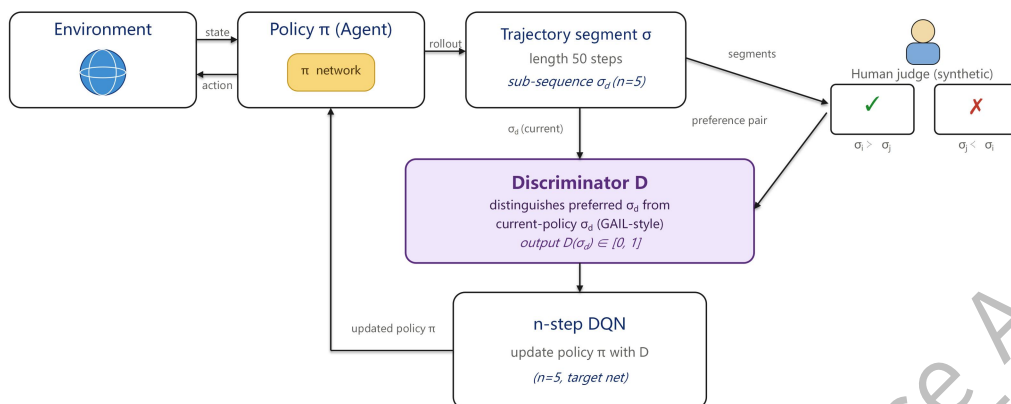


图 4-1 Preil 强化学习网络架构

第 5 章 实验

本章通过在 Atari 2600 游戏上的实验来验证 PreIL 方法的效果。

5.1. 实验设置

实验在六个 Atari 2600 游戏上进行，分别是 AirRaid、Alien、Breakout、Seaquest、SpaceInvaders、Tutankham。选择依据是覆盖不同玩法类型——AirRaid 和 SpaceInvaders 侧重反应速度，Alien 和 Tutankham 涉及空间探索与路径规划，Breakout 考验精准操控，Seaquest 需要多阶段策略切换。episode 长度统一设为 1000 步以保证不同游戏间的公平对比。

本文把 PreIL 和下面几种方法做对比：

1. DQN：用环境真实奖励训练的标准 1 步 DQN，作为性能参考的上限。
2. NSDQN：用环境真实奖励训练的 n 步 DQN，用来看看 n 步回报本身对性能有多大影响。
3. PrefDQN：基于 Bradley-Terry 模型的传统两阶段偏好学习方法[6]，先恢复奖励函数再训练策略。
4. PrefNSDQN：PrefDQN 的 n 步版本，用恢复的奖励函数和 n 步 DQN 训练策略。

5. PreIL: 本文提出的方法, 绕开显式奖励恢复, 直接通过时序判别器从轨迹偏好中学习策略。

所有偏好学习方法用的偏好反馈数量相同: 总共 5000 对偏好比较。本文用模拟专家来提供偏好, 模拟专家根据环境的真实累计奖励来判断——累计奖励更高的轨迹片段就是更优的。每条用于偏好比较的轨迹片段长度是 50 步。

所有神经网络用 PyTorch[24]实现。实验在 Arcade Learning Environment (ALE) [25]上运行, 通过 OpenAI Gym 接口调用。本文用的是 Atari 游戏的 RAM 版本, 也就是输入是 128 维的游戏内存状态, 不是像素图像, 这和现有偏好学习研究的做法一致[6,8]。

主要的超参数设置见表 5-1。所有方法的策略网络结构和训练超参数都保持一致, 以保证公平比较。

超参数	取值	说明
n 步长度	5	n 步 DQN 的 n 值
折扣因子 γ	0.99	标准设置
学习率	0.0003	Adam 优化器
Batch size	128	策略更新批次大小
经验回放缓冲区	1M	最大存 100 万步
目标网络更新频率	每 10000 步	硬更新
判别器隐藏维度	256	LSTM 隐藏层大小
最大偏好反馈数	5000 对	总偏好比较数量
训练总步数	300 万步	每个游戏训练 3M 步
随机种子	5 个	报告均值和标准差

表 5-1 实验超参数设置

5.2. 主实验结果

表 5-2 展示了不同方法在六个 Atari 游戏上的性能, 用人类标准化得分 (Human Normalized Score, HNS) 的百分比表示。所有结果是 5 个随机种子的平均值。

