ArenaRL：基于锦标赛相对排名的开放式智能体强化学习（ICML2026）

张凡瑞 | 2026年7月3日

目录

- **背景介绍：**

  开放式任务评估与RL训练

- **方法介绍：**

  ArenaRL算法流程与benchmark构建

- **结果介绍：**

  实验结果与使用

背景介绍

确定性任务

Weng earns $12 an hour for baby-sitting. Yesterday, she just did 50 minutes of babysitting. How much did she earn?

Weng earns 12/60 = $<<12/60 =0.2>>0.2 per minute.
Working 50 minutes, she earned 0.2 x 50 = $<<0.2*50=10>>10.
10

可验证的奖励

开放式任务

我从上海徐汇出发，周末想找一条人，少有遮阴可推婴儿车的城市绿道。途中顺路找一家低糖面包店买点心，再去一家可以预约包间的本帮菜馆诊断路线，希望尽量不走台阶。傍晚前回到地铁站。

已深度搜索 38秒 >

🚨 必读事项

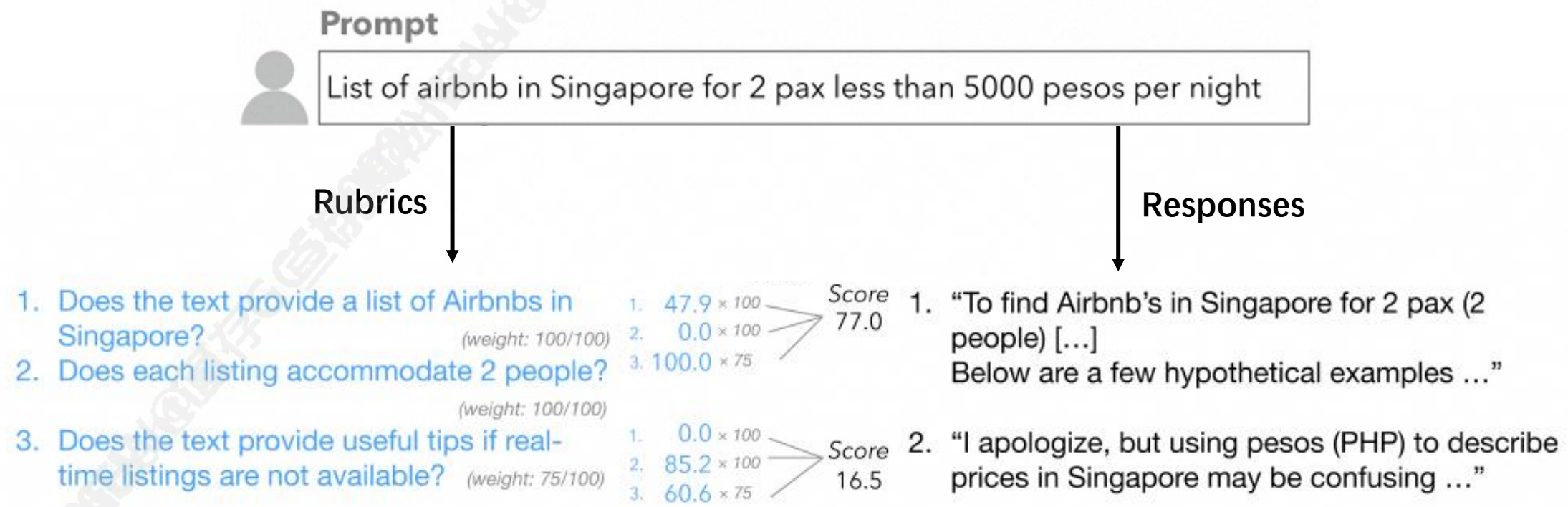
⚠️ 绿道人流与设施信息有限

- 徐汇滨江绿道为徐汇区主要城市绿道，适合推婴儿车，但周末人流密度、遮阴情况、无障碍设施及台阶分布暂无权威公开数据。
- 影响：高峰时段绿道可能人流较多，部分路段遮阴有限，存在少量台阶或坡道。
- 解决方案：建议您错峰出行（上午或傍晚），优先选择绿道主干道，避开高峰时段。如遇人流密集，可适当调整路线或缩短停留时间。

