

PR #29834 完整报告

sgl-project/sglang

Fix scheduler crash on prefill-unreachable decode abort

合并时间: 2026-07-08 14:06

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/29834>

执行摘要

- 一句话: 修复解码崩溃, 清除 pending_reqs 并防御空 receiver
- 推荐动作: 值得快速合入。修复明确且测试充分, 使用 id() 进行身份比较的设计决策值得类似场景参考。

功能与动机

根据 PR 描述: 当预填充启动服务器不可达时, 一个 DecodeRequest 同时存在于 self.queue 和 self.pending_reqs (add() 慢路径)。清除路径 (PR #28022) 将其 kv_receiver 设置为 None, 但只有 self.queue 被清理。剩余的 pending_reqs 引用在下一个 _resolve_pending_reqs 中导致崩溃: AttributeError: 'NoneType' object has no attribute 'abort'。

实现拆解

1. 在 _ensure_prefill_info 中增加保护: 当超时重试失败后, 在调用 decode_req.kv_receiver.abort() 之前检查 kv_receiver is not None, 防止在返回值已清除时崩溃。
2. 在 pop_preallocated 中增加双列表清理: 在移除队列中失败的请求后, 使用 id() 构建失败请求集合, 并从 self.pending_reqs 中过滤掉相同身份的对象, 确保 _resolve_pending_reqs 不再访问已清理的请求。
3. 测试配套: 新增两个单元测试 test_prealloc_abort_also_drops_from_pending_reqs 和 test_ensure_prefill_info_tolerates_cleared_receiver, 分别验证共享对象移除和空 receiver 兼容性。

关键文件:

- python/sglang/srt/disaggregation/decode.py (模块 分解模块; 类别 source; 类型 core-logic; 符号 _ensure_prefill_info, pop_preallocated) : 核心修复文件, 包含两处修改: _ensure_prefill_info 的 None 检查和 pop_preallocated 的 pending_reqs 同步清理。
- test/registered/unit/disaggregation/test_decode_queue_cleanup.py (模块 分解测试; 类别 test; 类型 test-coverage; 符号 test_prealloc_abort_also_drops_from_pending_reqs, BadEqReceiver, eq, test_ensure_prefill_info_tolerates_cleared_receiver) : 新增两个回归测试, 覆盖共享对象移除和空 receiver 兼容性, 确保修复正确性。

关键符号: `_ensure_prefill_info`, `pop_preallocated`,
`test_prealloc_abort_also_drops_from_pending_reqs`,
`test_ensure_prefill_info_tolerates_cleared_receiver`

关键源码片段

[test/registered/unit/disaggregation/test_decode_queue_cleanup.py](#)

新增两个回归测试，覆盖共享对象移除和空 receiver 兼容性，确保修复正确性。

```
def test_prealloc_abort_also_drops_from_pending_reqs(self):
    # Same DecodeRequest lives in both queue and pending_reqs (add() slow
    # path). Aborting must drop it from both, and compare by identity since
    # DecodeRequest's dataclass __eq__ would compare the tensor receiver.
    class BadEqReceiver(FakeReceiver):
        def __eq__(self, other):
            raise TypeError("use identity comparison, not value equality")

        __hash__ = object.__hash__

    receiver = BadEqReceiver()
    req = SimpleNamespace(
        rid="abort-shared",
        finished_reason=FINISH_ABORT("aborted"),
        return_logprob=False,
    )
    decode_req = SimpleNamespace(req=req, kv_receiver=receiver)

    queue = DecodePreallocQueue.__new__(DecodePreallocQueue)
    queue.queue = [decode_req]
    queue.pending_reqs = [decode_req] # same object, dual ownership
    queue.retracted_queue = []
    queue._resolve_pending_reqs = MagicMock()
    queue._update_handshake_waiters = MagicMock()
    queue._uses_swa_tail_prealloc = MagicMock(return_value=False)
    queue._allocatable_token_budgets = MagicMock(return_value=0)
    queue._hicache_pending_restore_tokens = MagicMock(return_value=0)

    scheduler = MagicMock()
    scheduler.running_batch.reqs = []
    scheduler.enable_priority_scheduling = False
    scheduler.enable_hisparse = False
    scheduler.output_streamer = MagicMock()
    queue.scheduler = scheduler

    # Must not raise on the receiver __eq__ above.
    preallocated, failed = queue.pop_preallocated()

    self.assertEqual(preallocated, [])
    self.assertEqual(failed, [decode_req])
```

```
self.assertEqual(queue.queue, [])
self.assertTrue(all(r is not decode_req for r in queue.pending_reqs))
self.assertIsNone(decode_req.kv_receiver)
```

评论区精华

ShangmingCai 对 pending_reqs 清理代码提出 'I think we could drop this', 可能指 if failed_reqs: 条件。作者 amote-i 回应 kv_receiver is not None 检查只是后盾, 真正的修复是清理 pending_reqs, 因为其他调用点仍在访问 pending_reqs。最终决定保留该清理逻辑。引用: 'The kv_receiver is not None check is just a backstop — other call sites (e.g. .init() in _resolve_pending_reqs) still touch entries in pending_reqs, so the cleanup is the actual fix.'

- pending_reqs 清理代码是否需要 if failed_reqs: 条件 (design): 保留 pending_reqs 清理逻辑和空检查。

风险与影响

- 风险: 主要风险是身份比较依赖 id(), 这可能受对象回收和重用影响, 但 DecodeRequest 是长生命周期对象, 风险较低。此外, 其他可能访问 pending_reqs 的路径 (如 _ensure_prefill_info 之外) 也需要确认一致性, 但当前测试覆盖了主要入口。没有引入新配置或外部依赖, 回退简单。
- 影响: 影响范围限于 disaggregation 模式下的解码队列清理; 修复了一个明确的崩溃, 提高了稳定性。对于启用 disaggregation 的用户, 此修复避免了在预填充服务器临时不可达时的调度器崩溃。对不启用 disaggregation 的用户无影响。测试覆盖主要回归场景, 验证了修复有效性。
- 风险标记: 身份比较依赖 id() 的健壮性, 仅修复了 pop_preallocated 路径, 其他 pending_reqs 访问点未审计

关联脉络

- PR #28022 [disaggregation] Decode abort clears kv_receiver: 该 PR 引入了在清除路径中将 kv_receiver 置为 None 的逻辑, 但未同步清理 pending_reqs, 导致当前 PR 修复的崩溃。