游戏	DQN	NSDQN	PrefDQN	PrefNSDQN	PreIL(本文)
AirRaid	82.2%	196.5%	21.1%	11.1%	149.3%
Alien	74.4%	71.0%	10.8%	12.0%	23.2%
Breakout	163.6%	225.0%	-83.9%	-126.6%	-6.5%
Seaquest	3.2%	3.4%	0.5%	0.5%	1.5%
SpaceInvaders	111.3%	132.5%	39.7%	37.3%	94.7%
Tutankham	85.5%	105.6%	-7.0%	-6.8%	11.8%
平均	86.7%	122.4%	-3.1%	-12.1%	45.7%

表 5-2 人类标准化得分 (%)

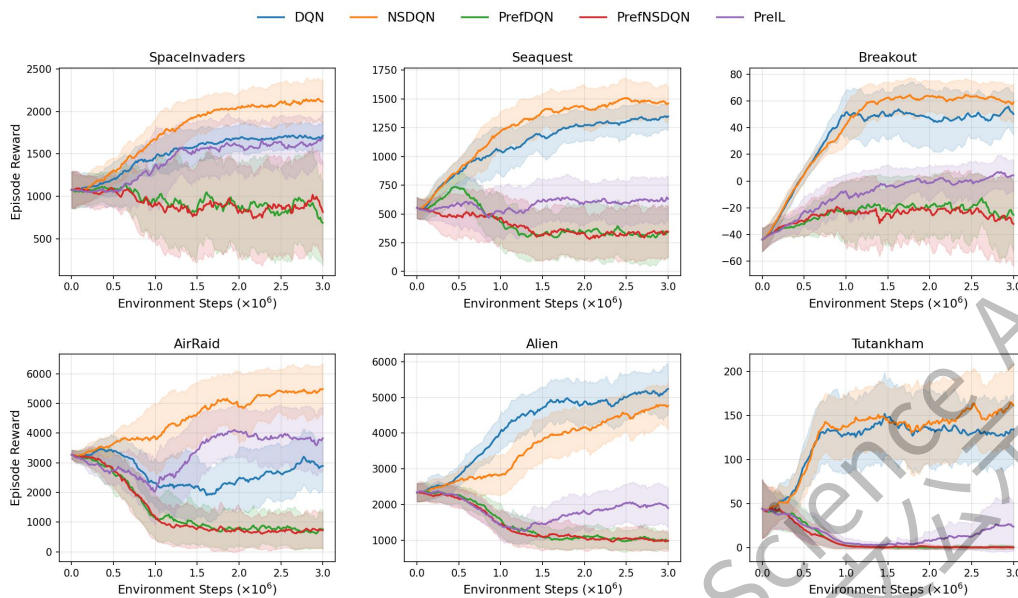


图 5-1 各方法的学习曲线对比

从实验结果可以看到：

1. 基于奖励恢复的方法整体表现很差。PrefDQN 和 PrefNSDQN 的平均人类标准化得分分别为-3.1%和-12.1%，在多数游戏上甚至低于随机策略水平。这说明从有限的偏好对中恢复奖励函数这一中间步骤确实存在根本问题——恢复的奖励在分布外状态下给出了错误的引导信号，反而把策略带偏了。这一实验结果验证了本文在第 4 章中基于 GTQ 性质分析得到的三重困境。

2. PreIL 绕开显式奖励恢复后，平均 HNS 达到 45.6%，在六个游戏上均优于 PrefDQN 和 PrefNSDQN。考虑到 PreIL 完全不知道真实奖励函数、只依赖有限的偏好比较（5000 次查询），这个结果说明绕开显式奖励恢复这一中间环节是有效的策略。PreIL 的优势来自模仿学习框架本身，而非 n 步 DQN：PrefNSDQN 虽然也用了 n 步 DQN，但因为依赖奖励恢复，性能反而更差（-12.1%），远不如 PreIL（45.6%）。

3. 和使用真实奖励的 DQN（86.7%）相比，PreIL 还有差距，约为 DQN 的 52%。这是可以理解的，因为偏好反馈提供的信息比稠密的真实奖励少很多。但在部分游戏上，PreIL 表现尤为突出：在 AirRaid 上，PreIL 的 HNS 达到 149.3%，甚至超过了使用真实奖励的 DQN（82.2%）；在 SpaceInvaders 上，PreIL（94.7%）也接近 DQN（111.3%）。这

