PR #33981 完整报告

sgl-project/sglang

[AMD] Add K3 verified mla kernel for DSpark on triton backend

合并时间：2026-08-08 13:59

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/33981

执行摘要

- 一句话：为 K3 DSpark 新增 MLA 验证 Triton 内核，最高提速 2.42x
- 推荐动作：该 PR 值得精读，尤其是 verify_mla.py 中针对 MLA 结构做的两阶段并行设计和 GEMM 拆分技巧，是理解如何为特殊 attention 结构定制高性能 kernel 的极佳示例。关注点包括：共享 latent 头块复用、floor 分区策略、bf16 中间存储的权衡。同时建议关注后续扩展到 gfx1250/942 的计划，以及是否需要补充自动化测试。

功能与动机

PR 描述指出：Kimi-K3 DSpark 吞吐较慢，高并发下甚至比非 DSpark 设置更慢。根因是 target-verify 注意力走了 verify_splitkv 内核，该内核针对标准 MHA 设计，每个 query head 启动一个程序，导致共享的 MLA latent（h_kv=1）被每个 head 重复加载（TP8 时约 12 次冗余加载）。在内存受限的验证场景中，这一步成为性能瓶颈，且随并发数增长而恶化。

实现拆解

实现分为两个主要部分：

1. 新增 python/sglang/kernels/ops/attention/verify_mla.py，实现 MLA 专用 split-KV 验证内核。 - 网格设计：grid 为 (bs, n_head_blocks, split)，每个程序处理 BLOCK_H 个 query head 和所有 draft queries，使共享 latent 只加载一次并在多个 head 间复用。 - GEMM 拆分：将 576 维 QK 点积拆为 nope（512）和 pe（64）两个 2 的幂次 GEMM，避免 576 填充到 1024 的浪费。 - Split 分配：使用 floor 划分（seqlen // active），最后一个 split 吸收余数，保证每个 active split 非空。 - 部分输出：stage-1 输出以 bf16 存储（lse 保持 fp32），减少一半的寄存器往返带宽。 - 两阶段架构：_verify_mla_prefix_stage1 计算每个 split 的部分 attention，_verify_mla_combine_stage2 合并 split 结果并进行 draft-draft 因果注意力。
2. 修改 python/sglang/srt/layers/attention/triton_backend.py，将新内核挂载到前向路径。 - 添加 verify_mla_fwd 的延迟导入和 torch.compiler.disable() 包装。 - 新增 self.use_verify_mla，门控条件为 is_gfx95_supported() and topk == 1 and self.use_mla and is_kimi_k3(model_runner.model_config.hf_config)。 - 在 forward_extend 的目标验证分支中，先尝试 verify_mla_fwd，失败或不适配时回退到 verify_splitkv_fwd，再回退到 extend_attention_fwd。

3. 测试配套：PR 未新增独立的单元测试文件，但提供了 GSM8k 精度测试（准确率 0.951）和并发基准（2~32 并发）。

关键文件：

- python/sglang/kernels/ops/attention/verify_mla.py（模块 注意力内核；类别 infra；类型 core-logic；符号 block_config, _active_splits, _verify_mla_prefix_stage1, _verify_mla_combine_stage2）：新增的 MLA 专用 split-KV 验证内核，是本 PR 的核心性能优化载体，包含两阶段架构、GEMM 拆分和 head-block 复用等关键设计。
- python/sglang/srt/layers/attention/triton_backend.py（模块 注意力后端；类别 source；类型 dependency-wiring；符号 TritonAttnBackend.init, TritonAttnBackend.forward_extend）：将新内核接入目标验证路径，并添加 Kimi-K3 专属门控（use_verify_mla），保证仅对 K3 MLA 生效，不干扰其他模型。

关键符号：block_config, _active_splits, _verify_mla_prefix_stage1, _verify_mla_combine_stage2, verify_mla_fwd, TritonAttnBackend.init, TritonAttnBackend.forward_extend

