

PR #1862 完整报告

THUDM/slime

chore: include length context in slice_log_prob_with_cp assert

合并时间: 2026-04-28 10:53

原文链接: <http://prhub.com.cn/THUDM/slime/pull/1862>

执行摘要

- 一句话: 增强断言信息以提升调试体验
- 推荐动作: 值得合并的小优化, 无需额外 review。

功能与动机

PR body 明确指出: 当 `len(log_prob) != response_length` 时, 裸 `assert` 仅抛出 `AssertionError`, 无法区分是 `rollout_log_probs` 被截断、`response_length` 过时还是 `prompt_length` 计算错误, 开发者必须添加打印并重跑才能定位。加入三个数值后, 根因可从栈追踪中直接推断。

实现拆解

1. 在 `slime/backends/megatron_utils/cp_utils.py` 文件的 `slice_log_prob_with_cp` 函数中, 将第 226 行的裸 `assert len(log_prob) == response_length` 替换为带格式化消息的 `assert`, 消息包含 `len(log_prob)`、`response_length` 和 `total_length` 三个变量的值。
2. 改动仅 4 行新增、1 行删除, 不涉及其他文件或测试。

关键文件:

- `slime/backends/megatron_utils/cp_utils.py` (模块 后端工具; 类别 `source`; 类型 `core-logic`): 核心变更文件, 修改了 `slice_log_prob_with_cp` 函数中的断言消息。

关键符号: `slice_log_prob_with_cp`

关键源码片段

`slime/backends/megatron_utils/cp_utils.py`

核心变更文件, 修改了 `slice_log_prob_with_cp` 函数中的断言消息。

```
# slime/backends/megatron_utils/cp_utils.py
```

```
def slice_log_prob_with_cp(
    log_prob: list[float] | torch.Tensor,
    total_length: int,
    response_length: int,
    qkv_format: str = "thd",
    max_token_len: int | None = None,
```

```

) -> list[float] | torch.Tensor:
    # 带上下文信息的断言，方便从栈追踪直接定位问题
    assert len(log_prob) == response_length, (
        f"log_prob length mismatch: len(log_prob)={len(log_prob)}, "
        f"response_length={response_length}, total_length={total_length}"
    )

    cp_size = mpu.get_context_parallel_world_size()

    if cp_size == 1:
        return log_prob

    prompt_length = total_length - response_length
    _, _, logits_offset, _ = get_logits_and_tokens_offset_with_cp(
        total_length, response_length, qkv_format, max_token_len
    )

    chunk_1 = log_prob[logits_offset[0][0] - (prompt_length - 1) : logits_offset[0][1] - (prompt_length - 1)]
    chunk_2 = log_prob[logits_offset[1][0] - (prompt_length - 1) : logits_offset[1][1] - (prompt_length - 1)]

    if isinstance(log_prob, list):
        return chunk_1 + chunk_2
    else:
        return torch.cat([chunk_1, chunk_2], dim=0)

```

评论区精华

无 review 评论或讨论。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。仅修改了 assert 消息格式，不改变任何控制流、数据或返回值。断言条件本身未变更，因此不会引入回归。无性能影响。
- 影响：影响范围极小，仅对 Context Parallel 训练中触发该断言失败的场景有正向调试体验提升。不影响正常路径。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR