PR #34349 完整报告

sgl-project/sglang

[Diffusion] Tune QK head LayerNorm for SM120

合并时间：2026-08-12 10:41

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/34349

执行摘要

- 一句话：调优 SM120 Diffusion 内核启动，RTX 5090 提速约 10.9%
- 推荐动作：建议内核调优和 Diffusion 性能相关工程师快速阅读。值得借鉴的是：用寄存器数 / 占用率分析定位瓶颈、用穷举行与 warp 扫描确定架构特定配置，并刻意保持 SM90 启动字节级不变以控制回归风险。该 PR 本身不涉及架构级设计，无需精读。

功能与动机

PR 正文指出原启动是为 H200 调优的（每 Triton program 64 行），在 SM120 上会使用 142 个寄存器 / 线程，把内核限制在 25% 理论 / 23.11% 实际占用率，明显慢于更小的 SM120 行分组。为了让 RTX 5090 等 SM120 设备获得接近理论性能，需要按架构选择行数与 warp 数。

实现拆解

1. 新增架构探测函数：在 python/sglang/kernels/ops/diffusion/triton/layernorm_modulate.py 中新增 _is_sm120_or_newer()，调用 sglang.kernels.jit.utils.get_jit_cuda_arch() 获取当前 JIT 编译目标架构，并通过 arch.major * 10 + arch.minor >= 120 判断是否为 SM120 及更新架构。
2. 接入启动参数分支：在 fused_qk_head_layernorm() 中，将原本固定的 rows = 64 改为 rows = 8 if is_sm120 else 64，num_warps 同理改为 4 if is_sm120 else 2；SM90 启动配置逐字节保持不变，避免影响 H100/H200。
3. 基准与扫描验证：对生产形状 (1, 4360, 32, 128) 做穷举行 /warp 扫描，用 CUDA-event 中位数测延迟（PR 说明 NCU 回放对更大 grid 的候选配置惩罚不成比例）；NCU 数据确认候选配置寄存器 37/ 线程、实际占用率 93.28%。
4. 测试配套：未新增测试文件，复用现有 bit-exact 测试：RTX 5090 变更路径 14 项通过，H200 变更路径 1265 项通过，H100 文件隔离扩散套件 3227 项通过（22 项跳过），确认融合内核与 eager 参考逐位一致。

关键文件：

- python/sglang/kernels/ops/diffusion/triton/layernorm_modulate.py（模块 内核模块；类别 infra；类型 performance-tuning；符号 _is_sm120_or_newer, fused_qk_head_layernorm）：唯一修改文件：新增 _is_sm120_or_newer() 架构探测，并让 fused_qk_head_layernorm() 在 SM120 上使用 8 行 /4 warp 启动配置，在保持

SM90 字节级不变的同时提升 RTX 5090 占用率与延迟。

关键符号：fused_qk_head_layernorm，_is_sm120_or_newer

关键源码片段

python/sglang/kernels/ops/diffusion/triton/layernorm_modulate.py

唯一修改文件：新增 _is_sm120_or_newer() 架构探测，并让 fused_qk_head_layernorm() 在 SM120 上使用 8 行 /4 warp 启动配置，在保持 SM90 字节级不变的同时提升 RTX 5090 占用率与延迟。

```
# python/sglang/kernels/ops/diffusion/triton/layernorm_modulate.py
# 融合 Q/K head LayerNorm：与 eager per-head LayerNorm 保持逐位一致。

from sglang.kernels.jit.utils import get_jit_cuda_arch


def _is_sm120_or_newer() -> bool:
    # SM120（RTX 5090）每 SM 寄存器文件比 H200 小，需要更小的行分组。
    # 用 major * 10 + minor 把架构号拉平成可比较的整数。
    arch = get_jit_cuda_arch()
    return arch.major * 10 + arch.minor >= 120


def fused_qk_head_layernorm(q, k, ...):
    # 在 CUDA 路径上按架构挑选启动配置：
    # SM90 保持 H200 调优的 64 行 / 2 warp 不变；
    # SM120 采用 8 行 / 4 warp，让寄存器占用从 142 降到 37。
    head_dim = q.shape[-1]
    n_rows = q.numel() // head_dim

    is_sm120 = _is_sm120_or_newer()
    rows = 8 if is_sm120 else 64
    q_out = torch.empty_like(q)
    k_out = torch.empty_like(k)
    with torch.cuda.device(q.device):
        # 其余启动参数与 SM90 完全一致，仅 ROWS 与 num_warps 按架构切换。
        kernel[(n_rows // rows,)](
            q, q_out, k, k_out,
            ROWS=rows,
            num_warps=4 if is_sm120 else 2,
        )
    return q_out, k_out
```

评论区精华

该 PR 没有任何 review 评论或审核线程；唯一评论来自作者 BBuf 附带的 CI 运行链接（https://github.com/sgl-project/sglang/actions/runs/31513291031/job/93978275236?pr=34349），因此不存在记录在案的设计权衡讨论。基准方法本身在 PR 正文中有说明：延迟取

CUDA-event 中位数，因为 NCU 回放对候选配置（更大 program grid）的惩罚不成比例。

- 暂无高价值评论线程

风险与影响

- 风险：风险集中在架构分支本身：_is_sm120_or_newer() 依赖 get_jit_cuda_arch() 的返回值，若 JIT 环境未正确设置架构，可能走到错误分支；另外 8 行 /4 warp 仅在 SM120 上实测，未来更新的 SM 产品也会落入该分支，存在未经调优的隐患，但通常不会比 64 行配置更差。CI 记录显示 Base/Extra 两轮测试标红，但 PR 已合并且讨论中未解释原因，需在合并后留意 nightly 结果。功能层面无需担心位精度，三套测试已覆盖 SM90 与 SM120。
- 影响：影响范围集中在 Diffusion 推理路径中调用 fused_qk_head_layernorm() 的模型（如 ERNIE 等 DiT）：SM120 用户获得约 10.9% 的内核级延迟收益与显著更高的占用率；SM90（H100/H200）行为完全不变；无 API、配置或依赖变更，团队无需额外迁移。改动仅 1 个文件、15 行，回归面小。
- 风险标记：SM120+ 未测新架构分支，依赖 get_jit_cuda_arch 探测，CI 记录存在失败运行

关联脉络

- PR #34347 [Diffusion][MiniMax H3] Fix SM120 QKNorm+RoPE rounding: 同为 SM120 上 Diffusion QK 相关内核的精度修复，属于同一架构调优 / 修复脉络。
- PR #34412 [Diffusion] Improve bit-exact fusion fallback diagnostics: 同一 Diffusion 融合内核基础设施，关注 bit-exact 回退诊断，与本 PR 的位精度测试策略互补。
- PR #34314 [diffusion] Ideogram-4: fuse Qwen3-style RoPE and SwiGLU silu-mul (denoise -5.1% H100 / -4.7% H200, bit-exact): 同属 Diffusion 融合内核性能优化并保持 bit-exact，体现 Diffusion JIT 内核持续调优的演进方向。