缺少唯一标准解

背景介绍

Rubrics评分

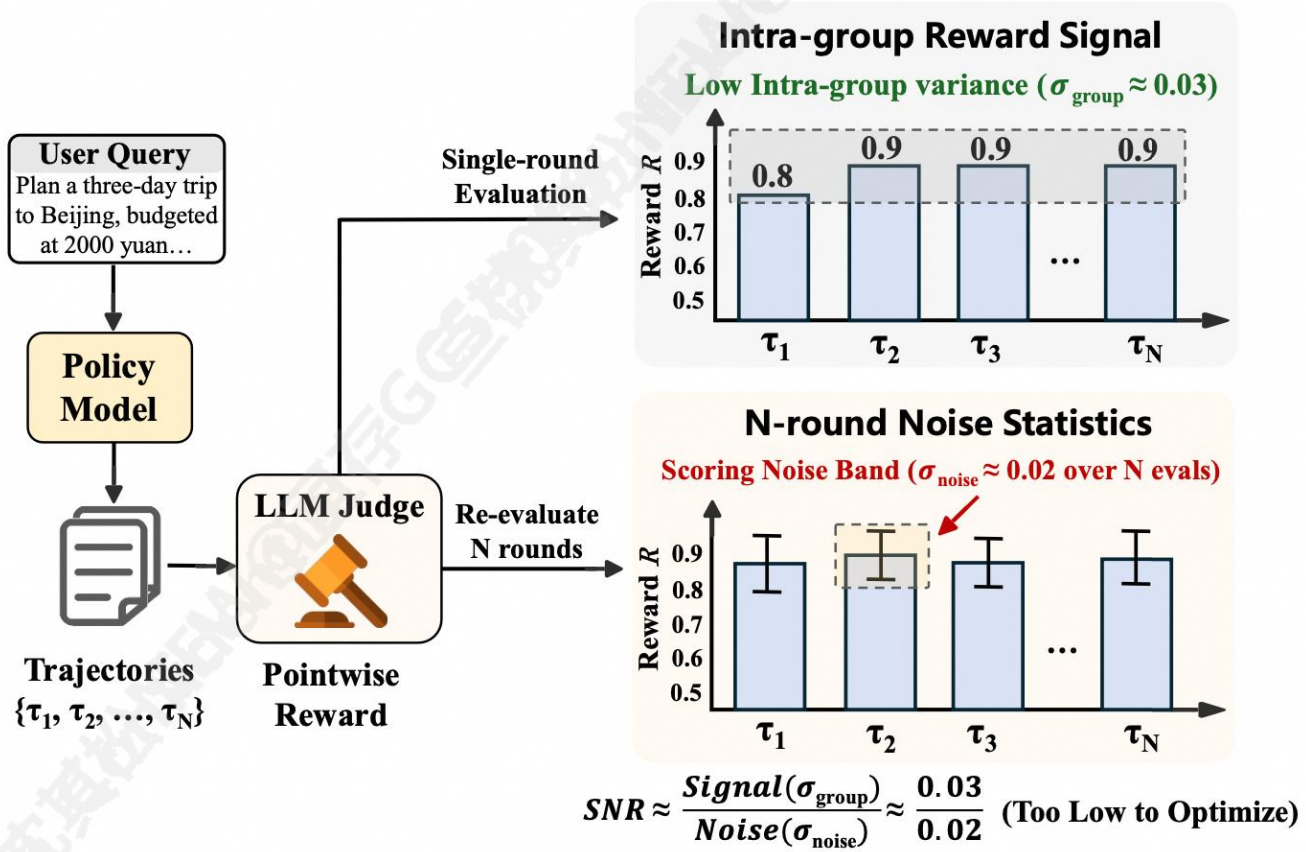


针对问题类型拆解为一系列评估标准，并基于pointwise打分，

得到每个response的奖励

背景介绍

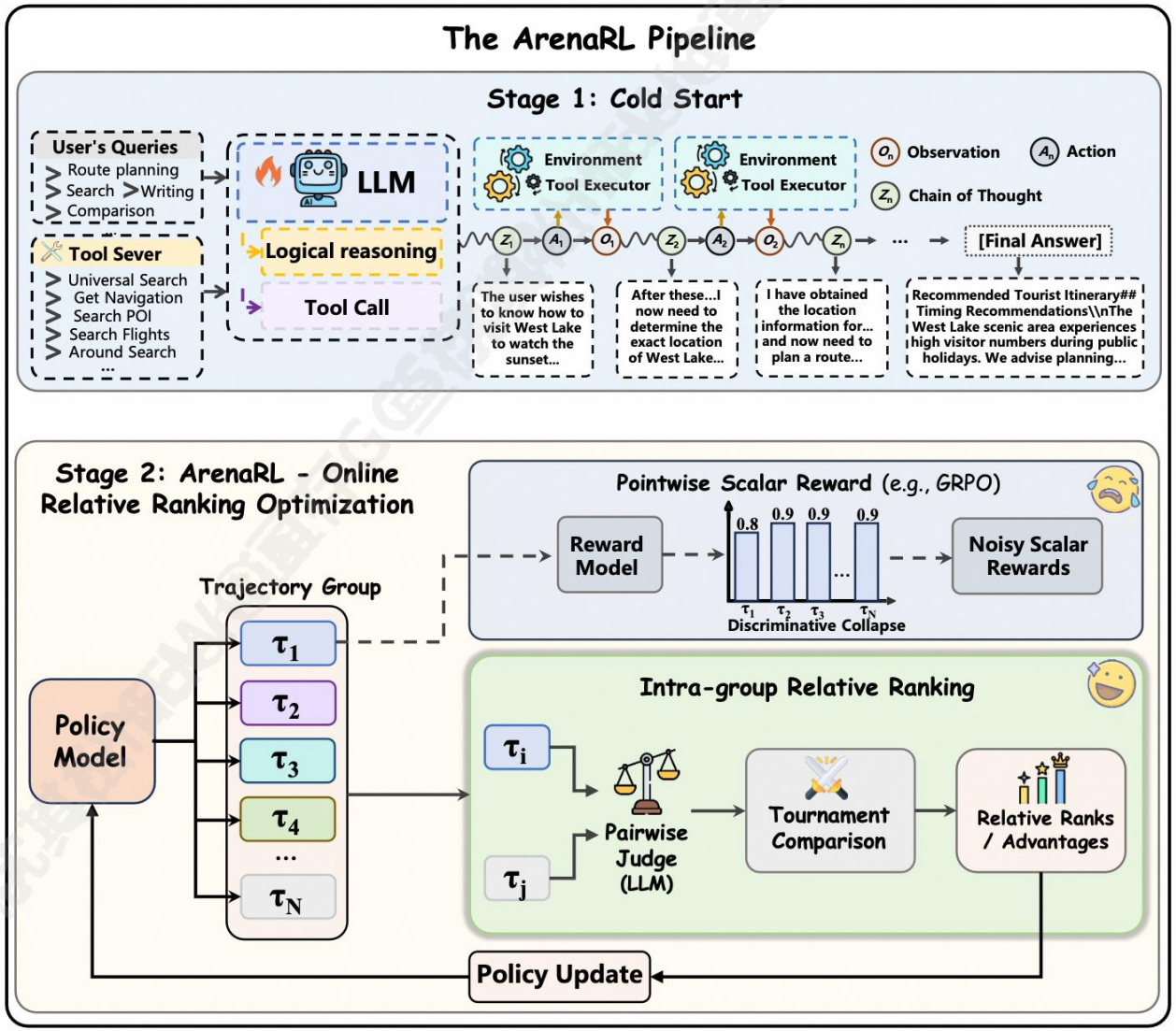
判别崩溃现象



- 随着策略模型的优化，组内生成轨迹的分布趋同，pointwise打分将奖励压缩在极窄范围内，难以区分轨迹优劣
- 奖励模型的噪声掩盖了微弱的组内质量差异信号，导致低信噪比
- pairwise对比评估相对pointwise打分天然具有更强的稳定性

方法介绍

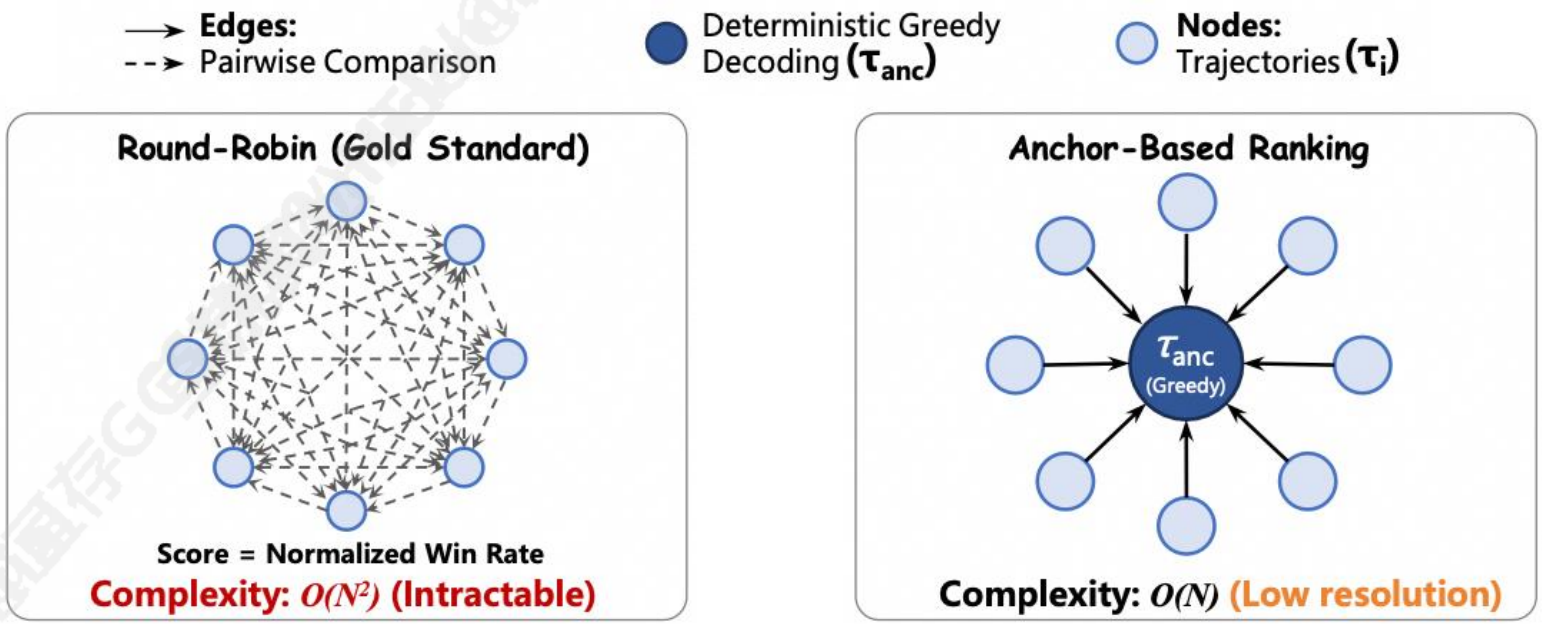
范式转移：从不稳定的pointwise打分转向基于pairwise评估的组内相对排名



- **过程感知成对评估：**构建多层次rubrics，不仅评估结果，更审查思维链逻辑和工具调用的有效性
- **算法概述：**构建一个组内竞技场，将组内轨迹进行两两对抗，得到相对质量排名，并以此计算奖励
- **核心需求：**在高精度排名和计算复杂度之间寻求最佳平衡

方法介绍

锦标赛模式1 & 2

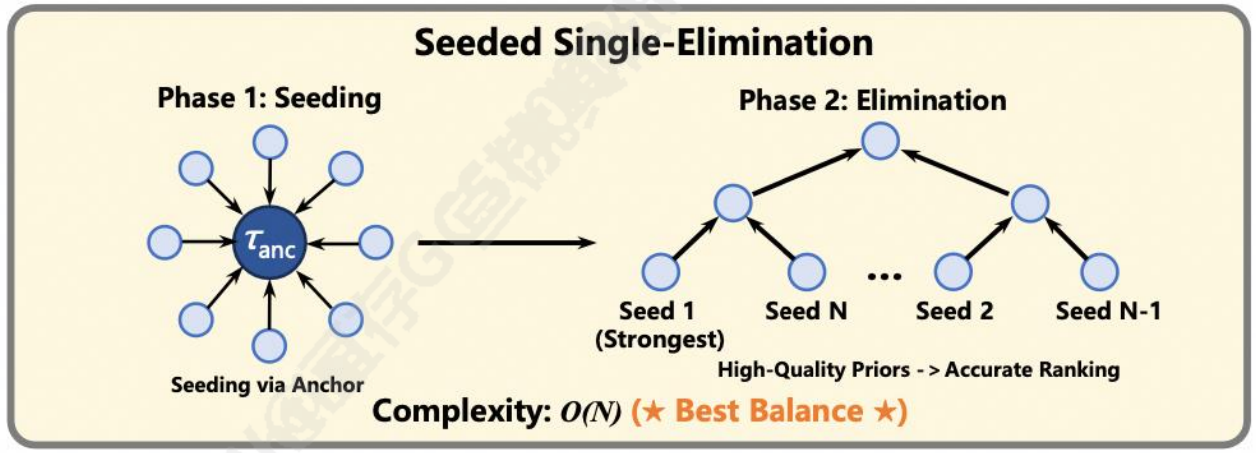


- **循环赛 (Round-Robin)：** 精度最高但计算复杂度$O(N^2)$

- **锚点排名 (Anchor-Based Ranking)：** 效率高，计算复杂度$O(N)$，但分辨率低。

方法介绍

锦标赛模式3



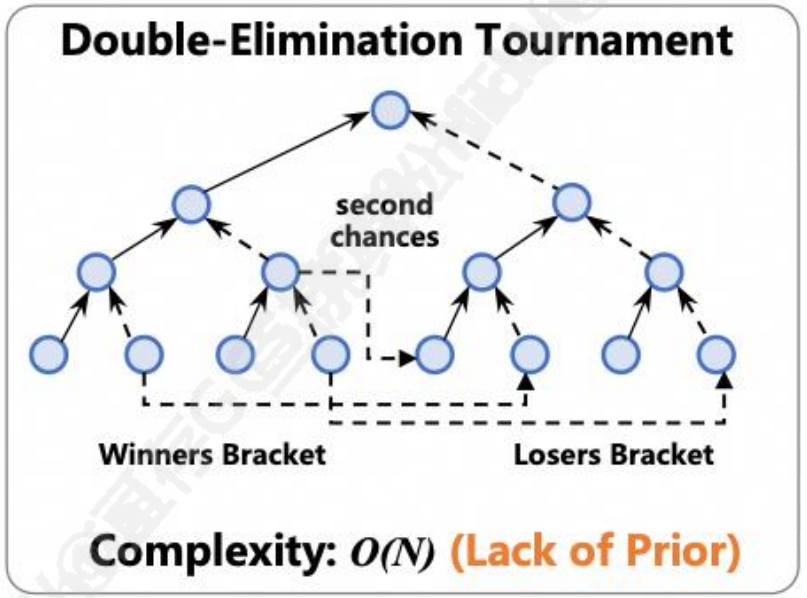
- **种子单败淘汰制 (Seeded Single-Elimination)：** 计算复杂度$O(N)$，兼顾效率与精度。

