

PR #25561 完整报告

sgl-project/sglang

fix(disagg): unstuck decode aborts under prealloc pressure

合并时间: 2026-05-18 22:57

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/25561>

执行摘要

- 一句话: 修复 PD decode 中止请求残留导致超时问题
- 推荐动作: 建议尽快合并, 属于关键 bugfix; 后续可考虑补充相关单元测试。

功能与动机

解决 PD 分离模式下中止的 decode 请求未立即释放, 直到 WAITING_TIMEOUT (~15 分钟) 才被处理的问题。

实现拆解

1. decode.py _update_handshake_waiters: 在 all waiting_for_input 的短路判断中, 增加对任一 receiver 处于 KVPoll.Failed 的检查, 避免已中止请求被忽略。
2. scheduler.py abort_request: 在 DECODE 模式的 prealloc 和 transfer 队列循环中, 调用 kv_receiver.abort() 后, 立即设置 req.finished_reason = FINISH_ABORT(), 确保后续 pop 操作正确移除。
3. tokenizer_manager.py abort_request: 增加空 rid 校验日志; 当 tokenizer_worker_num > 1 时, 即使 rid 不在本地 rid_to_state, 也转发给 scheduler (负载均衡可能导致 abort 发给非 owner worker)。

关键文件:

- python/sglang/srt/managers/scheduler.py (模块 调度器; 类别 source; 类型 core-logic) : 核心 bugfix: 为 prealloc/transfer 队列中的请求设置 finished_reason, 确保被正确弹出。
- python/sglang/srt/managers/tokenizer_manager.py (模块 请求路由; 类别 source; 类型 core-logic) : 修复 abort 转发逻辑: 空 rid 防御, 多 worker 场景无条件转发。
- python/sglang/srt/disaggregation/decode.py (模块 分离架构; 类别 source; 类型 core-logic) : 修复握手轮询短路条件, 避免已失败的 receiver 被跳过。

关键符号: abort_request, _update_handshake_waiters

关键源码片段

[python/sglang/srt/managers/scheduler.py](#)

核心 bugfix: 为 prealloc/transfer 队列中的请求设置 finished_reason, 确保被正确弹出。

```

# scheduler.py abort_request DECODE 分支 (关键改动)
elif self.disaggregation_mode == DisaggregationMode.DECODE:
    # 遍历 prealloc 队列, 中止匹配请求
    for decode_req in self.disagg_decode_prealloc_queue.queue:
        if recv_req.abort_all or decode_req.req.rid.startswith(recv_req.rid):
            logger.debug(f"Abort prealloc queue request. {decode_req.req.rid}")
            decode_req.kv_receiver.abort()
            # 关键修复: 标记 finished_reason, 否则 pop_preallocated 不会丢弃该请求
            if not isinstance(decode_req.req.finished_reason, FINISH_ABORT):
                decode_req.req.finished_reason = FINISH_ABORT()

    # 遍历 transfer 队列, 同理
    for decode_req in self.disagg_decode_transfer_queue.queue:
        if recv_req.abort_all or decode_req.req.rid.startswith(recv_req.rid):
            logger.debug(f"Abort transfer queue request. {decode_req.req.rid}")
            decode_req.kv_receiver.abort()
            if not isinstance(decode_req.req.finished_reason, FINISH_ABORT):
                decode_req.req.finished_reason = FINISH_ABORT()

```

python/sclang/srt/managers/tokenizer_manager.py

修复 abort 转发逻辑: 空 rid 防御, 多 worker 场景无条件转发。

```

# tokenizer_manager.py abort_request (关键改动)
def abort_request(self, rid: str = "", abort_all: bool = False):
    # 防御: 空 rid 且非 abort_all 会误匹配所有请求, 直接拒绝
    if not abort_all and not rid:
        logger.warning("Ignore abort_request with empty rid and abort_all=False")
        return
    # 单 worker 时, 本地 rid 不存在可直接忽略
    if (
        not abort_all
        and self.server_args.tokenizer_worker_num == 1
        and rid not in self.rid_to_state
    ):
        return
    req = AbortReq(rid=rid, abort_all=abort_all)
    self.send_to_scheduler.send_pyobj(req)
    ...

```

python/sclang/srt/disaggregation/decode.py

修复握手轮询短路条件, 避免已失败的 receiver 被跳过。

```

# decode.py _update_handshake_waiters (关键改动)
def _update_handshake_waiters(self, rids_to_check=None):
    if not self.queue:
        return

    # 旧条件: 全 waiting_for_input 则跳过轮询
    # 新条件: 全 waiting_for_input 且没有 receiver 处于 Failed 状态

```

```
# 否则可能错过已中止但未收到 poll 结果的请求
if all(decode_req.waiting_for_input for decode_req in self.queue) and not any(
    getattr(decode_req.kv_receiver, "conclude_state", None) == KVPoll.Failed
    for decode_req in self.queue
):
    return

polls = poll_and_all_reduce(...)
...
```

评论区精华

reviewer ShangmingCai 指出 scheduler.py 中设置 finished_reason 可能与 PD 模块的 prepare_abort 冗余；另一注释不清晰，作者随后删除。最终 reviewer 批准。

- scheduler.py 设置 finished_reason 的冗余性 (design): 作者保留了改动，因为 prepare_abort 在 handshake 阶段调用，而 scheduler 中的处理路径不同，设置 finished_reason 确保后续弹出队列。
- 注释清晰度 (documentation): 作者在后续 commit 中删除了该注释。

风险与影响

- 风险：变更集中在 PD 分离模式 decode 路径的 abort 逻辑，影响范围较小；主要风险为新增设置 FINISH_ABORT 与现有 prepare_abort 重复但无副作用；空 rid 校验添加了防御性代码。无测试配套，但改动验证已有 CI 通过。
- 影响：影响所有使用 PD 分离模式 (decode) 的场景，确保中止请求立刻释放，减少资源浪费和超时等待。对非 PD 模式无影响。
- 风险标记：核心路径变更，缺少测试覆盖

关联脉络

- PR #25599 [PD] Add conclude_state to fake KV backend: 同为 PD 分离架构的 bugfix, 涉及 fake KV backend 的状态管理，与本 PR 的 conclude_state 检查相关。
- PR #25637 Move batch-result processing to SchedulerBatchResultProcessor and retire output_processor mixin: 同为 scheduler.py 重构的一部分，但本 PR 是 bugfix, 修订调度器的 abort 路径。