PR #23671 完整报告

sgl-project/sglang

[AMD][bugfix] add gate rocm >= 7.2 for bpreshuffle

合并时间：2026-04-25 04:26

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23671

执行摘要

- 一句话：ROCm 7.0 编译 bpreshuffle 时回退到 Triton GEMM
- 推荐动作：该 PR 值得快速合并，它解决了一个关键精度回归问题，且设计清晰、风险可控。建议未来考虑测试环境覆盖 ROCm 7.0 场景，防止类似编译器回归。

功能与动机

gemm_a8w8_blockscale_bpreshuffle（#23319 引入）在 ROCm 7.0 hipcc 下被误编译，导致 GLM-5.1 FP8 精度在 MI35x 上显著下降（从 0.948 降至 0.944）。ROCm 7.2 无此问题，因此需要编译版本门控来保证精度。

实现拆解

1. 添加版本检测函数(python/sglang/srt/utils/common.py)：新增 get_hip_version() 函数，解析 torch.version.hip 字符串（如 "7.0.0-0423"）返回 (major, minor, patch) 元组，无 HIP 时返回 (0, 0, 0)。
2. 定义门控常量(python/sglang/srt/layers/quantization/fp8_utils.py)：导入 get_hip_version 并定义 _use_aiter_bpreshuffle_gfx95 = _use_aiter_gfx95 and get_hip_version() >= (7, 2, 0)。该变量控制是否启用 bpreshuffle 路径。
3. 修改条件判断(python/sglang/srt/layers/quantization/fp8_utils.py 和 python/sglang/srt/layers/quantization/fp8.py)：将原有 _use_aiter_gfx95 引用全部替换为 _use_aiter_bpreshuffle_gfx95，确保在 ROCm < 7.2 时 use_triton = True，从而回退到 Triton 实现。
4. 无测试或配置变更：此 PR 未添加新测试，仅通过源码级编译规避问题。

关键文件：

- python/sglang/srt/utils/common.py（模块 工具函数；类别 source；类型 core-logic；符号 get_hip_version）：新增 get_hip_version() 函数，为门控逻辑提供版本号检测能力。
- python/sglang/srt/layers/quantization/fp8_utils.py（模块 量化层；类别 source；类型 core-logic；符号 _use_aiter_bpreshuffle_gfx95, aiter_w8a8_block_fp8_linear）：引入 _use_aiter_bpreshuffle_gfx95 常量，控制 bpreshuffle 路径是否启用，并修改函数 aiter_w8a8_block_fp8_linear 的条件判断。
- python/sglang/srt/layers/quantization/fp8.py（模块 量化层；类别 source；类型 core-logic）：在 Fp8LinearMethod.process_weights_after_loading 中将引用更新为

_use_aiter_bpreshuffle_gfx95，确保权重预处理逻辑与门控一致。

关键符号：get_hip_version

关键源码片段

python/sglang/srt/utils/common.py

新增 get_hip_version() 函数，为门控逻辑提供版本号检测能力。

```
def get_hip_version():
    # 从 torch.version.hip 中提取版本号，格式如 "7.0.0-0423"
    if torch.version.hip:
        # 先按 "-" 分割去掉构建标识，再按 "." 分割为整数元组
        return tuple(map(int, torch.version.hip.split("-")[0].split(".")))
    # 非 HIP 环境返回 (0, 0, 0)，表示不可用
    return (0, 0, 0)
```

python/sglang/srt/layers/quantization/fp8_utils.py

引入 _use_aiter_bpreshuffle_gfx95 常量，控制 bpreshuffle 路径是否启用，并修改函数 aiter_w8a8_block_fp8_linear 的条件判断。

```
# 原有变量 _use_aiter_gfx95 仅检查硬件支持
_use_aiter_gfx95 = _use_aiter and _is_gfx95_supported

# 新增：在硬件支持基础上，进一步检查 ROCm 版本 >= 7.2
# ROCm 7.0 hipcc 会误编译 gemm_a8w8_blockscale_bpreshuffle 导致精度下降
_use_aiter_bpreshuffle_gfx95 = _use_aiter_gfx95 and get_hip_version() >= (7, 2, 0)

def aiter_w8a8_block_fp8_linear(input, weight, block_size, weight_scale, ...):
    ...
    # 原条件为 if _use_aiter_gfx95，现改为使用门控版本
    if _use_aiter_bpreshuffle_gfx95:
        # 仅在 ROCm >= 7.2 时尝试使用 bpreshuffle 优化
        use_triton = use_aiter_triton_gemm_w8a8_tuned_gfx950(n, k)
    else:
        # ROCm < 7.2 或非 bpreshuffle 场景，回退到 Triton 实现
        use_triton = True
```

评论区精华

无显著的 review 讨论（评论数为 0）。但有 1 条 Issue 评论来自 gemini-code-assist[bot] 提示配额已满，未实际参与讨论。

- 暂无高价值评论线程

风险与影响

- 风险：

    - 回归风险低：仅当 ROCm >= 7.2 且使用 gfx95 时启用 bpreshuffle，低版本自动回退，行为与 #23319 前一致。
    - 性能影响：ROCm < 7.2 的 gfx95 用户失去 bpreshuffle 加速，回退至 Triton GEMM，但确保了正确性。
    - 兼容性：未引入新依赖，get_hip_version 解析逻辑简单，无副作用。
- 影响：
    - 用户影响：影响使用 AMD MI35x (gfx95) 且 ROCm 版本 < 7.2 的用户，修复了 FP8 精度问题。
    - 系统影响：运行时无额外开销，仅模块导入时调用一次 get_hip_version。
    - 团队影响：小范围修复，无需文档更新或 CI 变更。
    - 风险标记：编译器回归，硬件特定路径

关联脉络

- PR #23319 [AMD] Add bpreshuffle FP8 GEMM for gfx95: 该 PR 引入了 gemm_a8w8_blockscale_bpreshuffle 并导致了 ROCm 7.0 下的精度问题，本 PR 是对其的补充修复。