算法流程

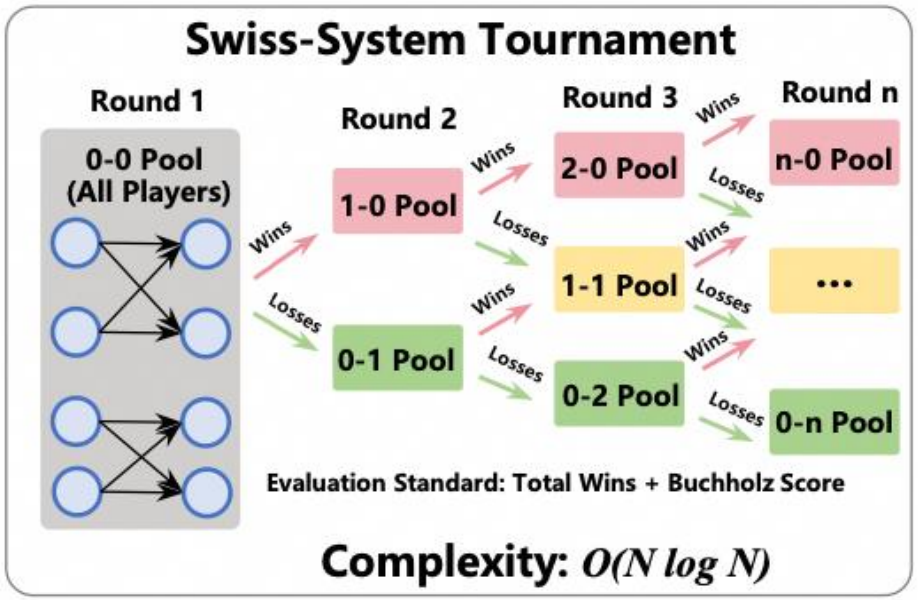
- **锚点预排序：** 使用贪婪解码轨迹作为质量锚点，建立初始种子分布。
- **单败淘汰赛：** 采用蛇形分组模式防止顶级轨迹过早碰撞，之后按照二叉树模式进行逐层竞争
- **排名分配：** 根据生存深度和累计评分确定最终排名。

方法介绍

锦标赛模式4 & 5



- **双败淘汰赛 (Double-Elimination)：**计算复杂度$O(N)$，缺乏初始先验，导致性能受限。



- **瑞士制 (Swiss-System)：**计算复杂度$O(NlogN)$，缺乏初始先验，比赛深度不足。

方法介绍

策略优化

- **Reward计算**

$$r_i = 1 - \frac{\text{Rank}(\tau_i)}{N-1}.$$

将锦标赛得到的组内排名线性插值为[0,1]范围内的奖励

- **优势计算**

$$A_i = \frac{r_i - \mu_r}{\sigma_r + \epsilon},$$

计算组内均值和标准差，并计算各轨迹最终优势

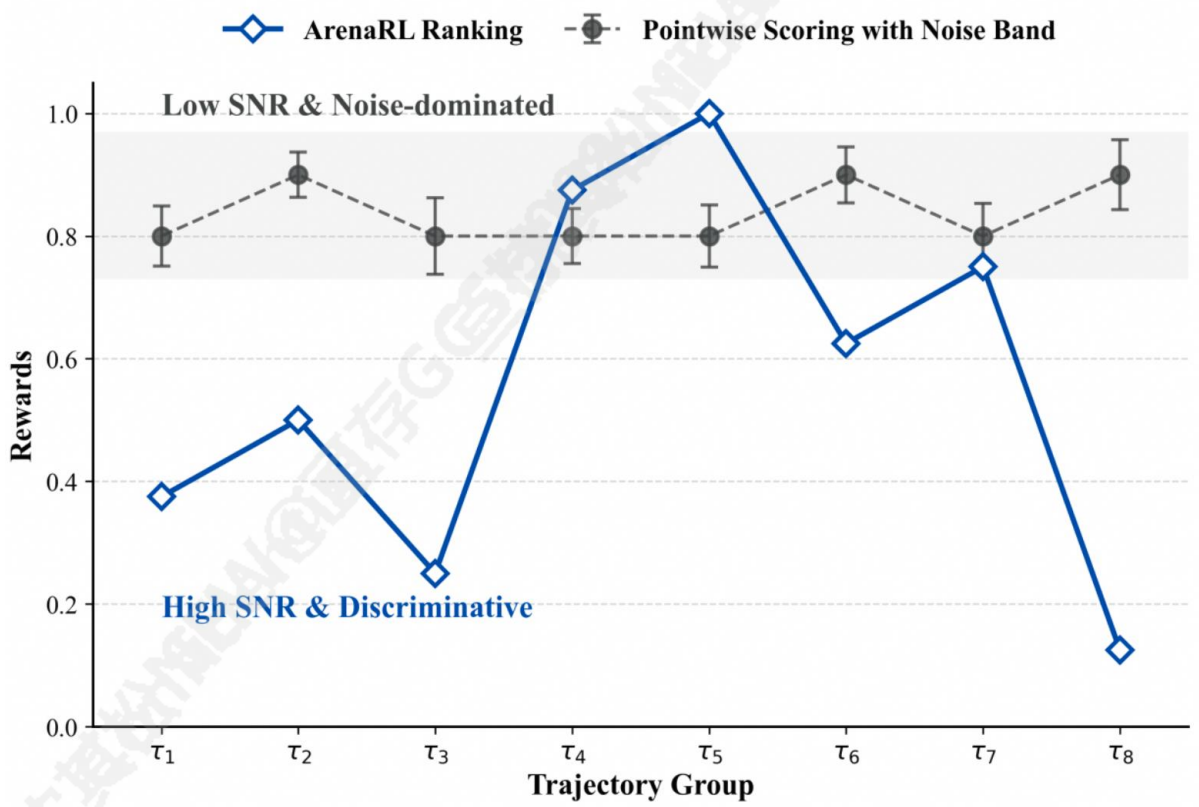
- **优化目标**

$$\mathcal{L}_{\text{ArenaRL}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\mathcal{G}\sim\pi_\theta}\left[\frac{1}{N}\sum_{i=1}^{N}\left(\min\left(\frac{\pi_\theta(\tau_i \mid x)}{\pi_{\text{old}}(\tau_i \mid x)}A_i,\ \text{clip}\left(\frac{\pi_\theta(\tau_i \mid x)}{\pi_{\text{old}}(\tau_i \mid x)}, 1-\epsilon, 1+\epsilon\right)A_i\right) - \beta\,\mathbb{D}_{\text{KL}}\left(\pi_\theta(\cdot \mid x)\,\|\,\pi_{\text{ref}}(\cdot \mid x)\right)\right)\right].$$

