PR #26232 完整报告

sgl-project/sglang

[SRT] minor: reuse req input id array for unpadded ids

合并时间：2026-05-26 08:58

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26232

执行摘要

- 一句话：复用 origin_input_ids 避免重复内存分配
- 推荐动作：该 PR 属于代码整洁性改进，逻辑简单，风险低，适合快速合入。对于关注性能细节的读者，可借此了解常见请求路径下的小型内存优化手法。

功能与动机

在常见的请求构造路径中，origin_input_ids_unpadded 和 origin_input_ids 值相同，但原代码会重复调用 array("q", ...) 创建两个相同的数组，造成不必要的内存分配。PR 描述明确说明 'Avoid a duplicate array("q") conversion/allocation for the common request construction path.'

实现拆解

1. 调整 Req.__init__ 中赋值顺序：先初始化 self.origin_input_ids 为 array("q", origin_input_ids)。
2. 条件式赋值 self.origin_input_ids_unpadded：仅当 origin_input_ids_unpadded 参数为非空时才创建新数组；否则直接引用 self.origin_input_ids。
3. 移除原代码中的无条件 array("q", origin_input_ids_unpadded or origin_input_ids) 调用，从而在常见路径下减少一次数组构造。

关键文件：

- python/sglang/srt/managers/schedule_batch.py（模块 调度器；类别 source；类型 core-logic；符号 Req.init）：该文件是唯一变更文件，核心修改了 Req 类的构造函数，优化了 origin_input_ids_unpadded 的初始化逻辑。

关键符号：Req.init

关键源码片段

python/sglang/srt/managers/schedule_batch.py

该文件是唯一变更文件，核心修改了 Req 类的构造函数，优化了 origin_input_ids_unpadded 的初始化逻辑。

```
# python/sglang/srt/managers/schedule_batch.py
# 修改前后对比（仅展示关键部分）
```

```
class Req:
    def __init__(self, ..., origin_input_ids, origin_input_ids_unpadded=None, ...):
        # 先创建 origin_input_ids 数组
        self.origin_input_ids = array("q", origin_input_ids)
        # 仅在提供了独立的 unpadded id 时才创建新数组，否则直接复用 origin_input_ids
        self.origin_input_ids_unpadded = (
            array("q", origin_input_ids_unpadded)
            if origin_input_ids_unpadded
            else self.origin_input_ids
        ) # Before image padding
        # 其余初始化保持不变 ...
```

评论区精华

只有自动机器人 gemini-code-assist[bot] 发表了评论，确认无其他审查反馈。无人工 reviewer 介入讨论。

- 暂无高价值评论线程

风险与影响

- 风险：变更范围极小（仅 8 行），且逻辑保持等价——当 origin_input_ids_unpadded 为 None 时，新代码直接复用 self.origin_input_ids 对象，而非创建新数组。由于 origin_input_ids_unpadded 在使用上仅为读取（注释 # Before image padding 表明其用于图像填充前的 id 序列），不存在后续修改导致意外共享的问题。风险很低。
- 影响：影响范围：仅 Req.__init__ 构造路径，对下游逻辑无直接功能影响。性能上，在无图像填充的常见请求（即 origin_input_ids_unpadded 为 None 时）可减少一次内存分配。由于该构造路径在高并发下频繁触发，积累的分配开销降低可能带来轻微性能提升。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR