PR #22767 完整报告

sgl-project/sglang

[HiCache] Fix memory host free logic when share_indices_with_anchor enabled

合并时间：2026-04-15 16:31

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22767

执行摘要

- 一句话：修复 HiCache 在共享索引时内存释放逻辑错误，避免内存泄漏。
- 推荐动作：该 PR 值得精读，因为它揭示了 HiCache 内存池中共享索引模式下的一个关键设计决策：当索引与锚点共享时，释放操作应仅作用于锚点池，避免重复释放导致状态不一致。关注 free 方法的简化如何纠正了原实现中的逻辑错误。

功能与动机

根据 PR body 描述，当 share_indices_with_anchor 设置为 true 时，入口主机池不需要显式的 alloc 或 free 操作。原 free 方法中的冗余释放会导致 free_slots 内存持续增加，引发内存泄漏。

实现拆解

1. 定位问题函数：修改位于 python/sglang/srt/mem_cache/memory_pool_host.py 的 HostPoolGroup.free 方法。
2. 移除冗余循环：删除原方法中遍历 self.entries 并对 share_indices_with_anchor 为 True 的条目执行 entry.host_pool.free(indices) 的循环逻辑。
3. 简化返回逻辑：将方法简化为直接返回 self.anchor_entry.host_pool.free(indices)，移除中间变量 n 和循环后的返回。
4. 无配套改动：本次变更仅涉及核心逻辑修复，未包含测试、配置或文档更新（但 review 中提示文档需同步更新）。

关键文件：

- python/sglang/srt/mem_cache/memory_pool_host.py（模块 内存缓存；类别 source；类型 core-logic；符号 free）：这是唯一修改的文件，包含 HostPoolGroup.free 方法的修复，直接解决内存泄漏问题。

关键符号：free

评论区精华

review 中仅有一条来自 gemini-code-assist[bot] 的评论，指出删除 free 方法中的镜像释放逻辑后，PoolEntry 数据类中关于 share_indices_with_anchor 的文档注释（第 1666 行）已过时且不正确，建议更新以反映镜像释放不再执行。该评论未引发争议，PR 已获 hzh0425 批准

并合并。

- 文档同步问题 (documentation): 未在 PR 中直接解决，但提示了后续需要更新的文档位置。

风险与影响

- 风险：风险较低：
- 回归风险：移除循环释放可能影响其他依赖此行为的代码路径，但根据 PR 描述，当 share_indices_with_anchor 启用时，这些条目本不应被释放，因此修复是必要的。
- 兼容性风险：无，因为这是内部内存管理逻辑调整，不涉及外部 API。
- 性能风险：无，简化逻辑可能轻微提升性能。
- 安全风险：无。 主要风险：文档未同步更新可能导致开发者误解 share_indices_with_anchor 的行为，但代码逻辑本身正确。
- 影响：影响范围：
- 系统影响：修复了 HiCache 内存池在特定配置下的内存泄漏问题，提升系统稳定性。
- 用户影响：对终端用户透明，但可能减少因内存泄漏导致的 OOM 或性能下降。
- 团队影响：开发者需注意 share_indices_with_anchor 的实际行为已变更，需更新相关文档或注释。 影响程度：中等，涉及核心内存管理模块，但变更范围小且针对特定配置。
- 风险标记：核心路径变更，文档未同步

关联脉络

- PR #22862 Streaming session: fix retract tail leak via _free_tail: 同样涉及内存泄漏修复，且都修改了 mem_cache 模块下的文件（session_aware_cache.py），关注 KV 缓存和一致性。
- PR #22753 Fix streaming session busy-check double-counting via active_pool_idxs: 同样修复内存统计问题，涉及 mem_cache 模块和调度逻辑，主题相关。
- PR #22755 Rename _alive_streaming_session_count; use _is_streaming helper: 涉及内存缓存（common.py）和调度器的重构，与本 PR 同属内存管理和会话处理领域。