对Policy Model进行优化

方法介绍

组内排序范式的奖励可视化

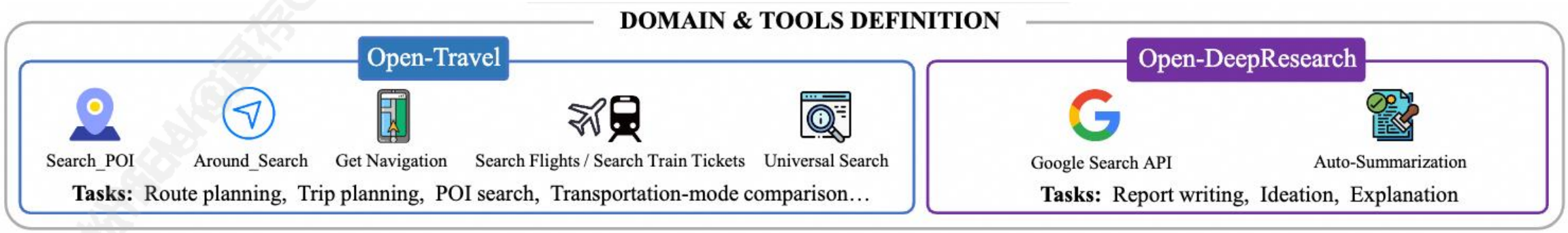


- 将组内轨迹的细微差别映射为高分辨率离散排名奖励
- Pairwise评估自然过滤point-wise打分噪声，显著提高了奖励信噪比

方法介绍

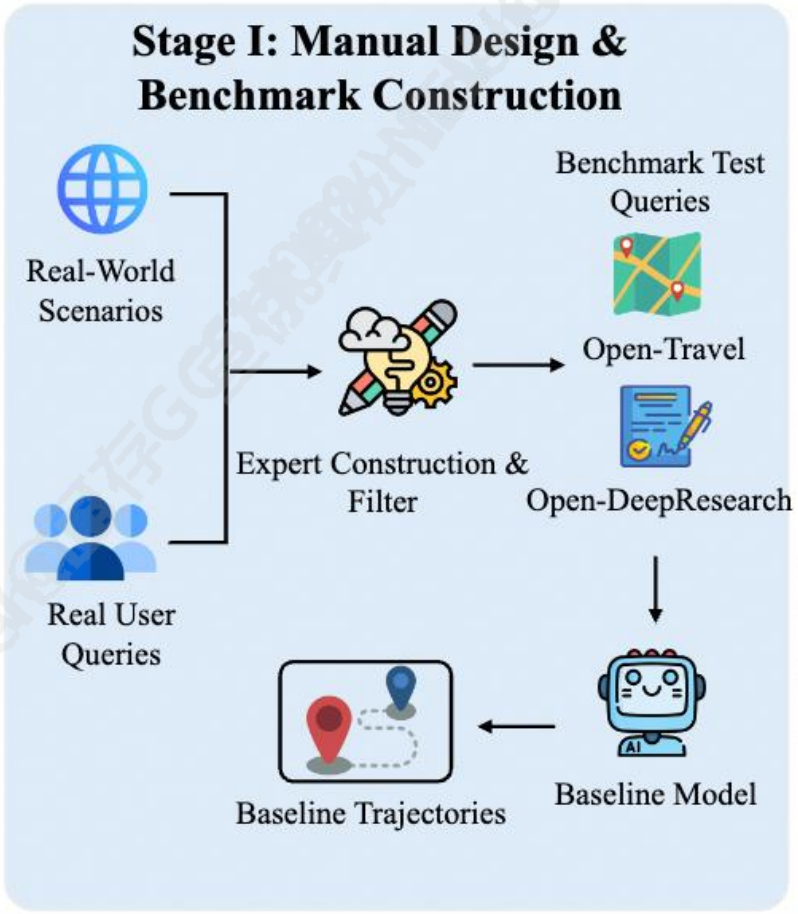
Benchmark简介

- **Open-Travel：**要求Agent执行五种类型的行程规划子任务，强调多约束推理、工具协调和个性化偏好对齐。
- **Open-DeepResearch：**评估Agent在多轮网络搜索和信息合成方面的熟练程度，以生成开放式响应。



