

PR #1957 完整报告

THUDM/slime

Minor refactor for coding agent rl logic and remove SWE_LIST_TRAJECTORY

合并时间: 2026-05-27 12:04

原文链接: <http://prhub.com.cn/THUDM/slime/pull/1957>

执行摘要

- 一句话: 重构 agent RL 生成逻辑并移除 SWE_LIST_TRAJECTORY 配置
- 推荐动作: 值得精读 generate.py 的模块化拆分方式, 以及如何将单体生成函数重构为管道式编排。同时注意配置移除可能对现有实验脚本的影响, 建议运行 fan-out 测试验证。

功能与动机

移除已弃用的 SWE_LIST_TRAJECTORY 配置 (该配置允许将多个 segment 折叠为单个 Sample), 现在始终生成每个 segment 对应的 Sample, 避免配置混乱并减少维护成本。同时将 generate() 函数模块化以提高可读性和可测试性。

实现拆解

1. 重构 generate.py: 将原 generate() 中的步骤提取为 _run_claude_code、_get_diff、_merge_samples 等辅助函数, 引入 dataclass RewardResult 作为中间数据载体。删除所有与 SWE_LIST_TRAJECTORY 相关的条件分支, generate() 现在始终返回 list[Sample]。
2. 修改 shell 启动脚本: 移除 SWE_LIST_TRAJECTORY 的 export 和注释, 调整 fan-out 语义说明。
3. 更新 README: 从环境变量表中移除 SWE_LIST_TRAJECTORY 行, 简化 Fan-out 章节描述。

关键文件:

- examples/coding_agent_rl/generate.py (模块 代理 RL; 类别 source; 类型 dependency-wiring; 符号 RewardResult, _start_session, _run_claude_code, _get_diff) : 核心重构文件, 涉及 generate() 函数拆分、SWE_LIST_TRAJECTORY 移除、新增辅助函数和 RewardResult 数据结构。
- examples/coding_agent_rl/run_qwen36_35b_a3b_swe_8nodes.sh (模块 部署脚本; 类别 other; 类型 core-logic) : 同步移除 SWE_LIST_TRAJECTORY 环境变量导出和相关注释, 调整 fan-out 语义描述。
- examples/coding_agent_rl/README.md (模块 文档; 类别 docs; 类型 documentation) : 从环境变量表中移除 SWE_LIST_TRAJECTORY 行, 简化 Fan-out 章节描述。

关键符号: generate, _run_claude_code, _get_diff, _merge_samples, _pop_segments, _start_session

关键源码片段

examples/coding_agent_rl/generate.py

核心重构文件，涉及 generate() 函数拆分、SWE_LIST_TRAJECTORY 移除、新增辅助函数和 RewardResult 数据结构。

文件：examples/coding_agent_rl/generate.py

重构后的 generate() 使用四个辅助函数组织流程，并通过 RewardResult 传递中间状态

```
@dataclass
```

```
class RewardResult:
```

```
    """保存 reward 评估结果，供后续合并使用"""
```

```
    reward: float
```

```
    metadata: dict
```

```
async def _run_claude_code(sample, state) -> middleware.Session:
```

```
    """阶段1: 启动 E2B 沙箱，安装 Node + Claude Code，运行 agent"""
```

```
    # 根据 sample 的 metadata 创建 sandbox，返回 session 对象
```

```
    ...
```

```
async def _get_diff(session) -> str:
```

```
    """阶段2: 在 agent 运行结束后捕获 git diff 补丁"""
```

```
    ...
```

```
async def _merge_samples(session, reward_result, state) -> list[Sample]:
```

```
    """阶段4: 从 middleware 拉取 token segment，结合 reward 组装为 Sample 列表"""
```

```
    segments = await middleware.pop_session_split(session.uid)
```

```
    # 每个 segment 对应一个 Sample，reward 均分
```

```
    samples = []
```

```
    for seg in segments:
```

```
        samples.append(Sample(
```

```
            prompt_ids=seg.prompt_ids,
```

```
            response_ids=seg.response_ids,
```

```
            loss_mask=seg.loss_mask,
```

```
            reward=reward_result.reward / len(segments),
```

```
            rollout_id=sample.rollout_id,
```

```
        ))
```

```
    return samples
```

```
async def generate(samples, args) -> AsyncIterator[Sample]:
```

```
    """四阶段编排器: agent -> diff -> evaluate -> merge"""
```

```
    state = _State(args)
```

```
    async for sample in samples:
```

```
        try:
```

```
            session = await _run_claude_code(sample, state)
```

```
            diff = await _get_diff(session)
```

```
            reward_result = await sandbox.evaluate(diff, sample.metadata)
```

```
            async for merged in _merge_samples(session, reward_result, state):
```

```
                yield merged
```

```
except Exception:
    logger.exception("sample failed, skipping")
    continue
```

评论区精华

无公开 review 讨论或评论。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险在于 SWE_LIST_TRAJECTORY=0 的 collapsed 模式被移除，若外部工作流程依赖该模式则行为改变。此外，新引入的 _merge_samples 等函数若未正确处理 reward 分割可能导致 reward 计算错误。建议确认所有依赖方已迁移到 fan-out 模式。
- 影响：影响范围限于 coding_agent_rl 示例的用户。用户需要删除 SWE_LIST_TRAJECTORY 设置（原 shell 中默认已设为 1，所以影响较小）。代码重构提升了可读性和模块性，便于后续扩展。
- 风险标记：移除旧配置可能破坏依赖 collapsed 模式的工作流

关联脉络

- PR #1923 [examples] add coding_agent_rl: agent-in-sandbox RL minimal demo: 该 PR 首次引入 coding_agent_rl 示例，当前 PR 在其基础上进行重构。
- PR #1958 Move coding_agent_rl's helper function to sandbox.py: 后续 PR 将部分辅助函数移至 sandbox.py，与当前重构存在文件交集和功能衔接。