说明在适合的游戏环境下，PreIL 有很大潜力。而在 Alien、Tutankham 等游戏上，PreIL 与真实奖励方法差距较大，可能与这些游戏的奖励结构更复杂、偏好信息不足以刻画全部策略空间有关。

综上，实验数据说明绕开显式奖励恢复、直接从偏好中进行模仿学习，才是性能提升的关键。本文的 PreIL 方法已经取得了不错的性能。

5.3. 参数敏感性分析

5.3.1. n 步长度的影响

在训练中， n 步回报的长度 n 是 PreIL 的一个重要超参数。本文在 SpaceInvaders、Seaquest、Breakout 共 3 个游戏上测试了不同 n 值 ($n=1, 2, 5, 10, 20$) 对算法性能的影响。

实验结果如下：

1. $n>1$ 时性能明显优于 $n=1$ 。用轨迹子序列代替单步状态-动作对来评估奖励是有效的。 $n=1$ 时，三个游戏的性能都明显低于 $n\geq 2$ 的情况，说明只靠单步状态-动作对的判别器输出，不足以提供足够的学习信号。

2. 最优的 n 值因游戏而异。在 SpaceInvaders 和 Seaquest 上， $n=5$ 性能较好；在 Breakout， $n=2$ 附近性能最好。这可能与不同游戏的奖励延迟结构有关——奖励越稀疏的游戏，可能需要越长的子序列才能捕捉到有效的奖励信号。

3. n 并不是越大越好。在 Breakout 上， n 增大到 20 时性能骤降；SpaceInvaders 上， $n=20$ 也比 $n=10$ 差。原因可能是过大的 n 步长度会引入更多的离策略偏差——回放记忆中的子序列来自旧策略，用来更新当前策略时，分布偏差会被放大。

综合来看， $n=5$ 在三个游戏上都有不错的表现，虽不总是最优，但也没有出现严重退化。考虑到整体效果和调参成本，主实验中我们取 $n=5$ 作为默认设置。

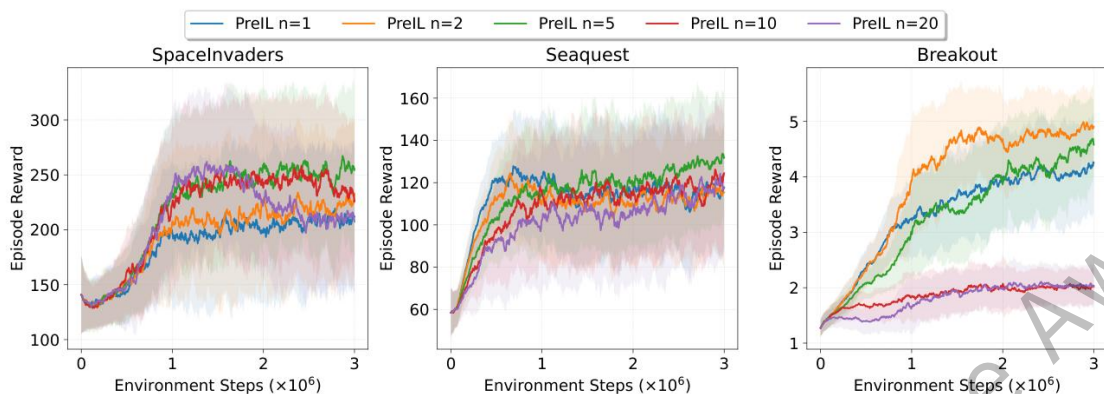


图 5-2 不同 n 步长度下 PreIL 的性能对比

5.3.2. 偏好反馈数量的影响

本文还在 Breakout、SpaceInvaders 和 Seaquest 三个游戏上测试了不同偏好反馈数量（1000 对、3000 对、5000 对）对 PreIL 性能的影响。

总体趋势显示：偏好反馈越多，性能越好。从 1000 对增加到 5000 对，游戏的平均性能有明显提升。Seaquest 上出现了一个值得注意的现象：三条曲线都在训练中期达到峰值，之后出现不同程度的退化。其中 5000 对偏好和 3000 对偏好最终性能相对接近，1000 对偏好的峰值较低。这可能是因为 Seaquest 的策略空间较复杂，过多的偏好数据反而让判别器过拟合到当前策略分布，策略更新后判别器未能及时跟上，导致性能回落。综上本文的主实验在训练中采用 5000 作为偏好反馈数量的超参数。

