PR #24587 完整报告

sgl-project/sglang

[AMD][aiter] Fix cuda_graph_kv_indices OOB under page_size>1

合并时间：2026-05-23 14:19

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24587

执行摘要

- 一句话：修复 Aiter 后端 page_size>1 时 KV indices 越界
- 推荐动作：该 PR 是关键的 bugfix，涉及显存安全，建议尽快合并并 cherry-pick 到稳定分支。开发者应关注 reviewer 提到的其他位置（max_kv_len 计算）是否存在类似问题，后续可能需要进一步修复。

功能与动机

Aiter attention backend 在 ROCm 上运行时，当 seq_lens_sum 增长超过 CUDA graph KV indices 缓冲区大小时会崩溃，表现为 HSA_STATUS_ERROR_MEMORY_APERTURE_VIOLATION 或 HIP error: an illegal memory access was encountered。根因是 init_cuda_graph_state 分配的缓冲区按页面粒度计算，但写入是按 token 粒度进行的，导致 page_size>1 时缓冲区不足。

实现拆解

1. 修复缓冲区分配大小：在 aiter_backend.py 的 init_cuda_graph_state 中，将 cuda_graph_kv_indices 的分配从 max_bs * max_num_blocks_per_seq 改为 max_bs * max_num_blocks_per_seq * self.page_size，确保能容纳 token 级写入。
2. 新增独立页面表缓冲区：为 use_triton_unified_attention 路径分配独立的 cuda_graph_page_table，尺寸为 (max_bs, max_num_blocks_per_seq)，避免与 token 级缓冲区混淆。
3. 修改 unified attention 路径：在 init_forward_metadata_capture_cuda_graph 和 init_forward_metadata_replay_cuda_graph 中，将原本通过 cuda_graph_kv_indices.view(-1, max_num_blocks_per_seq) 获取页面表的操作，改为直接使用新的 cuda_graph_page_table。
4. 添加TODO注释：记录未来彻底支持page_size>1所需的工作（per-pageindiceskernel、metadata 修改等）。

关键文件：

- python/sglang/srt/layers/attention/aiter_backend.py（模块 注意力后端；类别 source；类型 core-logic；符号 init_cuda_graph_state, init_forward_metadata_capture_cuda_graph, init_forward_metadata_replay_cuda_graph）：唯一修改的文件，包含所有修复逻辑：缓冲区分配修正、新增独立页面表、修改 capture/replay 方法中的页面表引用。

关键符号：init_cuda_graph_state, init_forward_metadata_capture_cuda_graph, init_forward_metadata_replay_cuda_graph

评论区精华

Reviewer kkHuang-amd 指出在 aiter_backend.py 第 2675 行和第 2922 行（forward_decode 和 forward_extend 中）也存在 max_kv_len = page_table.shape[1] * self.page_size 的计算，在 page_size>1 时会导致错误。但这些评论并未在后续 commit 中处理，且 PR 最终被合并，可能这些位置与本次修改的缓冲区分配问题不直接相关，或已单独修复。

- 其他位置存在类似问题 (correctness): 未在 PR 中处理这些位置，可能这些位置与本次修改不直接相关，或已在后续独立修复。PR 作者未回应。

风险与影响

- 风险：主要风险是缓冲区分配增加（乘以 page_size），可能带来显存开销。但该缓冲区大小原本就是按 token 粒度需求设计的，乘以 page_size 后符合预期。另外，新增的 cuda_graph_page_table 会额外分配 max_bs * max_num_blocks_per_seq 个 int32 元素，显存占用不大。修改集中在 init_cuda_graph_state 及相关 capture/replay 方法，不影响其他 attention backend。
- 影响：影响范围限于使用 --attention-backend aiter 且 --page-size > 1 的用户，修复了显式内存访问越界导致的崩溃，使此类配置变得可用。对 --page-size=1 无影响（buffer_numel 不变）。不会影响其他 attention backend。
- 风险标记：reviewer 指出遗漏修复，仅测试环境验证

关联脉络

- PR #20978 Pad max_bs for higher cuda-graph coverage: PR body 中提及，该 PR 引入了 padding max_bs 的逻辑，与当前 PR 的初始化代码相邻。