

PR #29989 完整报告

sgl-project/sglang

[diffusion] fix: slice img_shapes per-sample in rollout response extractor

合并时间: 2026-07-07 08:56

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/29989>

执行摘要

- 一句话: 修复 rollout 响应中 img_shapes 逐样本切片错误
- 推荐动作: 值得合并, 修复了明确的功能回归 bug。改动简单、安全, 已获得批准。建议合并后补充一个针对 _extract_single_sample_tensor 的单元测试, 覆盖 img_shapes 的 batch_size > 1 场景, 以预防未来回归。

功能与动机

经过 Qwen-Image 条件批对齐修复 (commit [a9d657bf3](#)) 后, pos_cond_kwargs 中的 img_shapes 列表长度变为 batch_size (每个多输出样本一个条目), 而非之前的长度 1。_extract_single_sample_tensor 递归遍历列表时未按 sample_idx 切片, 导致每个单样本响应携带了完整的 N 长度 img_shapes, 破坏了下游消费者对 img_shapes 长度为 1 的约定。

实现拆解

1. 修改函数签名: 在 python/sglang/multimodal_gen/runtime/entrypoints/post_training/rollout_api.py 中, 为 _extract_single_sample_tensor 添加可选关键字参数 current_key: str | None = None, 用于在递归过程中传递当前字典键名。
2. 字典分支传播 key: 当递归进入 dict 分支时, 将当前键 k 作为 current_key 传递到子调用, 使列表分支能感知父级键名。
3. 列表分支特殊处理 img_shapes: 在 list 分支中, 检查 current_key == "img_shapes" 且 len(obj) == batch_size 时, 直接返回 [obj[sample_idx]] 而非递归深入列表元素, 从而正确提取单个样本的 shape 并保持长度 1 的约定。
4. 维护其他递归路径: tuple 和普通 list 分支同样传递 current_key, 确保递归过程中键名传播的一致性。

关键文件:

- python/sglang/multimodal_gen/runtime/entrypoints/post_training/rollout_api.py (模块响应提取; 类别 source; 类型 entrypoint; 符号 _extract_single_sample_tensor): 该文件是实现响应提取的核心入口, 修改了 _extract_single_sample_tensor 函数以支持 per-sample 的 img_shapes 切片。所有变更集中于此文件。

关键符号: _extract_single_sample_tensor

关键源码片段

python/sglang/multimodal_gen/runtime/entrypoints/post_training/rollout_api.py

该文件是实现响应提取的核心入口，修改了 `_extract_single_sample_tensor` 函数以支持 per-sample 的 `img_shapes` 切片。所有变更集中于此文件。

```
def _extract_single_sample_tensor(
    obj: Any,
    sample_idx: int,
    batch_size: int,
    *,
    current_key: str | None = None, # 新增：当前处理的字典键名，用于判断是否特殊处理
) -> Any:
    if isinstance(obj, torch.Tensor):
        if obj.dim() >= 1 and obj.shape[0] == batch_size:
            return obj[sample_idx].contiguous()
        return obj
    if isinstance(obj, dict):
        # 递归时传递当前键名，使子节点感知上下文
        return {
            k: _extract_single_sample_tensor(
                v, sample_idx, batch_size, current_key=k
            )
            for k, v in obj.items()
        }
    if isinstance(obj, list):
        # 关键修复：如果父键名是 img_shapes 且列表长度等于 batch_size，
        # 说明该列表每个元素对应一个样本，直接按索引取出并包装为单元素列表
        if current_key == "img_shapes" and len(obj) == batch_size:
            return [obj[sample_idx]]
        # 否则递归处理每个元素
        return [_extract_single_sample_tensor(
            v, sample_idx, batch_size, current_key=current_key
        ) for v in obj]
    if isinstance(obj, tuple):
        return tuple(_extract_single_sample_tensor(
            v, sample_idx, batch_size, current_key=current_key
        ) for v in obj)
    return obj
```

评论区精华

无 review 讨论。PR 仅有一个批准（来自 mickqian），无评论或争议。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。变更仅影响响应提取路径中的 `_extract_single_sample_tensor` 函数，且为纯 Python 控制流调整：新增一个关键字参数 `current_key` 和一个 if 分支。不修改模型

前向逻辑，不影响性能热点。未添加单元测试，但改动逻辑简单，回归风险小。需注意如果后续有新的 dict 键包含 batch_size 长度列表且需要按元素深入，此特殊处理可能不适用，但可通过扩展 current_key 判断逻辑适配。

- 影响：影响范围局限于 rollout_api.py 中的响应提取流程，仅当使用多输出样本（batch_size > 1）时触发。修复后，下游消费者（如 RL 训练脚本）收到的 img_shapes 字段正确为长度 1，避免了解析错误或行为异常。对单样本场景无影响。
- 风险标记：缺少测试覆盖

关联脉络

- 暂无明显关联 PR