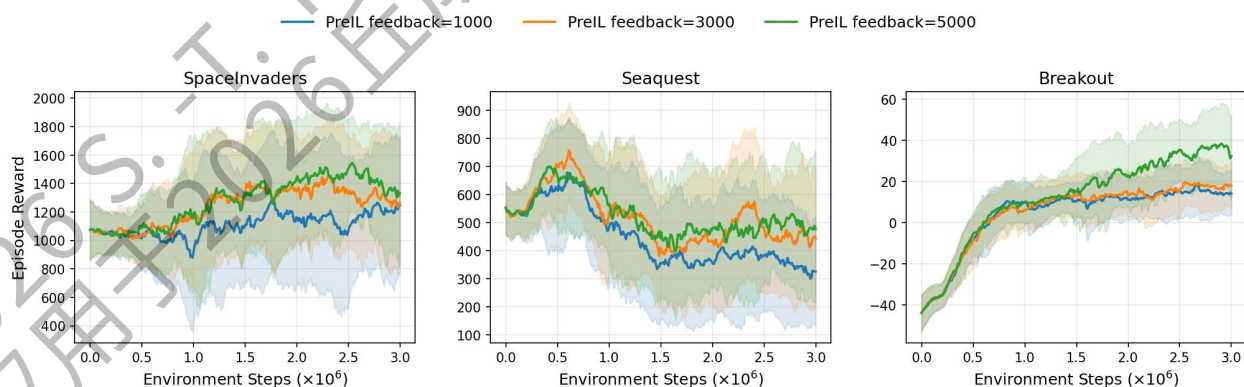


图 5-3 不同偏好反馈数量下 PreIL 的性能对比

5.4. 奖励恢复失败的可视化分析

为了更直观地理解为什么基于奖励恢复的方法效果不好，本文做了可视化分析。本文按照 Christiano 等人[6]的方法从人类偏好恢复出一个奖励函数，再用 DQN 用它训练 3M 步。然后在策略训练的每一步，分别用真实奖励函数和恢复的奖励函数来计算当前策略产生的 episode 奖励。

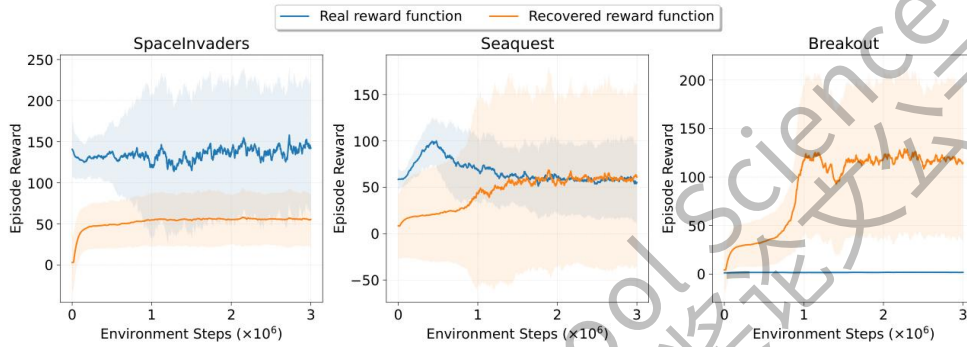


图 5-4 奖励恢复失败的可视化分析

本文发现了一个有意思的现象：用恢复的奖励函数算出来的 episode 奖励（橙线）随着训练一直在上升，说明在恢复奖励的坐标系下，策略确实在变好；但是用真实奖励函数算出来的 episode 奖励（蓝线）几乎没涨，有的游戏上甚至还下降了。这说明恢复的奖励函数并没有把策略引向真正好的方向。

这个现象说明了奖励恢复方法的根本问题：偏好比较只能提供有限的约束，有很多不同的奖励函数都能完美拟合训练集中的偏好对，但这些奖励函数在没见过的状态上可能给出任意的奖励值。当策略优化到训练分布之外的状态时，恢复的奖励函数给出的值和真实奖励可能完全没关系。这也验证了本文在理论分析中说的分布外外推问题。

