

PR #6089 完整报告

verl-project/verl

[rollout] fix: truncate routed_experts to response_length to match truncated response_ids

合并时间: 2026-04-24 15:20

原文链接: <http://prhub.com.cn/verl-project/verl/pull/6089>

执行摘要

- 一句话: 修复工具调用循环中 MoE 路由专家张量未截断导致形状不匹配的 bug
- 推荐动作: 该 PR 值得精读, 它演示了在处理 MoE 路由专家信息时一个常见的陷阱: routed_experts 与 response_ids 的语义差异 (前者包含 prompt 部分, 后者不包含), 以及如何通过正确的截断来避免形状不匹配。同时这是一个很好的 review 案例, 展示了 AI 辅助 review 如何抓住核心逻辑错误并给出精准建议。对于维护 agent_loop 的开发者, 建议关注其他可能使用 routed_experts 的地方是否有类似问题。

功能与动机

该 PR 修复 issue #6027: 在 ToolAgentLoop 中, 当 response_ids 超过 response_length 被截断后, routed_experts 未被截断, 导致在 agent_loop.py 中将路由专家张量赋值到切片时出现形状不匹配的错误 (RuntimeError)。用户描述了在 MoE 模型配合 router_replay=R3、小 data.response_length 和 ToolAgentLoop 时必现的异常。

实现拆解

该 PR 只涉及一个文件 verl/experimental/agent_loop/tool_agent_loop.py, 核心变更位于 AgentStateMachine.run 方法的末尾, 在构造 AgentLoopOutput 时修改了 routed_experts 字段的传入值。步骤如下:

1. 定位问题: 在 run() 方法中, 构造 AgentLoopOutput 时, response_ids、response_mask、response_logprobs 都已按 self.response_length 截断, 但 routed_experts 直接传入了 agent_data.routed_experts (完整长度), 导致下游 agent_loop.py 中将此张量赋值到已截断的切片时报错。
2. 修正截断逻辑: 将 routed_experts 替换为 agent_data.routed_experts[:len(prompt_ids) + self.response_length]。注意这里截断长度是 prompt_ids 长度加 response_length, 而非仅 self.response_length, 因为 routed_experts 同时包含 prompt 和 response 两部分的路由专家信息 (与 response_ids 等仅含 response 部分不同)。同时添加了 None 守卫, 以兼容未收集路由专家信息的场景。
3. 保留原有构造顺序: 除 routed_experts 外, 其他字段传递逻辑不变, 最小化改动风险。

关键文件:

- verl/experimental/agent_loop/tool_agent_loop.py (模块 Agent 循环; 类别 source; 类型 core-logic; 符号 AgentStateMachine.run, AgentLoopOutput) : 包含核心

AgentStateMachine.run 方法，是 bug 定位和修复的唯一文件。

关键符号：AgentStateMachine.run

关键源码片段

[verl/experimental/agent_loop/tool_agent_loop.py](#)

包含核心 AgentStateMachine.run 方法，是 bug 定位和修复的唯一文件。

```
# run() 方法末尾构造 AgentLoopOutput
# 注意: routed_experts 含 prompt + response 两部分, 需使用 len(prompt_ids) + self.response_
length 截断
output: AgentLoopOutput = AgentLoopOutput(
    prompt_ids=prompt_ids,
    response_ids=response_ids[: self.response_length],
    response_mask=agent_data.response_mask[: self.response_length],
    multi_modal_data=multi_modal_data,
    response_logprobs=agent_data.response_logprobs[: self.response_length]
    if agent_data.response_logprobs
    else None,
    num_turns=agent_data.user_turns + agent_data.assistant_turns + 1,
    metrics=agent_data.metrics,
    # 修正: routed_experts 需截断到 prompt+response 总长度
    routed_experts=(
        agent_data.routed_experts[: len(prompt_ids) + self.response_length]
        if agent_data.routed_experts is not None
        else None
    ),
    extra_fields=agent_data.extra_fields,
)
```

评论区精华

1. 人工助手代码审查的重点: gemini-code-assist[bot] 在审查时指出, 初始提交中直接将 routed_experts 截断为 self.response_length 是错误的, 因为 routed_experts 包含 prompt 和 response 两部分 (与 response_ids 仅含 response 不同), 正确做法应是截断到 len(prompt_ids) + self.response_length, 并给出了确切的代码建议。
 2. 讨论采纳与合并: 此建议被作者采纳 (通过后续提交 0d7f3111 为 wuxibin89 的修正) 。评审者 wuxibin89 最终批准了该 PR。
- routed_experts 截断长度应为 prompt + response 而非仅 response (correctness): 采纳建议, 修正截断逻辑为 agent_data.routed_experts[: len(prompt_ids) + self.response_length]。

风险与影响

- 风险:

1. 回归风险（低）：变更仅影响 `routed_experts` 的截断逻辑，且严格对齐了已有的 `response_ids` 截断模式，回归概率低。若 `prompt_ids` 在截断后长度发生变化（如后续修改 `prompt` 截断逻辑），需同步检查此处的长度一致性。
 2. 性能影响（无）：仅增加一次切片操作，开销可忽略。
 3. 兼容性风险（低）：未收集 `routed_experts` 的场景（`agent_data.routed_experts is None`）已通过 `None` 守卫正确处理，无兼容问题。
 4. 数据完整性风险（低）：截断可能丢弃部分路由专家信息，但这是由 `response_length` 限制的预期行为——与 `response_ids` 的截断同步，确保下游张量形状一致。
- 影响：
 - 影响范围：仅限于 MoE 模型使用 `ToolAgentLoop` 且触发响应截断的场景（通常是 `response_length` 较小的配置）。
 - 用户影响：修复了用户报出的 `RuntimeError`，使相关训练 / 推理流程不再因形状不匹配而崩溃。
 - 系统影响：无系统性影响。
 - 团队影响：低，这是一个用户报告的 bug 修复，非架构性变更。
 - 风险标记：核心路径变更

关联脉络

- PR #6055 [rollout] feat: improve error messages for malformed tool calls: 同属 `agent_loop` 模块，修改了相同的 `tool_agent_loop.py` 文件（函数 `AgentStateMachine.run`）