方法介绍

Benchmark构建Stage 1

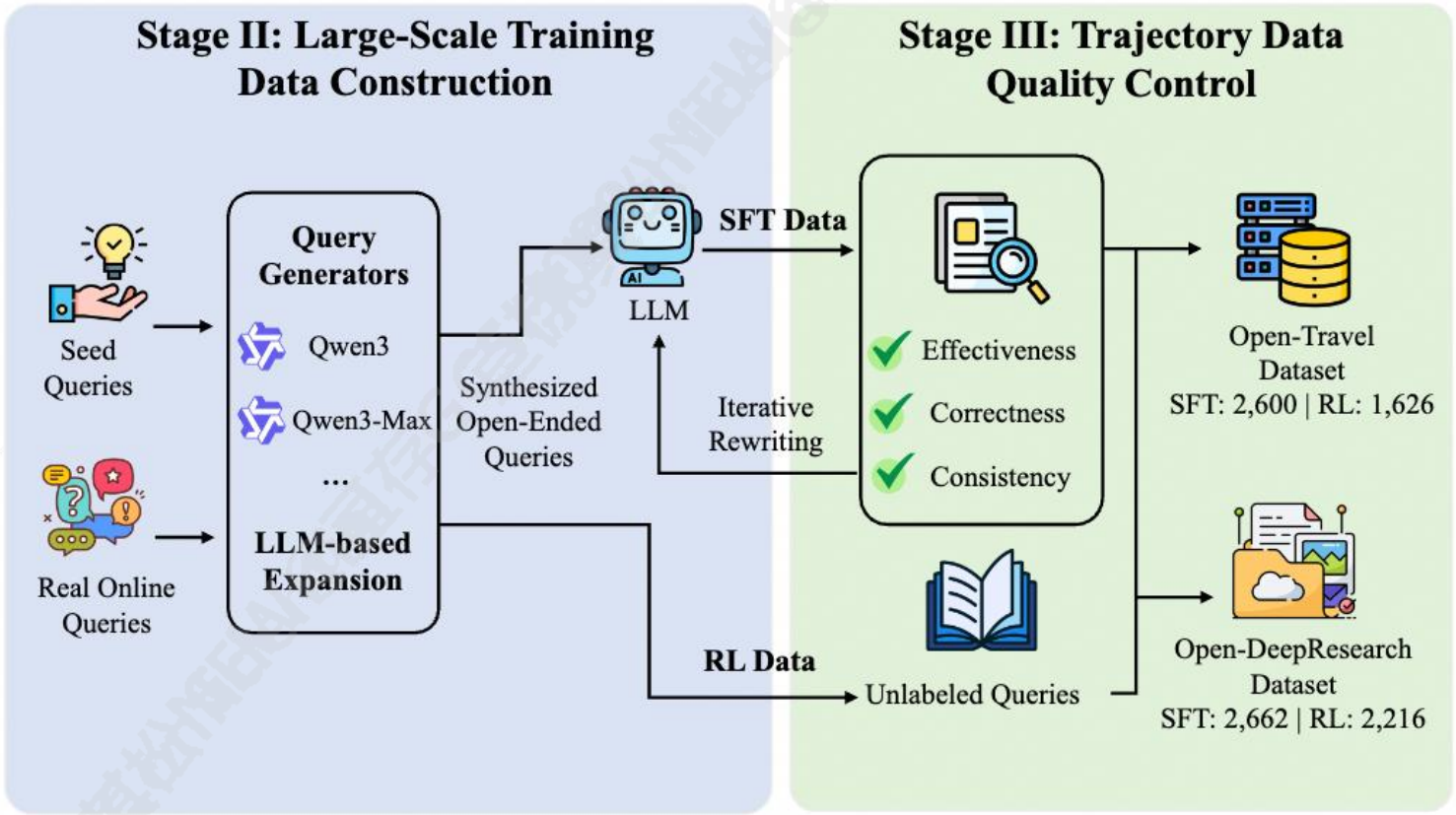


- 从真实世界的应用场景收集种子query，并由标注专家进行过滤和修正

- 抽取部分query作为基准测试集，采用Claude-Sonnet-4生成完整的工具使用轨迹和相应的开放式答案，作为后续两两比较和胜率评估的参考基线

方法介绍

Benchmark构建Stage 2 & 3



Stage 2:

- 使用多种类型LLM针对各类任务合成大规模query，并使用LLM合成相应的轨迹，作为SFT数据
- 未标注的query作为RL训练数据

Stage 3:

- 使用LLM质量检测模块对合成轨迹进行检查，并对不合格的轨迹进行重写

评估方式：使用两个LLM作为评估模型，将被测试模型的轨迹与基准中Stage 1构建的参考轨迹进行pairwise对比评估，并计算各个评估标准的平均胜率

结果介绍

五种锦标赛拓扑的实验对比

Topology	Comparison Cost	Open-Travel					Mean
		Direction	Search	Compare	1-Day	M-Day	
SFT	-	10.6	29.7	14.1	20.4	7.1	16.4
Anchor-Based Ranking	$N-1$	18.0	41.3	30.9	31.1	17.6	27.8
Swiss-System	$N \log N$	20.9	43.0	27.9	38.6	11.1	28.3
Double-Elimination	$2N-2$	12.6	52.4	**33.7**	**39.9**	12.3	30.2
Seeded Single-Elimination	$2N-2$	16.9	**69.9**	22.9	34.9	18.1	32.5
Round-Robin (Upper Bound)	$N(N-1)/2$	**23.3**	66.3	23.6	32.1	**19.0**	**32.9**

实验表明，种子单败淘汰制 (Seeded Single-Elimination)在训练效率和模型性能之间的达到了较优平衡

结果介绍

Open-Travel和Open-DeepResearch基准的实验结果

Method	Open-Travel						Open-DeepResearch							
	Direction	Search	Compare	1-Day	M-Day	**Mean**	Frm.	Tool.	Cov.	Rel.	Acc.	Dep.	Cla.	**Mean (Val. %)**
Closed-source Models														
GPT-4o	2.4	5.0	3.1	1.6	0.7	2.6	5.1	24.4	21.0	9.1	12.5	2.3	10.8	12.2 (88.0)
Grok-4	17.0	21.3	9.7	24.7	11.3	16.8	33.7	36.8	43.4	36.8	39.2	36.1	17.5	34.8 (83.0)
Gemini-2.5-pro	8.6	12.5	7.4	11.9	12.4	10.6	15.8	19.0	17.9	32.6	28.3	45.7	38.6	28.3 (92.0)
Claude-3.7-Sonnet	18.6	59.6	14.7	43.6	21.3	31.6	10.1	13.5	22.5	23.6	19.7	27.0	17.4	19.1 (89.0)
Fine-tuning & RL														
SFT	10.6	29.7	14.1	20.4	7.1	16.4	14.1	20.3	23.4	14.1	15.6	15.6	14.1	16.7 (32.0)
GRPO	11.0	26.3	14.3	21.9	8.6	16.4	20.6	35.3	35.3	23.5	23.5	26.5	11.8	25.2 (17.0)
GSPO	10.0	30.6	13.1	21.1	11.4	17.2	23.8	33.3	40.5	16.7	21.4	31.0	9.5	25.2 (21.0)
ArenaRL	**32.1**	**66.1**	**31.7**	**58.0**	**21.0**	**41.8**	**62.6**	**77.3**	**78.8**	**57.1**	**55.6**	**57.1**	**61.6**	**64.3 (99.0)**