相比之下，PreIL 没有这个问题。因为判别器是和策略一起联合优化的——每次策略更新后，新的交互数据会被加入缓冲区，新的偏好查询基于最新的策略分布，判别器也跟着更新。这种端到端的联合优化让判别器总能适应当前的策略分布。

第 6 章 讨论与结论

本文提出了偏好模仿学习（PreIL）方法，它的核心想法是绕开显式奖励恢复这个中间环节，直接通过时序判别器从成对偏好比较中学习策略。本文分析了 Bradley-Terry 模型的三个问题，并基于全局时序品质假设的三条基本性质说明了为什么绕开显式奖励恢复是有道理的。在六个 Atari 游戏上的实验验证了 PreIL 的有效性，对比实验还说明性能提升主要来自模仿学习框架本身。

本研究还有一些不足：

1. 实验用的是模拟的人类偏好（基于真实累计奖励），不是真实的人类反馈。
2. 轨迹子序列的长度 n 是个超参数，需要针对不同的环境调整。未来可以进一步研究的方向包括：在真实人类反馈上验证方法、探索自适应选择 n 步长度的方法、把方法扩展到更复杂的连续控制任务。

近年来在 LLM 对齐领域，DPO 等方法也开始探索跳过奖励模型直接优化策略的方向。但 DPO 基于概率重参数化，基于 RLHD 的精确重参数化，将偏好比较直接转化为策略优化目标，适用于离散 token 生成；而本文的 PreIL 基于生成对抗模仿学习框架，通过 LSTM 时序判别器处理离散动作序列，适用于 Atari 等离散控制环境。两者技术路线不同，面向的问题也不同。

总的来说，本文认为直接从偏好学习策略是一个有意义的研究方向。与其花很大力气去恢复一个可能不可靠的奖励函数，不如尊重人类偏好的本质特点，直接在策略层面进行偏好驱动优化。这不仅在思路更直接，实践中效果也更好。

第 7 章 参考文献

- [1] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529-533.
- [2] Levine S, Finn C, Darrell T, et al. End-to-end training of deep visuomotor policies[J]. Journal of Machine Learning Research, 2016, 17(1): 1334-1373.
- [3] Kendall A, Hawke J, Janz D, et al. Learning to drive in a day[C]//2019 International Conference on Robotics and Automation (ICRA). IEEE, 2019: 8248-8254.
- [4] Krakovna V, Orseau L, Ramanauskas S, et al. Specification gaming: the flip side of AI ingenuity[J]. Nature Machine Intelligence, 2020, 2(11): 721-724.
- [5] Ho J, Ermon S. Generative adversarial imitation learning[C]//Advances in Neural Information Processing Systems. 2016: 4565-4573.
- [6] Christiano P, Leike J, Brown T, et al. Deep reinforcement learning from human preferences[C]//Advances in Neural Information Processing Systems. 2017: 4299-4307.
- [7] Bradley R A, Terry M E. Rank analysis of incomplete block designs: I. The method of paired comparisons[J]. Biometrika, 1952, 39(3/4): 324-345.
- [8] Seo M, Lee S, Liu X, et al. PEBBLE: Feedback-efficient reinforcement learning via bootstrapping error ensembles[C]//International Conference on Machine Learning. PMLR, 2021: 9450-9461.
- [9] Lee K, Lee H, Shin J, et al. Better-than-demonstration imitation learning from preferences[C]//International Conference on Learning Representations. 2021.
- [10] Pomerleau D A. Efficient training of artificial neural networks for autonomous navigation[J]. Neural Computation, 1991, 3(1): 88-97.
- [11] Ziebart B D, Maas A L, Bagnell J A, et al. Maximum entropy inverse reinforcement learning[C]//Proceedings of the 23rd National Conference on Artificial Intelligence. 2008: 1433-1438.
- [12] Finn C, Levine S, Abbeel P. Guided cost learning: Deep inverse optimal control via policy optimization[C]//International Conference on Machine Learning. PMLR, 2016: 49-58.
- [13] Fu J, Luo K, Levine S. Learning robust rewards with adversarial inverse reinforcement learning[J]. arXiv preprint arXiv:1710.11248, 2017.
- [14] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in Neural Information Processing Systems. 2014: 2672-2680.
- [15] Ghasemipour S K S, Zemel R, Gu S. A divergence minimization perspective on imitation learning methods[C]//Conference on Robot Learning. PMLR, 2020: 1259-1277.
- [16] Akrou R, Schoenauer M, Sebag M. Preference-based policy learning[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Berlin, Heidelberg, 2011: 12-27.
- [17] Bıyık E, Hu R, Palan M, et al. Batch active preference-based learning of reward functions[C]//Conference on Robot Learning. PMLR, 2020: 261-272.

