PR #21404 完整报告

sgl-project/sglang

fix mamba cache leak when adder fails to add a matched req.

合并时间：2026-03-30 16:45

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/21404

执行摘要

- 一句话：修复 Mamba 缓存内存泄漏问题，确保调度器在添加请求失败时正确释放资源。
- 推荐动作：建议技术管理者和 scheduler 模块开发者精读此 PR，了解 Mamba 缓存泄漏的根因和修复策略，以及 review 中关于资源管理封装的设计讨论。关注 _get_new_batch_prefill_raw 函数的修改点，以掌握调度器中的资源释放时机。

功能与动机

作者在运行中遇到内存泄漏错误，日志显示 'token_to_kv_pool_allocator memory leak detected'，并关联到 Issue 18300。通过调试，发现泄漏发生在 MambaRadixCache 和 PrefillAdder 交互过程中：当请求匹配 Mamba 资源后，若 adder 未能将其添加到运行队列，已分配的 mamba_pool_idx 未被释放，导致内存泄漏。PR body 中详细描述了泄漏的步骤和影响。

实现拆解

修改集中在 python/sglang/srt/managers/scheduler.py 文件的 _get_new_batch_prefill_raw 函数中。关键改动包括：

1. 在循环中添加一个检查，判断请求是否已成功添加到 adder.can_run_list；
2. 如果请求未添加且 mamba_pool_idx 不为 None，则调用 tree_cache.req_to_token_pool.mamba_pool.free 释放资源，并将 req.mamba_pool_idx 设为 None。这通过两个 commit 演进：第一个 commit 添加基础释放逻辑，第二个 commit 根据 review 反馈增加添加检查以避免误释放。

关键文件：

- python/sglang/srt/managers/scheduler.py（模块 scheduler）：核心调度逻辑文件，直接包含修复内存泄漏的关键代码，修改了 _get_new_batch_prefill_raw 函数以添加资源释放逻辑。

关键符号：_get_new_batch_prefill_raw

评论区精华

review 中主要有两个讨论线程：

1. gemini-code-assist[bot] 建议将资源清理逻辑封装到 req 对象的方法中，以提高代码可维护性和封装性；作者未明确采纳，但讨论提供了设计权衡视角。
2. hzh0425 质疑是否在请求已添加但其他条件失败时应避免释放资源，作者通过第二个 commit 添加检查是否已添加来解决此正确性问题，确保只在未添加时释放资源。讨论最终达成一致，PR 被批准。

- 资源清理逻辑封装 (design): 作者未采纳此建议，但讨论提供了设计洞察，强调了资源管理与对象所有权的权衡。
- 添加检查的正确性 (correctness): 作者在第二个 commit 中添加了检查逻辑，确保只在请求未添加时释放资源，review 者认可此修改。

风险与影响

- 风险：主要风险是回归风险：新增的释放逻辑可能在某些边界条件下错误释放或未释放资源，影响调度器正确性。由于修改位于核心调度路径 _get_new_batch_prefill_raw，可能引入性能开销或副作用。缺少专项单元测试覆盖此场景，依赖现有 CI 测试，但 review 中未提及测试补充。风险可控，因为变更针对特定泄漏点且经过 review 验证。
- 影响：直接影响是修复内存泄漏，提升系统内存使用效率和稳定性，避免因泄漏导致的运行时错误或崩溃。对用户透明，改善服务可靠性。对开发者影响较小，代码变更集中且简单，但为 scheduler 模块的资源管理提供了参考模式。影响范围限于调度器和缓存交互，不涉及外部接口。
- 风险标记：核心路径变更，缺少测试覆盖

关联脉络

- PR #21620 fix: Mistral Small 4 fails to start due to config/weight format mismatch: 同为 bugfix PR，但涉及不同模块（模型启动配置），无直接代码关联，仅作为同仓库近期 bugfix 示例。