

PR #27581 完整报告

sgl-project/sglang

[AMD] fix: handle per-frame 4D shift in native scale-shift kernel

合并时间: 2026-06-10 01:31

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/27581>

执行摘要

- 一句话: 修复 ROCm 上 4D per-frame shift 形状处理
- 推荐动作: 该 PR 值得 CI 维护者和 AMD 平台开发者精读。核心价值在于: 1) 展示了如何通过平台条件编译修复跨后端差异; 2) 揭示了 Triton kernel 在形状假设上的潜在陷阱; 3) 提供了清晰的 root cause 分析和测试验证方法。

功能与动机

LingBot-World 模型 PR (#26954) 引入 causal LingBot 模型后, `multimodal-gen-test-1-gpu-amd` CI 测试 `test_diffusion_generation[lingbot_world_realtime_plastic_beach]` 持续崩溃。错误信息为 `shape '[4680, 5120]' is invalid for input of size 5120`, 原因是 native scale-shift kernel 的 4D 分支错误地将 per-frame shift `[B, F, 1, C]` 当作 per-token `[B, L, C]` 处理。

实现拆解

1. 定位问题: 在 `python/sglang/jit_kernel/diffusion/triton/scale_shift.py` 的 `fuse_scale_shift` 函数中, `if scale.dim() == 4` 分支原本直接假设 shift 为 `[B, L, C]` 并 `reshape(rows, C)`, 当 shift 为 `[B, F, 1, C]` 时尺寸不匹配。
2. 添加条件分支: 在 shift 重塑前, 判断 `shift.dim() == 4 and current_platform.is_hip()`。若为真 (即 ROCm 平台接收了 per-frame 4D shift), 则通过 `shift.expand(B, num_frames, frame_seqlen, C).reshape(rows, C).contiguous()` 将 shift 广播为 per-token 形状 `[B, L, C]` 再 `flatten`; 否则保持原有逻辑。
3. 平台隔离: 使用 `current_platform.is_hip()` 将修复限制在 ROCm 路径, 避免影响 CUDA (使用 CUTLASS 原生支持 4D shift) 或其他后端。
4. 验证: 在 AMD MI355X 上复现并验证修复, 修复后输出与 3D per-token 路径 bitwise 一致; CI 中 relevant shard 从 1 failed, 6 passed 转为 7 passed。

关键文件:

- `python/sglang/jit_kernel/diffusion/triton/scale_shift.py` (模块 JIT 内核; 类别 source; 类型 core-logic): 唯一变更文件, 修复 native Triton kernel 中 4D per-frame shift 的形状处理错误。

关键符号: `fuse_scale_shift`

关键源码片段

python/sglang/jit_kernel/diffusion/triton/scale_shift.py

唯一变更文件，修复 native Triton kernel 中 4D per-frame shift 的形状处理错误。

```
# 文件：python/sglang/jit_kernel/diffusion/triton/scale_shift.py
# 关键修改：在 4D scale/shift 分支中处理 per-frame shift

# Compact scale [B, F, 1, C] -> [B*F, C] (per-frame)
scale_resaped = scale.squeeze(2).reshape(-1, C).contiguous()

# 新增：针对 ROCm 平台的 per-frame shift 处理
# (CUDA 有 CUTLASS fused kernel 原生支持 [B, F, 1, C]，无需此修复)
if shift.dim() == 4 and current_platform.is_hip():
    # ROCm 上没有 fused CUTLASS scale-shift kernel，因此 native path
    # 必须处理 causal Wan / LingBot 输出 AdaLN 传入的 per-frame shift
    # [B, F, 1, C]。先 broadcast 到每帧所有 token，变为 [B, L, C]，
    # 再 flatten 为 [B*L, C]，以匹配 _fused_scale_shift_4d_kernel 的
    # per-token 索引
    shift_resaped = (
        shift.expand(B, num_frames, frame_seqlen, C)
        .reshape(rows, C)
        .contiguous()
    )
else:
    # shift 已经是 per-token [B, L, C] -> [B*L, C]
    shift_resaped = shift.reshape(rows, C).contiguous()
```

评论区精华

无 review 讨论。PR 获得 single approval。

- 暂无高价值评论线程

风险与影响

- 风险：低风险。变更仅影响 ROCm 平台下 `shift.dim() == 4` 的路径，且通过 `current_platform.is_hip()` 严格隔离。CUDA 路径及非 4D shift 路径完全不变。回归风险低。
- 影响：影响范围限定于 AMD ROCm 平台上的 causal diffusion 模型（LingBot-World, Wan），修复了 blocking CI 问题。对其他平台、模型及非 causal 扩散无影响。
- 风险标记：平台特定代码

关联脉络

- PR #24180 [JIT] Re-land native fuse_scale_shift_kernel: 引入了 native Triton `fuse_scale_shift_kernel` 及其 4D 分支，该分支假设 `shift` 为 per-token `[B, L, C]`，是本次 bug 的根因。

- PR #26954 [diffusion] model: support lingbot-world: 添加了 causal LingBot 模型, 其输出 AdaLN 传入 per-frame 4D shift [B, F, 1, C], 触发了本 bug。