- [18] Thurstone L L. A law of comparative judgment[J]. Psychological Review, 1927, 34(4): 273.
- [19] Ng A Y, Russell S J. Algorithms for inverse reinforcement learning[C]//Proceedings of the 17th International Conference on Machine Learning. 2000: 663-670.
- [20] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.
- [21] Watkins C J C H, Dayan P. Q-learning[J]. Machine Learning, 1992, 8(3): 279-292.
- [22] Hernandez-Garcia J F, Sutton R S. Understanding multi-step deep reinforcement learning: A systematic study of the DQN target[J]. arXiv preprint arXiv:1901.07510, 2019.
- [23] Horgan D, Quan J, Budden D, et al. Distributed prioritized experience replay[C]//International Conference on Learning Representations. 2018.
- [24] Paszke A, Gross S, Massa F, et al. PyTorch: An imperative style, high-performance deep learning library[C]//Advances in Neural Information Processing Systems. 2019: 8024-8035.
- [25] Bellemare M G, Naddaf Y, Veness J, et al. The arcade learning environment: An evaluation platform for general agents[J]. Journal of Artificial Intelligence Research, 2013, 47: 253-279.

附 录

一、实验超参数设置

下表列出了所有实验中使用的超参数及其取值。所有方法的策略网络结构和训练超参数保持一致，以保证公平比较。

超参数	取值
策略网络结构	FC(256)-FC(256)-FC(256)
策略学习率	0.0003
策略 minibatch 大小	128
回放记忆大小	1,000,000
初始探索率	1.0
最终探索率	0.05
探索比例	0.3
总训练步数	3,000,000
策略更新间隔	2,000
目标网络更新间隔	10,000
开始学习前的步数	20,000
策略学习 epoch 数	50
帧跳变	4
最大 No-op 数	30
n 步 DQN 的 n	5
奖励函数/判别器网络结构	Pref(NS)DQN: FC(256)-FC(256)-FC(256); PreIL: LSTM(256)
奖励函数/判别器学习率	0.0003
奖励函数/判别器 minibatch 大小	128
奖励函数/判别器学习 epoch 数	50
人类反馈片段长度	50
人类偏好查询总次数	5,000
人类查询会话间隔步数	20,000
每次会话反馈数	1,000

表 A-1 实验超参数一览表

二、实验环境

所有实验在同一台服务器上完成，硬件配置如下：

项目	配置
硬件配置	32 核心，128 GB，NVIDIA RTX 4060 Ti 16GB
操作系统	Linux
深度学习框架	PyTorch
实验总时长	约 15 天

表 A-2 实验硬件环境配置

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

致 谢

本研究是在指导老师的悉心指导下完成的。从选题方向的确定，研究思路的梳理，到实验方案的设计和论文写作的修改，每一个环节都凝聚着指导老师的心血和智慧。指导老师严谨的治学态度、深厚的学术功底和耐心的指导方式，使作者受益匪浅。在此向指导老师表示最诚挚的感谢。

一、研究背景与师生关系说明

本研究项目由指导老师提出选题方向，学生作为主要研究者完成具体研究工作。指导老师为作者的中学信息科技教师，从事强化学习和人工智能领域的研究工作，为本研究提供了学术指导和思路启发。指导为无偿性质，不存在任何商业或利益关系。

本研究的核心思想——偏好模仿学习（PreIL）——是在指导老师指导下，由学生主导完成研究和实现的。学生在研究过程中提出了全局时序品质假设（GTQ）和三重困境分析等理论见解，并完成了算法设计、代码实现和实验验证等全部工作。

二、研究过程与分工说明

本研究的整体过程和具体分工如下：

1. 选题阶段：指导老师提出偏好强化学习的研究方向，学生在广泛阅读文献的基础上，将研究问题聚焦于“能否绕开显式奖励恢复，直接从偏好学习策略”这一核心问题。选题过程中学生完成了约 50 篇相关文献的阅读和综述，指导老师提供了文献推荐和研究思路的指导。

