

PR #27645 完整报告

sgl-project/sglang

[BUG FIX]Fix DSA CPU offload mamba indices signature

合并时间: 2026-06-10 05:03

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/27645>

执行摘要

- 一句话: 修复 DSA CPU offload 缺少 mamba_indices 参数的 bug
- 推荐动作: 该 PR 是典型的边界案例 bugfix, 虽代码量小但在特定场景 (DSA + mamba + decode disaggregation) 上是关键修复。建议快速合入, 并在相关功能测试 (如 DSA CPU 卸载集成测试) 中补充对 mamba_indices 的端到端验证。

功能与动机

Decode disaggregation 在持续负载下可能 retract 运行中的请求并将它们的 KV cache 卸载到 CPU。分配器在 CPU 卸载时会传递可选的 mamba_indices 参数给底层 KV pool。

DSATokenToKVPool 的 get_cpu_copy() 和 load_cpu_copy() 方法没有接受这个参数, 这会在 DSA KV cache 被 retract 期间卸载时引发运行时 TypeError。

实现拆解

1. 修改 DSATokenToKVPool 方法签名: 在 python/sglang/srt/mem_cache/memory_pool.py 中, 为 DSATokenToKVPool.get_cpu_copy 和 load_cpu_copy 分别添加 mamba_indices=None 可选参数。
2. 透传 mamba_indices 给父类: 在这两个方法内部, 调用父类 (如 MLATokenToKVPool) 的对应方法时, 通过 mamba_indices=mamba_indices 将参数透传, 确保父类的 CPU 卸载逻辑能正确处理 mamba 索引。
3. 新增单元测试验证签名: 在 test/registered/unit/mem_cache/test_dsa_pool_host_unit.py 中新建 TestDSAOffloadSignatures 测试类, 使用 inspect.signature 检查 DSATokenToKVPool.get_cpu_copy 和 load_cpu_copy 是否包含 mamba_indices 参数, 确保未来签名不会被意外破坏。
4. 仅修改签名, 不改变语义: 该方法签名修改是向后兼容的, 因为新增参数带有默认值 None, 现有调用无需改动。

关键文件:

- python/sglang/srt/mem_cache/memory_pool.py (模块 内存池; 类别 source; 类型 core-logic; 符号 get_cpu_copy, load_cpu_copy): 核心修复文件: 为 DSATokenToKVPool 的 get_cpu_copy 和 load_cpu_copy 方法添加可选 mamba_indices 参数并透传给父类。

- test/registered/unit/mem_cache/test_dsa_pool_host_unit.py（模块 内存池测试；类别 test；类型 test-coverage；符号 TestDSAOffloadSignatures, test_cpu_copy_methods_accept_mamba_indices）：新增单元测试验证 DSATokenToKVPool 的方法签名包含 mamba_indices 参数。

关键符号：get_cpu_copy, load_cpu_copy

关键源码片段

python/sglang/srt/mem_cache/memory_pool.py

核心修复文件：为 DSATokenToKVPool 的 get_cpu_copy 和 load_cpu_copy 方法添加可选 mamba_indices 参数并透传给父类。

```
# 在 DSATokenToKVPool 中修复方法签名缺失
# 问题: decode disaggregation retract 时, allocator 会传递 mamba_indices
# 但 get_cpu_copy 和 load_cpu_copy 未定义该参数, 导致 TypeError

def get_cpu_copy(self, indices, mamba_indices=None):
    # DSA 额外维护 index_k_with_scale_buffer, retract 时必须一并卸载
    # 否则恢复时 index 数据错乱导致注意力计算错误
    kv_cache_cpu = super().get_cpu_copy(indices, mamba_indices=mamba_indices) # 透传
    # ... 原有 index_k 卸载逻辑保持不变 ...
    return {"kv": kv_cache_cpu, "index_k": index_k_cpu}

def load_cpu_copy(self, kv_cache_cpu_dict, indices, mamba_indices=None):
    super().load_cpu_copy(
        kv_cache_cpu_dict["kv"], indices, mamba_indices=mamba_indices # 透传
    )
    # ... 原有 index_k 加载逻辑保持不变 ...
```

评论区精华

该 PR 无 review 评论，快速获批。合并者 hnyls2002 添加了 /tag-and-rerun-ci 命令的评论，表明 CI 流程已触发。

- 暂无高价值评论线程

风险与影响

- 风险：
 1. 回归风险低：变更仅涉及添加可选参数和透传，没有改变现有逻辑或控制流，对非 mamba 场景无影响。
 2. 测试覆盖：新增的单元测试仅验证签名是否存在，未测试实际功能路径（如 mamba_indices 被正确使用）。不过现有 TestDSAHiCacheTransfer 仍覆盖 CPU 卸载 / 加载功能，因此整体风险可控。
 3. 兼容性：新增参数有默认值 None，所有现有调用点无需修改，向后兼容。
- 影响：

1. 用户：修复后，使用 DSA KV cache 且启用 decode disaggregation 的场景下，retract 操作不会再因缺少 mamba_indices 参数而崩溃。
2. 系统：影响范围限于 DSATokenToKVPool 类的 CPU 卸载路径，仅在 DSA 后端和 mamba 模型组合时触发。
3. 团队：代码变更小且直观，维护成本低。 - 风险标记：新增测试仅验证签名未覆盖功能逻辑

关联脉络

- PR #27550 fix(hiradix): wait for extra pool IO: 同属 KV cache 卸载 / 加载路径的 bugfix，涉及 HiRadix 缓存。