

PR #1963 完整报告

THUDM/slime

Fix trajectory merging logic

合并时间: 2026-05-27 18:19

原文链接: <http://prhub.com.cn/THUDM/slime/pull/1963>

执行摘要

- 一句话: 修复轨迹合并逻辑, 简化 middleware 损失掩码计算
- 推荐动作: 建议精读 `slime/agent/trajectory.py` 中 `merge_turns` 的 `common_prefix` 对齐设计, 该模式可推广至其他序列拼接场景。注意检查下游代码对 `output_loss_mask` 的依赖, 并及时适配 `output_log_probs`。新增测试用例可作为后续修改的回归保障。

功能与动机

原始 middleware 中的 `raw-splice` 和 `TITO` 验证逻辑过于复杂且难以维护, `prompt drift` 处理不精确导致训练 `target` 污染。本次重构将序列对齐与 `mask` 计算职责转移到 `trajectory` 模块, 通过最长公共前缀匹配精确处理 `drift`, 并移除冗余的 `output_loss_mask` 字段, 使数据流更清晰。

实现拆解

1. 重构 `slime/agent/trajectory.py`: 移除 `TurnRecord.output_loss_mask` 字段, 新增 `output_log_probs` 和 `TokenSegment.rollout_log_probs`; 新增 `TurnSegment` 数据类及 `make_turn_segment` 工厂函数; 重写 `merge_turns`, 将之前基于严格前后缀匹配的逻辑改为基于 `_common_prefix_len` 的最长公共前缀匹配, 在 `prompt drift` 时仅保留匹配前缀并对应的 `loss_mask` 置 0, 同时记录 `output_spans` 以处理跨 `turn` 的掩码。
2. 简化 `examples/coding_agent_rl/middleware.py`: 移除 `_RAW_PH_PREFIX/SUFFIX`、`_common_prefix_len`、`verify_tito_for_turn`、`_record_turn` 等函数, 不再在 middleware 中维护 `raw splice` 和 `per-turn loss mask`; 导入 `TurnSegment`、`make_turn_segment`、`merge_turn_segments`, 将损失掩码决策全部交给 `trajectory.merge_turns`; 新增 `_build_tools_schema`、`_anthropic_tools_to_chat_tools` 等工具函数以支持动态工具注入。
3. 新增测试文件 `tests/test_agent_trajectory.py`: 用 6 个 `@pytest.mark.unit` 用例覆盖匹配前缀保留、跳过中间 `turn`、连续 `drift`、输出被拆分、`token` 数变化、`prompt base` 改变等场景, 验证 `merge_turns` 的 `loss_mask` 和 `rollout_log_probs` 输出。
4. 更新 CI 注册: 将 `test_agent_trajectory.py` 加入 `.github/workflows/pr-test.yml` 的 CPU 测试矩阵, 并同步更新 `Jinja2` 模板。
5. 辅助变更: `scripts/run-qwen3.5-27B.sh` 新增 `rollout` 配置参数 (TP/DP/ 内存利用率); `examples/coding_agent_rl/run_qwen36_35b_a3b_swe_8nodes.sh` 新增 `sclang mamba` 调度策略; `Skil` 文档补充 CI 兼容性要点。

关键文件：

- `slime/agent/trajectory.py`（模块 轨迹；类别 `source`；类型 `core-logic`；符号 `_prompt_matches_current`, `TurnSegment`, `_output_mask`, `make_turn_segment`）：核心文件：重写了轨迹合并逻辑，引入 `common_prefix` 精确对齐，移除 `loss_mask` 字段，新增 `logprobs` 支持。
- `examples/coding_agent_rl/middleware.py`（模块 中间件；类别 `source`；类型 `core-logic`；符号 `_make_segment`, `_build_tools_schema`, `_anthropic_tools_to_chat_tools`, `_common_prefix_len`）：大幅精简：移除 `raw-splice` 和 `per-turn` 验证逻辑，将损失掩码责任转移至 `trajectory`，新增工具注入函数。
- `tests/test_agent_trajectory.py`（模块 测试；类别 `test`；类型 `test-coverage`；符号 `test_merge_turns_preserves_matched_prefix_on_prompt_drift`, `test_merge_turns_drops_middle_turn_when_next_prompt_skips_it`, `test_merge_turns_handles_consecutive_prompt_drifts`, `test_merge_turns_masks_whole_output_when_prompt_drift_splits_it`）：新增单元测试，全面覆盖 `merge_turns` 的各种 `drift` 场景，确保重构后逻辑正确。
- `.github/workflows/pr-test.yml`（模块 `CI`；类别 `infra`；类型 `infrastructure`）：将新测试文件注册到 `CI CPU` 测试矩阵，确保 `CI` 执行。
- `.github/workflows/pr-test.yml.j2`（模块 `CI`；类别 `infra`；类型 `infrastructure`）：同步更新 `Jinja2` 模板以生成 `CI` 配置文件。