2. 理论分析阶段：学生陈雨涵提出了全局时序品质假设（GTQ）的概念及其三条性质，系统分析了 Bradley-Terry 模型的重重困境。指导老师对理论框架的逻辑严密性进行了指导，帮助学生完善了论证结构。

3. 算法设计阶段：学生陈雨涵与阮邦城基于理论分析设计了 PreIL 算法框架，包括时序判别器的架构设计和 n 步策略优化机制。指导老师对算法的可行性和技术细节提出了修改建议。

4. 代码实现阶段：学生林子博主要完成了核心代码实现，包括基于 PyTorch 的判别器实现、DQN 变体的修改、偏好模型的设计等。指导老师对代码结构和调试提供了建议。

5. 实验阶段：学生陈雨涵、林子博和阮邦城共同设计了实验方案，主要由林子博完成了所有实验的运行和数据整理工作。实验在六个 Atari 游戏上进行，每个游戏 5 个随机种子。指导老师辅助申请了计算资源用于实验，并对实验结果的分析和解读提供了指导。

6. 论文撰写阶段：学生陈雨涵完成了论文初稿的撰写，阮邦城负责程序核心算法的梳理和撰写，林子博负责实验数据部分的整理和撰写，最后指导老师对论文结构、逻辑论证和语言表达和格式规范进行了细致的修改和指导。论文前后经历了多轮修改和完善。

三、遇到的困难与解决过程

在研究过程中，作者遇到了以下主要困难：

第一，理论构建的困难。作者本身为信息竞赛学生，对算法有系统化的认知，并具备一定编程能力。但是对于人工智能的认知基础较为薄弱。经过了 2 个多月的系统学习，逐步构建了强化学习方向的基础架构。但如何从哲学层面论证奖励恢复范式的缺陷，是研究初期面临的最大挑战。最初，作者只有一些模糊的直觉——“人类偏好不是标量可加的，奖励函数恢复的不准确怎么办？”，但如何将这一直觉系统化、理论化，经历了一个漫长的思考过程。通过反复阅读偏好学习、认知科学和价值理论的相关文献，作者逐渐形成了 GTQ 基本性质的雏形，并最终发展出三重困境分析框架。在这一过程中，指导老师推荐的认知科学相关文献对作者启发很大。

第二，实验设备获取和实验开展的困难。强化学习必须依赖于强大的硬件设备，论文作者虽然已有相关的理论积累和研究方向，但是没有实验所需的设备和环境。在学校的帮助下，申请硬件资源用于开展实验。而在实验过程中，Atari 游戏的强化学习训练本身具有较高的不稳定性，而偏好学习引入了额外的不确定性，使得调试更加困难。在实验初期，PreIL 的性能经常出现大幅波动，甚至完全不收敛。通过系统地调整学习率、判别器更新频率、奖励尺度等超参数，并引入梯度裁剪等稳定技术，作者最终找到了一组稳定有效的超参数配置。

第三，论文写作的困难。如何将理论分析和实验结果有机地组织成一篇结构完整、逻辑清晰的学术论文，对作者而言也是一个挑战。在指导老师的建议下，作者学习了多篇顶会论文的写作结构，经过多次修改，最终形成了现在的论文框架。

四、AI 工具使用说明

在本研究过程中，作者使用了以下 AI 工具作为辅助：

1. 文献检索与管理：使用了西南大学图书馆的数据库进行论文检索和学习。借助 AI 工具 Workbuddy 辅助文献整理和阅读，但所有文献的阅读和理解均由作者本人完成。
2. 理论学习：通过网页版 AI 工具（如 DeepSeek）和智能体 Workbuddy 进行基础理论知识的梳理和学习。
3. 代码辅助：在代码编写过程中，使用了 AI 工具 Workbuddy 辅助编写代码。核心算法的设计和实现由作者主导完成，AI 工具仅提供语法层面的辅助。
4. 语言润色：在论文撰写的后期阶段，使用了语言模型对英文摘要进行语法检查和表达润色。论文的核心内容、论证逻辑和学术观点均由作者原创，AI 工具 Workbuddy 仅辅助语言层面的优化。

本研究的核心思想、理论分析、算法设计、实验实现和论文主体内容均由作者主导完成，AI 工具的使用仅限于辅助性工作，未对研究的原创性构成影响。