关键源码片段

python/sglang/srt/layers/attention/triton_backend.py

将新内核接入目标验证路径，并添加 Kimi-K3 专属门控（use_verify_mla），保证仅对 K3 MLA 生效，不干扰其他模型。

```
# python/sglang/srt/layers/attention/triton_backend.py
# 在 TritonAttnBackend.__init__ 中，新增 MLA 验证内核的导入与包装
from sglang.kernels.ops.attention.verify_mla import verify_mla_fwd
from sglang.kernels.ops.attention.verify_splitkv import verify_splitkv_fwd

self.verify_splitkv_fwd = torch.compiler.disable(verify_splitkv_fwd)
# MLA split-KV EAGLE-verify 内核；仅 topk==1 时启用
self.verify_mla_fwd = torch.compiler.disable(verify_mla_fwd)
…

# 门控：仅在 gfx95 + topk==1 + MLA + Kimi-K3 时启用新内核，
# 避免对未经过验证的模型 / 平台造成回归
self.use_verify_mla = (
    is_gfx95_supported()
    and self.topk == 1
    and self.use_mla
    and is_kimi_k3(model_runner.model_config.hf_config)
)
…

# forward_extend 的目标验证分支：优先使用 MLA 内核，否则回退到 splitkv，
# 两者均返回布尔值表示是否成功执行，失败时继续走 extend_attention_fwd
if self.use_verify_mla:
    verify_fwd = self.verify_mla_fwd
elif self.use_verify_splitkv:
```

```
        verify_fwd = self.verify_splitkv_fwd
    else:
        verify_fwd = None

    if (
        verify_fwd is not None
        and score_mod is None
        and forward_batch.forward_mode.is_target_verify()
        and verify_fwd(...) # 传入与 verify_splitkv 相同的参数
    ):
        return o
```

评论区精华

唯一的 review 来自合并者 HaiShaw，给出了 APPROVED 并附带说明：

> "gfx950 gated. LGTM. @1am9trash extend it to gfx1250, gfx942 later."

这表明审查者确认了 AMD gfx95 平台门控的正确性，同时提出后续应扩展到 gfx1250 和 gfx942 平台的期望。没有其他争议性讨论。

- 内核平台扩展性（gfx1250/gfx942） (design): 当前仅 gfx95 启用，扩展留待后续 PR。

风险与影响

- 风险：
  - 平台特定性：该内核仅在 AMD gfx95（is_gfx95_supported()）且 Kimi-K3 模型、topk==1 的 MLA 场景下启用，其他平台 / 模型不受影响，但若未来 gfx 1250/942 复用此路径，需要重新验证。
  - 正确性依赖 can_handle 的边界条件：虽然 PR 通过 GSM8k 验证了基本正确性，但 verify_mla_fwd 的 can_handle 门控（非因果、sink、滑动窗口、ragged、topk>1 等情况）与 verify_splitkv 一脉相承，若服务端出现 PR 未覆盖的场景（如特殊 mask），可能回退到 extend 路径，但不会出错。
  - 数值精度：部分输出使用 bf16 存储，可能产生轻微精度损失，但 GSM8k 准确率保持 0.951，说明影响可忽略；但其他任务（如数学推理、长文本生成）可能需要更多验证。
  - 缺少自动化测试：PR 未添加单元测试文件，kernel 的正确性主要依赖人工基准和 evals，长期回归风险存在。
- 影响：
  - 用户影响：Kimi-K3 DSpark 用户在高并发场景下可获得显著吞吐提升（ITL 最高 2.42x，TTT 最高 1.77x），直接削减服务延迟和运营成本。
  - 系统影响：修改位于目标验证注意力路径，仅影响 AMD gfx95 上的 Kimi-K3 DSpark 配置；其他注意力后端和平台不受影响。
  - 团队影响：为后续其他 MLA 模型（如 DeepSeek 系列）引入类似优化提供了可参考的模板，也明确了将 MLA 内核扩展到更多 AMD 平台的路线。
  - 风险标记：核心路径变更，缺少测试覆盖，特定平台门控，数值精度风险

关联脉络

- PR #33898 [inkling] Render tool-result media instead of coercing content to str: 同属 Kimi 系列模型优化（Inkling），但重点不同，关联度较低。