PR #31173 完整报告

sgl-project/sglang

[PD] Stride KV token->page indices on device before D2H copy

合并时间：2026-07-15 01:08

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/31173

执行摘要

- 一句话：设备上步进 KV token 到页索引，减少 D2H 拷贝量
- 推荐动作：值得精读：展示了如何通过调整 D2H 拷贝时机来优化性能，变更量小但收益明显，适合学习性能优化思路。

功能与动机

PR body: 'kv_to_page_indices copied every KV token index to host and then strided on CPU, so the D2H traffic on every PD send_kv_chunk / send_metadata scaled with the number of tokens. Stride and divide on the device tensor first and copy only the resulting ~num_pages indices to host — roughly page_sizex less D2H per send.'

实现拆解

1. 修改核心函数：在 python/sglang/srt/mem_cache/common.py 中，kv_to_page_indices 函数签名由接受 np.ndarray 改为接受 torch.Tensor，并在设备上执行 [::page_size] // page_size 操作后才调用 .cpu().numpy()，移除了 page_size == 1 的特判分支。
2. 清理调用方：在 python/sglang/srt/disaggregation/decode.py 和 python/sglang/srt/disaggregation/prefill.py 中，移除所有调用处对切片结果显式的 .cpu().numpy() 调用，直接将设备张量传递给 kv_to_page_indices；同时将 astype(np.int32) 移至 decode 侧 send_metadata 前的最终结果上，确保 ZMQ 序列化兼容。

关键文件：

- python/sglang/srt/mem_cache/common.py（模块 KV 缓存；类别 source；类型 core-logic；符号 kv_to_page_indices）：核心函数 kv_to_page_indices 的变更所在，优化了 PD 路径的 D2H 效率
- python/sglang/srt/disaggregation/decode.py（模块 解耦解码；类别 source；类型 core-logic）：移除多处不必要的 .cpu().numpy() 调用，调整 astype(np.int32) 位置，配合 kv_to_page_indices 变更
- python/sglang/srt/disaggregation/prefill.py（模块 解耦预填充；类别 source；类型 core-logic）：移除不必要的 .cpu().numpy() 调用，直接传递设备张量，并调整 kv_indices 获取位置

关键符号：kv_to_page_indices

关键源码片段

python/sglang/srt/mem_cache/common.py

核心函数 kv_to_page_indices 的变更所在，优化了 PD 路径的 D2H 效率

```
# file: python/sglang/srt/mem_cache/common.py

def kv_to_page_indices(kv_indices: torch.Tensor, page_size: int) -> np.ndarray:
    """
    在设备上步进和除以 page_size，仅将约 num_pages 个索引拷贝到主机，
    相比旧实现（逐个 token 拷贝并步进）减少了约 page_size 倍的 D2H 流量。
    旧实现接受 np.ndarray 并假设数组已在主机上。
    """
    # 在设备上执行步进和除法，结果张量留在设备上
    # kv_indices 是连续的 token 索引（每页 page_size 个 token），
    # kv_indices[::page_size] 取每页第一个 token 的索引，
    # // page_size 将其转换为物理页号。
    device_result = kv_indices[::page_size] // page_size
    # 仅将约 num_pages 个结果拷贝到主机，并转为 numpy 数组
    return device_result.cpu().numpy()
```

评论区精华

review 中 gemini-code-assist[bot] 建议为 kv_to_page_indices 添加类型注解（接受 torch.Tensor/np.ndarray，返回 np.ndarray），以提升可读性和类型安全。该建议在 PR 合并前未采纳。

- 建议添加类型注解 (style): PR 合并时未采纳该建议，函数签名未添加类型注解。

风险与影响

- 风险：风险较低。主要潜在风险： 1) 若其他未修改的代码仍传递 np.ndarray 给 kv_to_page_indices 会因类型不匹配出错，但仓库内所有调用点已修改； 2) 移除 page_size == 1 分支后多了一次不必要的除法，性能影响微乎其微； 3) ZMQ 序列化要求 int32，state payload 路径未显式转换，与旧行为一致（旧代码也未转换），维持兼容。
  - 影响：对用户：减少 PD 传输延迟，降低 D2H 带宽占用，大型序列场景收益更明显。对团队：代码简化，维护成本降低。对其他系统：无影响。
  - 风险标记：核心路径变更，缺少测试配套

关联脉络

- PR #30937 fix: avoid double KV release on disaggregated prefill grammar errors: 同样修改了 python/sglang/srt/disaggregation/prefill.py，与当前 PR 同属 disaggregation 路径优化 / 修复