ArenaRL表现出了强大的性能，相对于基于pointwise打分的GRPO和GSPO算法，取得了极大的性能优势，并且超过了四个闭源模型

结果介绍

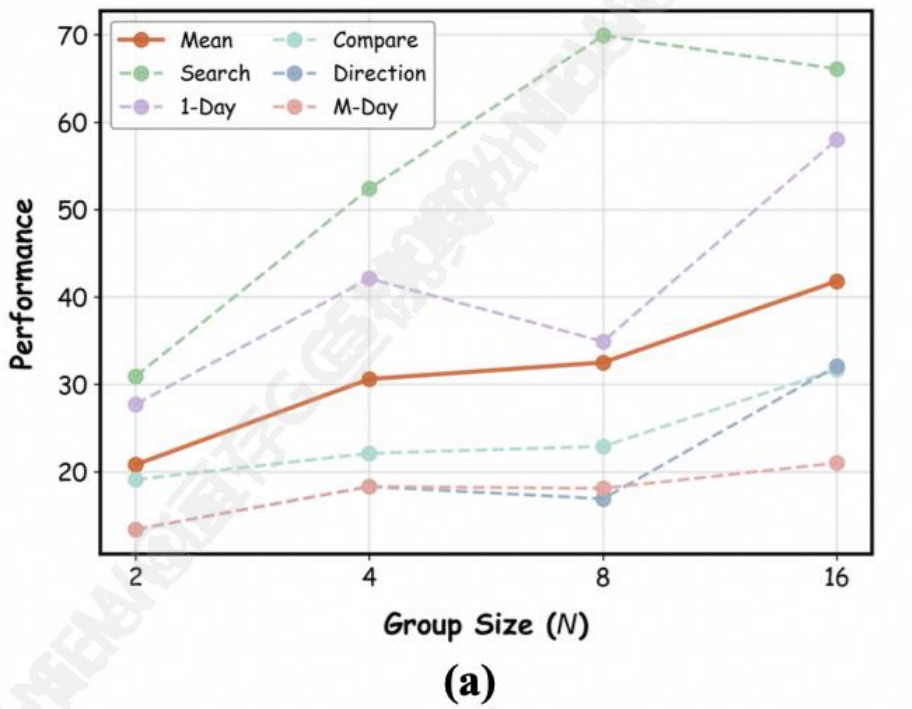
三个开放式写作基准上的实验结果

Method	WritingBench						HelloBench			LongBench	Mean
	WB-A	WB-B	WB-C	WB-D	WB-E	WB-F	QA	Summ.	Heur.	Quality	
Closed-source Models											
GPT-4o	67.90	66.34	68.56	69.95	70.70	72.17	81.03	84.26	89.14	90.43	76.05
Grok-4	80.32	78.65	79.75	81.46	81.19	80.92	88.42	85.58	94.65	96.52	84.75
Gemini-2.5-pro	80.89	80.39	82.49	84.33	83.53	82.61	85.67	82.43	93.79	98.69	85.48
Claude-3.7-Sonnet	68.36	66.53	68.70	70.34	71.42	71.47	80.81	74.68	95.82	98.34	76.65
Fine-tuning & RL											
SFT	70.71	69.36	67.88	63.72	69.69	70.64	78.44	63.42	82.35	85.52	72.17
GRPO	71.62	71.18	68.67	66.84	72.56	70.33	79.09	64.94	83.76	86.96	73.60
GSPO	71.56	70.70	68.87	66.08	72.29	69.83	**80.06**	63.97	81.75	85.21	73.03
ArenaRL	**78.14**	**77.70**	**77.58**	**75.02**	**79.35**	**77.16**	79.11	**73.82**	**91.33**	**93.78**	**80.30**

ArenaRL系统地增强了模型的推理和表达能力，证明了其对于除工具增强Agent之外的广泛的开放式生成任务的有效性

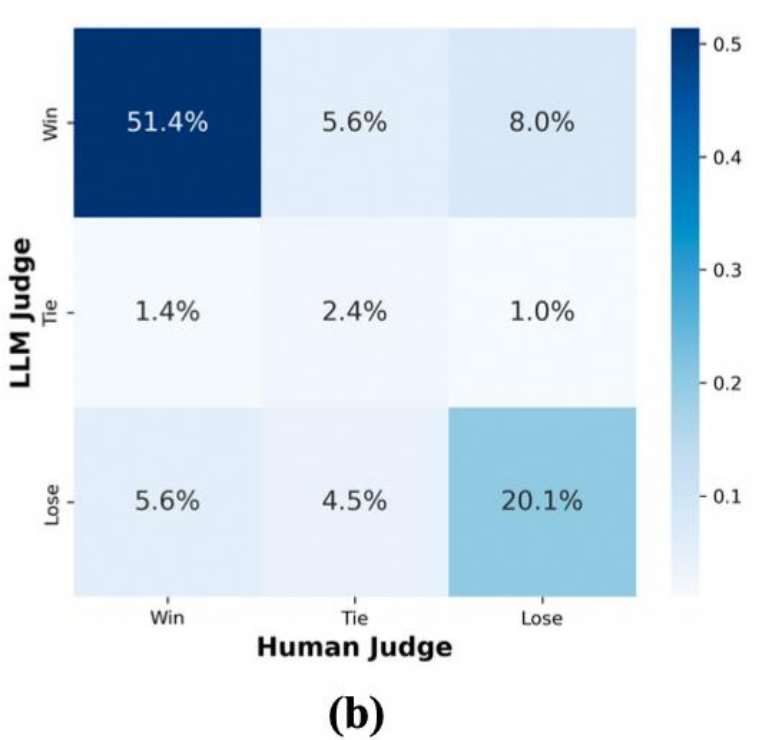
结果介绍

进一步分析



(a)

- **Group size N的影响：**随着Group size的增加，模型性能有明显的持续改善

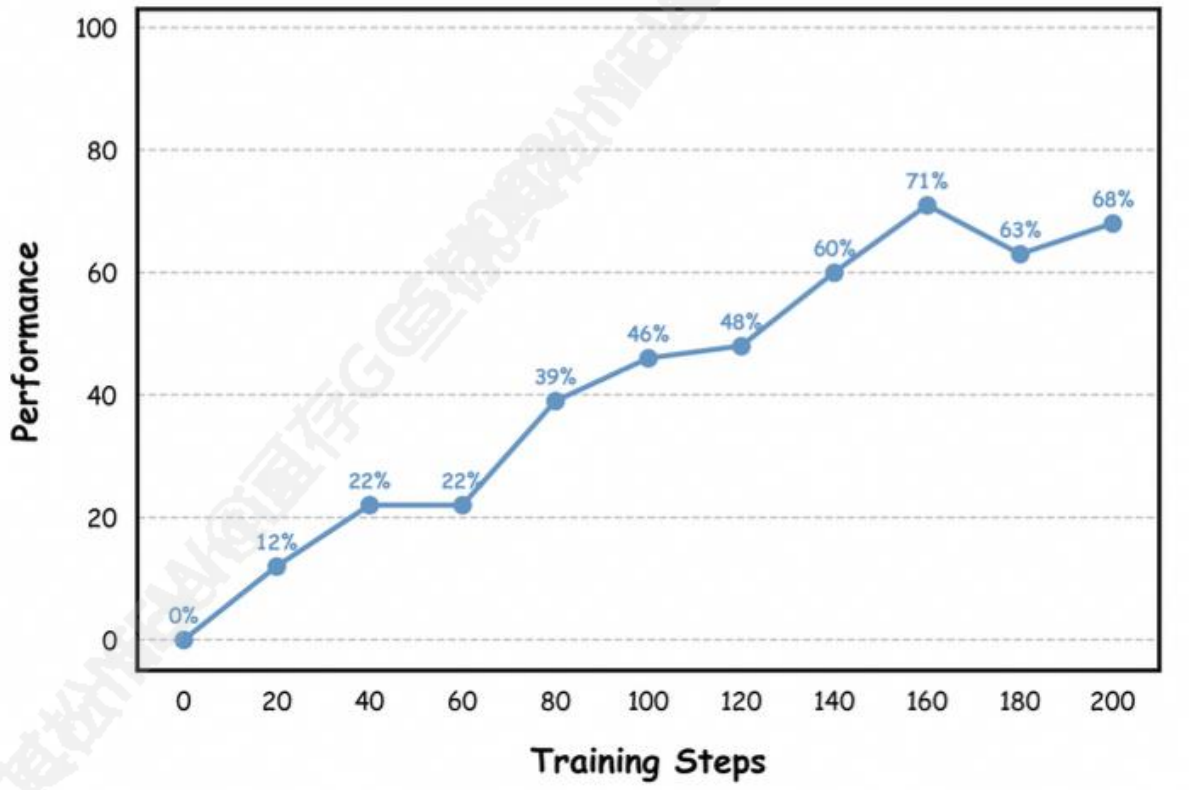


(b)