关键符号：`merge_turns`, `_common_prefix_len`, `_output_log_probs`, `make_turn_segment`, `merge_turn_segments`, `_build_tools_schema`, `_anthropic_tools_to_chat_tools`

关键源码片段

`slime/agent/trajectory.py`

核心文件：重写了轨迹合并逻辑，引入 `common_prefix` 精确对齐，移除 `loss_mask` 字段，新增 `logprobs` 支持。

`slime/agent/trajectory.py` - 核心合并函数

```
def merge_turns(turns: list[TurnRecord], *, metadata: dict[str, Any] | None = None) -> TokenSegment | None:
    if not turns:
        return None

    prompt_ids = list(turns[0].prompt_ids)
    response_ids: list[int] = []
    loss_mask: list[int] = []
    rollout_log_probs: list[float] = []
    output_spans: list[tuple[int, int]] = [] # 记录每个 turn 输出在 response_ids 中的 [start, end]

    for i, turn in enumerate(turns):
        if i > 0:
            # 如果新的 prompt 不再包含之前的 base_prompt，说明上下文完全变了，直接重启
```

```

segment
if turn.prompt_ids[: len(prompt_ids)] != prompt_ids:
    logger.warning("[trajectory] merge prompt base changed; starting segment from
drifted prompt")
    prompt_ids = list(turn.prompt_ids)
    response_ids = []
    loss_mask = []
    rollout_log_probs = []
    output_spans = []
else:
    # 取得新 prompt 中超出之前 response 的部分
    prompt_suffix = turn.prompt_ids[len(prompt_ids) :]
    matched_len = _common_prefix_len(response_ids, prompt_suffix)
    if matched_len < len(response_ids):
        # 只匹配了部分历史输出, 未匹配部分必须从响应中截掉, 并将对应的 loss 置 0
        logger.warning(
            "[trajectory] merge prefix drift; truncating %d unstitched response tokens",
            len(response_ids) - matched_len,
        )
        for start, end in output_spans:
            if start < matched_len < end:
                # 如果这个 turn 的输出被部分截断, 截断部分的 loss_mask 和 logprob
                # 必须清零
                loss_mask[start:matched_len] = [0] * (matched_len - start)
                rollout_log_probs[start:matched_len] = [0.0] * (matched_len - start)
            response_ids = response_ids[:matched_len]
            loss_mask = loss_mask[:matched_len]
            rollout_log_probs = rollout_log_probs[:matched_len]
            output_spans = [
                (start, min(end, matched_len)) for start, end in output_spans if start <
                matched_len
            ]
        # 新的 context tail (未匹配部分) 加入响应, loss 为 0
        context_tail = prompt_suffix[matched_len:]
        response_ids.extend(context_tail)
        loss_mask.extend([0] * len(context_tail))
        rollout_log_probs.extend([0.0] * len(context_tail))
    # 追加当前 turn 的生成输出
    start_pos = len(response_ids)
    response_ids.extend(turn.output_ids)
    loss_mask.extend([1] * len(turn.output_ids)) # 默认全为 1, 后续可能会被部分清零
    rollout_log_probs.extend(_output_log_probs(turn))
    output_spans.append((start_pos, len(response_ids)))

return TokenSegment(
    prompt_ids=prompt_ids,
    response_ids=response_ids,
    loss_mask=loss_mask,
    rollout_log_probs=rollout_log_probs,

```

```
metadata=dict(metadata or {}),  
)
```

评论区精华

本 PR 无公开 review 评论。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险：① 移除 TurnRecord.output_loss_mask 字段会破坏现有直接引用该字段的代码（如自定义 rollout 后处理），需确认无外部依赖；② output_log_probs 要求上游引擎提供 logprobs，若未提供将默认为空列表，可能导致下游计算出错；③ 新的 merge_turns 逻辑对极长序列对齐性能需关注；④ 测试覆盖了常见 drift 场景，但未测试 finish_reason 异常、空输出等边界。
- 影响：影响 coding_agent_rl 示例的全部 rollout 流程，以及任何直接使用 slime.agent.trajectory 模块的组件。轨迹合并逻辑是 agent RL 训练数据的核心管道，正确性直接影响训练效果。简化 middleware 可降低后续维护成本，但向下游暴露了新字段 output_log_probs 和 rollout_log_probs。
- 风险标记：核心路径变更，数据格式迁移（移除 output_loss_mask），测试覆盖有限

关联脉络

- PR #1956 Add slime/agent/ and move sandbox impl inside: 建立了 slime/agent 模块，为 trajectory.py 提供了组织归属。
- PR #1960 Extract more util code from coding_agent_rl example: 将 trajectory 和 parsing 等工具从示例抽取到核心库，与本 PR 形成一致方向。
- PR #1923 [examples] add coding_agent_rl: agent-in-sandbox RL minimal demo: 首次引入 coding_agent_rl 示例，其中包含了此处重构前的 raw-splice 逻辑。