

PR #2124 完整报告

THUDM/slime

fix(agent) SWE coding-agent RL stability bugs (abort handling, session cleanup)

合并时间: 2026-06-23 14:57

原文链接: <http://prhub.com.cn/THUDM/slime/pull/2124>

执行摘要

- 一句话: 修复 SWE coding-agent RL 稳定性问题
- 推荐动作: 建议精读。PR 展示了 agent RL 训练中异常处理的最佳实践: 通过 `drop_session` 分离清理与样本生成, 通过 `rollout_log_probs=[0.0]` 和 `remove_sample=True` 优雅处理中止样本。测试覆盖不足, 后续可补充。

功能与动机

修复 SWE coding-agent RL 训练中的稳定性 bug, 确保中止样本不产生梯度, 会话正确清理, 以及避免 `run_command` 在 `done` 文件为空时崩溃。

实现拆解

1. 在 `slime/agent/adapters/common.py` 中新增 `drop_session` 方法: 仅执行清理 (关闭会话、弹出 `store`), 并调用 `manager.drop_session` 丢弃轨迹树和轮次计数, 不生成 `Sample` 对象。同时将 `finish_session` 的 `base_sample` 改为必需参数。
2. 在 `slime/agent/trajectory.py` 中新增 `drop_session` 方法: 幂等地弹出 `_trees` 和 `_turn_count`, 与 `get_trajectory` 中的清理逻辑一致。
3. 在 `examples/coding_agent_rl/generate.py` 中: 将 `finally` 块中的 `finish_session` 替换为 `drop_session`, 确保异常 / 超时后仅做清理; 并在 `_abort_result` 中设置 `rollout_log_probs=[0.0]` 和 `remove_sample=True`, 使中止样本不参与 GRPO/GSPO 基线计算。
4. 在 `slime/agent/harness/common.py` 的 `run_command` 中: 读取 `done` 文件内容后, 先检查是否为空字符串, 避免 `int('')` 崩溃。
5. 在 `examples/coding_agent_rl/run_qwen36_35b_a3b_swe_8nodes.sh` 中: 将优势估计器从 `gspo` 改为 `grpo`, 并调整 `eps-clip` 为 `0.2/0.28`, 适配新版算法配置。

关键文件:

- `slime/agent/adapters/common.py` (模块 适配器; 类别 `source`; 类型 `core-logic`; 符号 `drop_session, finish_session`): 新增 `drop_session` 方法, 并调整 `finish_session` 参数, 是核心清理逻辑的入口。
- `slime/agent/trajectory.py` (模块 轨迹管理器; 类别 `source`; 类型 `core-logic`; 符号 `drop_session`): 新增 `drop_session` 方法, 幂等清理轨迹数据和轮次计数。

- examples/coding_agent_rl/generate.py (模块 训练脚本; 类别 source; 类型 core-logic ; 符号 generate, _abort_result) : 核心训练脚本: 将 finally 块改为 drop_session, 并在 _abort_result 中设置 rollout_log_probs=[0.0] 与 remove_sample=True, 实现中止样本的正确处理。
- slime/agent/harness/common.py (模块 沙箱适配器; 类别 source; 类型 core-logic) : 修复 run_command 在 done 文件内容为空时 int('') 崩溃问题。
- examples/coding_agent_rl/run_qwen36_35b_a3b_swe_8nodes.sh (模块 部署脚本; 类别 other; 类型 core-logic) : Shell 脚本调整算法参数: 优势估计器从 GSPO 改为 GRPO, eps-clip 从 1e-4 改为 0.2/0.28。

关键符号: drop_session, finish_session, run_command, generate, _abort_result

关键源码片段

slime/agent/adapters/common.py

新增 drop_session 方法, 并调整 finish_session 参数, 是核心清理逻辑的入口。

```
# slime/agent/adapters/common.py
# 新增的方法: 只做清理, 不生成 Sample, 用于异常 / 超时后的 finally 块
async def drop_session(self, sid: str, *, wait_timeout: float = 5.0) -> None:
    await self.shutdown_session(sid, wait_timeout=wait_timeout) # 关闭并等待 inflight 完成
    self.store.pop(sid, None) # 移除 session 对象
    self.manager.drop_session(sid) # 丢弃轨迹树和轮次计数

# finish_session 的 base_sample 参数从可选变为必需, 强制调用者明确传入
async def finish_session(
    self,
    sid: str,
    *,
    base_sample, # 去掉了 =None, 变为必需参数
    reward: float = 0.0,
    extra_metadata: dict | None = None,
    wait_timeout: float = 5.0,
) -> list:
    # ... 原有逻辑不变 ...
```

examples/coding_agent_rl/generate.py

核心训练脚本: 将 finally 块改为 drop_session, 并在 _abort_result 中设置 rollout_log_probs=[0.0] 与 remove_sample=True, 实现中止样本的正确处理。

```
# examples/coding_agent_rl/generate.py
# 改动 1: finally 块使用 drop_session 确保只清理, 不产生 Sample
finally:
    await state.adapter.drop_session(session_id) # cleanup only, idempotent

# 改动 2: 中止样本标记为无梯度, 并排除出组基线
def _abort_result(sample: Sample, reason: str, instance_id: str) -> list[Sample]:
    sample.tokens = [0, 0]
```

```

sample.response = ""
sample.response_length = 1
sample.loss_mask = [0]
sample.rollout_log_probs = [0.0] # 新增: 设为 0 使 loss 贡献为 0
sample.reward = 0.0
sample.remove_sample = True # 新增: 从组基线中排除
sample.status = Sample.Status.ABORTED
sample.metadata = {**(sample.metadata or {}), "abort_reason": reason}
return [sample]

```

slime/agent/harness/common.py

修复 `run_command` 在 `done` 文件内容为空时 `int('')` 崩溃问题。

```

# slime/agent/harness/common.py # run_command 函数片段
if ec == 0:
    exit_code_text = (out or "").strip() # 先获取文本
    if exit_code_text: # 检查是否为空
        exit_code = int(exit_code_text) # 非空才转换
        break
    # 如果为空, 继续等待或超时返回 EXIT_TIME_BUDGET_EXCEEDED
return exit_code

```

评论区精华

无 review 讨论。

- 暂无高价值评论线程

风险与影响

- 风险: 风险较低。变更集中在 `agent` 异常路径和会话清理, 正常路径不受影响。
`drop_session` 幂等且被 `finally` 块保障, 不会漏清理。`remove_sample=True` 在 GRPO 训练中可能影响基线计算, 但符合预期。Shell 脚本参数变化需验证训练效果。
- 影响: 影响范围: 仅 SWE coding-agent RL 训练流程。用户: 训练稳定性提升, 避免中止样本导致梯度异常或 `crash`。系统: 会话资源及时释放, 减少内存泄漏风险。团队: 该 PR 修复了线上遇到的问题, 值得尽快合并。
- 风险标记: 核心路径变更, 缺少测试覆盖, 参数调整

关联脉络

- PR #2125 feat(coding_agent_rl): select claude_code/codex harness+adapter pair via SWE_AGENT: 同一功能线 (coding-agent RL), 此 PR 为其稳定性修复
- PR #2005 [coding-agent-rl] Refactor coding-agent RL: turn-node TrajectoryManager + pluggable harness layer: 该 PR 引入的架构为此 PR 的修复提供了基础, 且修改了相同的文件区域 (trajectory.py, adapters/common.py)