- **LLM与人工评估的一致性：**总体一致率为73.9%，表明ArenaRL的优化方向与人类偏好相贴合

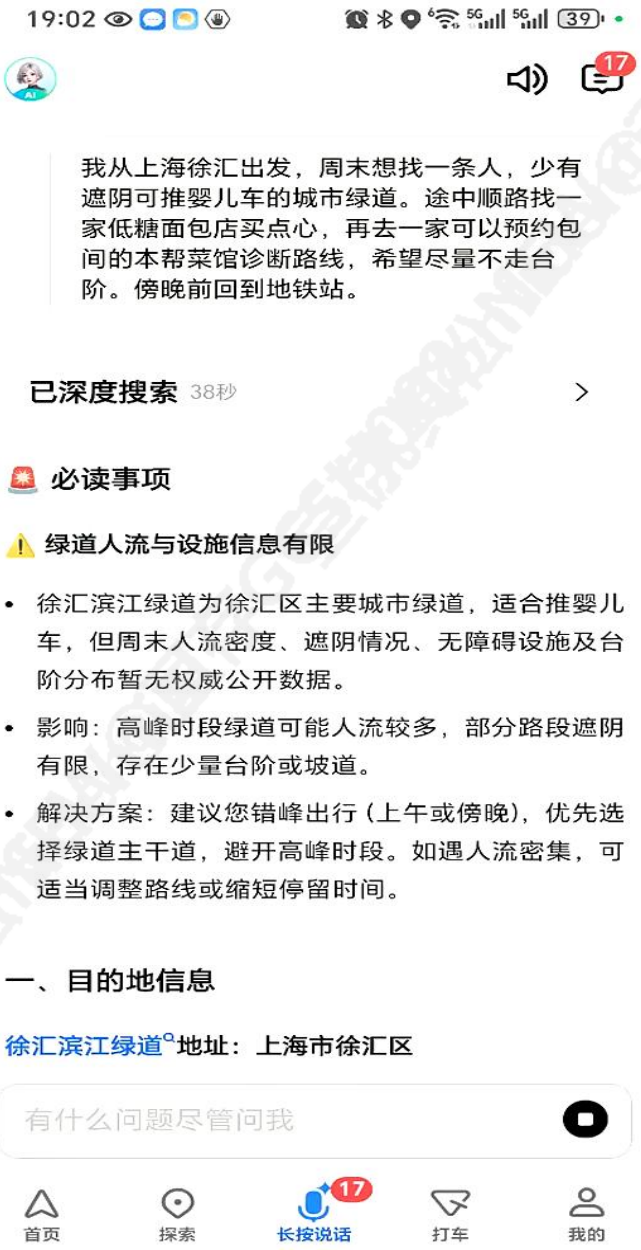
结果介绍

无冷启动直接RL训练的性能



- Backbone为Qwen3-8B-Instruct模型，在Open-Travel任务上进行训练，其Step 0时的性能为0
- RL训练过程中，性能呈现稳定上升趋势，并在Step 160时达到71%的峰值胜率

结果介绍



高德地图真实业务场景落地

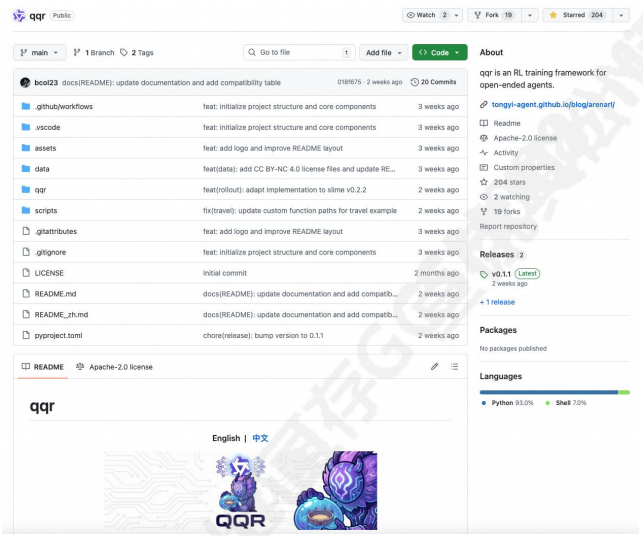
- **确定性 POI 搜索：** 在规则明确且竞争激烈的 POI 搜索任务中，ArenaRL 展现了对刚性约束的极强适应力。相比基线，核心业务准确率从**75% 提升至 83%**。

- **复杂开放式规划：** 面对涉及多步推理的开放式行程规划——例如寻找外滩带江景露台且适合约会的酒吧，或北京往返天津、需兼顾行李与时间的跨城出行等复杂需求，业务核心指标从 **69% 提升至 80%**。模型展现出了更强的逻辑自洽性，能够有效处理模糊的用户意图

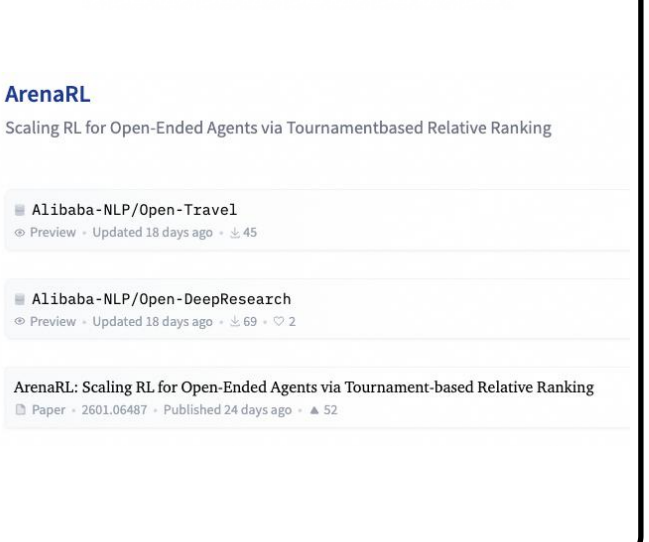
结果介绍

如何使用

GitHub



Hugging Face



ModelScope

后续规划

- 算法优化
- 扩展至多模态Agent
- 提高无冷启动直接RL训练的性能

感谢聆听

GitHub

Huggingface

ModelScope

Arxiv