PR #22979 完整报告

sgl-project/sglang

[HiSparse]: Adding e2e ut for hisparse

合并时间：2026-04-16 23:20

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22979

执行摘要

- 一句话：为 HiSparse 分层稀疏 KV 缓存系统添加端到端单元测试。
- 推荐动作：对于关注测试设计或 HiSparse 模块的工程师，此 PR 提供了单元测试的参考实现，展示了如何构建最小化组件栈进行端到端测试，值得参考以了解 HiSparse 系统测试策略。

功能与动机

根据 PR 标题 '[HiSparse]: Adding e2e ut for hisparse' 和文件内容，动机是增加对 HiSparse 功能的测试覆盖，确保其在 CUDA/ROCm 环境下的正确性，并通过 CI 集成验证，提升代码可靠性。

实现拆解

1. 创建测试文件：新增 test/registered/unit/managers/test_hisparse_unit.py，定义测试配置常量（如 SIZE=2048、PAGE_SIZE=64）和辅助函数 _make_req 用于模拟请求对象。
2. 定义测试类：实现 TestHiSparseUnit 类，继承 unittest.TestCase，包含 setUpClass 方法初始化 HiSparse 组件栈（如设备池、分配器）、tearDownClass 清理资源，并注册 CI 通过 register_cuda_ci。
3. 实现测试方法：添加多个测试方法如 _alloc_req_slot、_free_req_slot、_alloc_kv，覆盖内存分配、请求生命周期和批处理正确性。
4. CI 集成：通过 register_cuda_ci(est_time=20, suite="stage-b-test-1-gpu-small") 将测试注册到 CI 的 GPU 小规模测试阶段，确保测试在合适环境下运行。

关键文件：

- test/registered/unit/managers/test_hisparse_unit.py（模块 测试单元；类别 test；类型 test-coverage；符号 _make_req, TestHiSparseUnit, setUpClass, tearDownClass）：新增的单元测试文件，覆盖 HiSparse 系统的端到端测试，是 PR 的唯一变更，定义了配置、辅助函数和测试类。

关键符号：_make_req, TestHiSparseUnit.setUpClass, TestHiSparseUnit.tearDownClass, TestHiSparseUnit.setUp, _alloc_req_slot, _free_req_slot, _alloc_kv

关键源码片段

test/registered/unit/managers/test_hisparse_unit.py

新增的单元测试文件，覆盖 HiSparse 系统的端到端测试，是 PR 的唯一变更，定义了配置、辅助函数和测试类。

```
@classmethod
def setUpClass(cls):
    """设置测试类的全局环境，初始化 HiSparse 组件栈。"""
    # 检查 CUDA 可用性和平台支持，跳过不兼容环境
    if not torch.cuda.is_available():
        raise unittest.SkipTest("CUDA is required for HiSparse tests.")
    if is_npu() or is_xpu():
        raise unittest.SkipTest("HiSparse tests only support CUDA/ROCm.")
    if not (is_cuda() or is_hip()):
        raise unittest.SkipTest("CUDA/ROCm not available.")

    # 初始化分布式环境，用于 TP 组，模拟单机单进程
    os.environ.setdefault("MASTER_ADDR", "127.0.0.1")
    os.environ.setdefault("MASTER_PORT", "29599")
    if not torch.distributed.is_initialized():
        torch.distributed.init_process_group(backend="gloo", rank=0, world_size=1)
    cls.tp_group = torch.distributed.group.WORLD

    # 替换内存分配函数为 pin memory 版本，以支持测试中的特殊内存管理
    from sglang.srt.mem_cache.memory_pool_host import (
        ALLOC_MEMORY_FUNCS,
        alloc_with_pin_memory,
    )
    cls._original_alloc = ALLOC_MEMORY_FUNCS["cuda"]
    ALLOC_MEMORY_FUNCS["cuda"] = alloc_with_pin_memory

    # 根据平台设置页面大小：ROCm 使用 1，CUDA 使用预定义 PAGE_SIZE
    global_page_size = 1 if is_hip() else PAGE_SIZE

    # 初始化 HiSparse 设备池和分配器，使用测试配置常量
    from sglang.srt.mem_cache.hisparse_memory_pool import (
        HiSparseNSATokenToKVPool,
        HiSparseTokenToKVPoolAllocator,
    )
    cls.device_pool = HiSparseNSATokenToKVPool(
        size=SIZE,
        page_size=global_page_size,
        kv_lora_rank=KV_LORA_RANK,
        dtype=torch.bfloat16,
        qk_rope_head_dim=QK_ROPE_HEAD_DIM,
        layer_num=LAYER_NUM,
        device="cuda",
        index_head_dim=128,
        enable_memory_saver=False,
```

```
        kv_cache_dim=KV_CACHE_DIM,
        host_to_device_ratio=HOST_TO_DEVICE_RATIO,
    )
    cls.allocator = HiSparseTokenToKVPoolAllocator(
        size=SIZE,
        page_size=global_page_size,
        dtype=torch.bfloat16,
        device="cuda",
        kvcache=cls.device_pool,
        need_sort=False,
        host_to_device_ratio=HOST_TO_DEVICE_RATIO,
    )
```

评论区精华

review 评论为空，只有 ispobock 的批准，表明变更被直接接受，无争议讨论或设计权衡。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低，主要为测试执行可能增加 CI 时间（估计 20 秒），并在非 CUDA/ROCm 环境下跳过测试，依赖特定硬件环境可能导致测试覆盖率不完整。无直接影响生产代码，但测试失败可能干扰 CI 流水线。
- 影响：对用户无直接影响；对开发团队，提高了 HiSparse 模块的测试覆盖率，有助于早期发现回归问题，增强代码可靠性和维护性；对 CI 流程，增加了 GPU 测试负载，但通过小规模配置控制影响范围。
- 风险标记：增加 CI 执行时间，依赖特定硬件环境

关联脉络

- PR #22592 [BugFix][RadixTree]:Fix stale eviction assertion in HiMambaRadixCache host eviction path: 同属 KV 缓存系统模块的维护性工作，当前 PR 为 HiSparse 添加测试，而 PR 22592 修复了相关缓存组件 HiMambaRadixCache 的 bug，两者均关注缓存系统的正确性和稳定性。