

PR #32389 完整报告

sgl-project/sglang

[Bugfix] Fix prefill suspension caused by delayed negotiate_should_allow_prefill invocation

合并时间: 2026-07-26 14:59

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/32389>

执行摘要

- 一句话: 修复 delayed negotiate 导致 prefill 挂起
- 推荐动作: 值得精读。这是一个典型的“缺失上下文导致状态错误”的调度 bugfix, 展示了在复杂调度组合下参数传递的重要性。关注点: negotiate_should_allow_prefill 的参数契约与调用方职责划分。建议后续为该场景补充单元测试, 防止回归。

功能与动机

PR body 指出: 当启用 prefill-delayer、chunked-prefill 和 dp-attention 时, 如果请求没有被拆分, 第一次调用 negotiate_should_allow_prefill 只在执行 add_one_req_ignore_eos 期间发生, 导致 max-running-req 被设置为 0, 后续无法调度新的 prefill batch, prefill 执行一直挂起直到下一个 decode 阶段结束。

实现拆解

1. 定位问题: 在 python/sglang/srt/managers/schedule_policy.py 的 add_one_req_ignore_eos 方法中, 当 prefill-delayer 启用时, 调用 negotiate_should_allow_prefill 仅传入了 local_prefillable=True, 未传递任何调度上下文参数。
2. 分析根因: negotiate_should_allow_prefill 内部需要根据当前系统状态 (如 running_batch 大小、最大并发限制等) 计算 max-running-req。由于缺失这些参数, 协商函数内部可能使用默认值或错误值, 导致 max-running-req 被置为 0, 从而拒绝所有后续 prefill 请求。
3. 实施修复: 在调用处补传 4 个关键参数——running_batch (当前 running batch 的请求数)、max_prefill_bs (最大 prefill batch size)、max_running_requests (最大 running 请求数)、waiting_queue_len (等待队列长度), 使协商逻辑基于真实负载做出决策。
4. 影响验证: 修复后, prefill 不再被挂起, 调度循环能正常推进。改动仅影响调度策略, 不影响模型推理逻辑, 且未新增测试, 依赖现有 CI 覆盖。

关键文件:

- python/sglang/srt/managers/schedule_policy.py (模块 调度器; 类别 source; 类型 core-logic; 符号 add_one_req_ignore_eos): 该文件是调度策略核心, 本次 PR 唯一改动文件。修复了 add_one_req_ignore_eos 调用 negotiate_should_allow_prefill 时未传调度

状态参数导致 prefill 挂起的问题。

关键符号：add_one_req_ignore_eos

关键源码片段

python/sglang/srt/managers/schedule_policy.py

该文件是调度策略核心，本次 PR 唯一改动文件。修复了 add_one_req_ignore_eos 调用 negotiate_should_allow_prefill 时未传调度状态参数导致 prefill 挂起的问题。

```
# 以下代码位于 add_one_req_ignore_eos 方法内，
# 用于在将请求加入 can_run_list 之前，与 prefill-delayer 协商是否允许本次 prefill。
# 修复前只传了 local_prefillable 参数，导致 prefill-delayer 无法感知
# running_batch 等真实状态，将 max-running-req 误设为 0，
# 从而使得后续调度循环因无空位而挂起 prefill。
# 现在补传 4 个调度状态参数，确保协商基于真实系统负载。
if (self.prefill_delayer_single_pass is not None) and (
    not self.prefill_delayer_single_pass.negotiate_should_allow_prefill(
        local_prefillable=True,
        running_batch=self.running_batch.batch_size(), # 当前正在运行的 batch 大小
        max_prefill_bs=self.max_prefill_bs, # 允许的最大 prefill batch size
        max_running_requests=self.max_running_requests, # 允许的最大 running 请求数
        waiting_queue_len=self.waiting_queue_len, # 当前等待队列长度
    )
):
    # 协商失败，拒绝该请求的本次调度
    return AddReqResult.OTHER
```

评论区精华

该 PR 无实质性技术讨论。sglang-npu-bot 直接 APPROVED。在 issue 评论中，sglang-npu-bot 说明一个 GPU 测试因环境占用失败（base-c-test-4-gpu-h100），其余测试全部通过，由于改动仅涉及混合 PD 场景下的调度，不影响其他场景，因此合并。另外 gemini-code-assist 自动评论表示已停止审查。

- 暂无高价值评论线程

风险与影响

- 风险：改动位于调度核心路径（schedule_policy.py 的 add_one_req_ignore_eos），虽然只是补传参数，但 Negotiation 逻辑的行为会随之变化。风险点包括：
 - 当 prefill-delayer 与其他调度特性（如 chunked-prefill、dp-attention）组合时，参数语义必须正确，否则可能引发新的调度偏差；
 - 缺少针对该场景的专项单元测试，回归风险依赖于现有 CI 覆盖；
 - 如果 running_batch 或 waiting_queue_len 等属性在调用时为 None 或未初始化，可能引入异常，但代码中已有默认路径，实际风险较低。

- 影响：影响范围集中在启用了 prefill-delayer + chunked-prefill + dp-attention 的用户，修复后 prefill 不再挂起，吞吐和服务可用性恢复。对其他场景（未启用 prefill-delayer）无影响，因为补传参数仅在 prefill-delayer 分支内生效。改动面小（1 文件、+5/-1），团队维护成本低。
- 风险标记：核心路径变更，缺少测试覆盖

关联脉络

- 暂无明显关联 PR