PR #34837 完整报告

sgl-project/sglang

[AMD] Add concat_and_cast_mha_k_pad_kernel to support 12-head and enable K3 aiter prefill kernel

合并时间：2026-08-16 05:39

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/34837

执行摘要

- 一句话：新增 pad 内核，支持 K3 12 头开启 AITER prefill
- 推荐动作：建议精读，核心在于处理 Triton tl.arange 非 2 的幂限制的通用技巧。值得关注的设计决策是保留原内核、仅通过 dispatch 新增 pad 分支，降低回归风险。后续若要在其他模型上启用类似优化，可复用该模式。

功能与动机

PR 正文指出 Triton 的 extend_attention_fwd 在 MLA prefill 上明显慢于 AITER 内核，而 K3 在 TP8 下每 GPU 12 头，现有 concat_and_cast_mha_k_kernel 因 tl.arange 要求 2 的幂范围而无法使用。因此需要新增支持非 2 的幂头数的 pad 内核，以启用 AITER prefill 内核。

实现拆解

1. 在 python/sglang/kernels/ops/kvcache/cache_ops.py 中新增 concat_and_cast_mha_k_pad_kernel Triton 内核，用 HEAD_BLOCK=triton.next_power_of_2(head_cnt) 铺开线程并掩码尾部，绕过 tl.arange 的 2 的幂限制。
2. 在 concat_and_cast_mha_k_triton 中增加头数判断，当 head_cnt != triton.next_power_of_2(head_cnt) 时跳转到 pad 内核；原内核和调用点保持不变，避免回归。
3. 通过环境变量 SGLANG_AITER_FP8_PREFILL_ATTN=0 关闭不支持 12 头的 FP8 kernel，并用 --prefill-attention-backend aiter 启用 AITER 后端。
4. 验证：GSM8k 准确率 95.1%，TTFT 提升 14.9%、吞吐量提升 4.6%（8k1k，cc2~64）。

关键文件：

- python/sglang/kernels/ops/kvcache/cache_ops.py（模块 缓存层；类别 source；类型 core-logic；符号 concat_and_cast_mha_k_pad_kernel, concat_and_cast_mha_k_triton）：新增 pad 内核并添加 dispatch 逻辑，是本次变更的核心文件。

关键符号：concat_and_cast_mha_k_pad_kernel, concat_and_cast_mha_k_triton

评论区精华

该 PR 无任何 review 评论，HaiShaw 直接批准合并，没有公开的设计争议。

- 暂无高价值评论线程

风险与影响

- 风险：新内核仅在头数非 2 的幂时启用，原有 2 的幂路径不受影响，回归风险低。但该改动没有配套单元测试，且依赖 SGLANG_AITER_FP8_PREFILL_ATTN=0 环境变量；若用户忘记设置，可能出现 FP8 kernel 不兼容或性能回退。此外新路径仅在 AMD 上验证，其他硬件平台未测试。
- 影响：对 Kimi-K3 用户：在 AMD 上 prefill 延迟降低（TTFT -14.9%），吞吐量提升（+4.6%），但需要额外配置环境变量。对代码库：仅新增一个内核分支，对其他模型无影响。对团队：AMD 内核路径的维护需要关注非 2 的幂头数场景。
- 风险标记：缺少测试覆盖，依赖环境变量配置，仅 AMD 路径验证

关联脉络

- PR #34517 [AMD][Spec] Accelerate Qwen3.5 verification with grouped-head shared KV: 同属 AMD 注意力内核性能优化方向，关注共享 KV 和分组头；但场景不同（speculative verification vs MLA prefill）。