PR #40857 完整报告

vllm-project/vllm

[CI][AMD][BugFix] Prevent triton compiler error when running test_moe_layer with use_ep = True on ROCm

合并时间：2026-05-14 16:44

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/40857

执行摘要

- 一句话：修复 ROCm 上 Triton MoE 因 scale 为 None 的编译错误
- 推荐动作：建议阅读此 PR，展示了一种在条件分支中 fallback 的安全修复方式，避免在调用链上游做更大改动。

功能与动机

当运行 test_moe_layer 时，频繁遇到 Triton 编译错误（ConstexprType 缺少 is_ptr），原因是 a1q_scale 为 None 被传入 Triton 内核。该问题出现在 MI300 上，因为需要 Triton kernels。

实现拆解

1. 根因分析：process_fp8_input_tensor_strategy_moe 返回的 scale 是 0 维 tensor，导致 naive_dp_ep.py 将 scales 设为 None，最终传入 Triton 内核。
2. 修改方案：在 TritonExperts.apply 中，将调用 invoke_fused_moe_triton_kernel 时的 a1q_scale 参数替换为 a1q_scale if a1q_scale is not None else self.a1_scale，实现类似 FlashInferExperts 的 fallback 逻辑。
3. 测试验证：运行 pytest -sv tests/kernels/moe/test_moe_layer.py，所有子测试均通过。

关键文件：

- vllm/model_executor/layers/fused_moe/experts/triton_moe.py（模块 MoE 层；类别 source；类型 data-contract；符号 TritonExperts.apply）：唯一修改的文件，修复 Triton 编译器错误的根源，在 scale 为 None 时 fallback 到 self.a1_scale。

关键符号：TritonExperts.apply

关键源码片段

vllm/model_executor/layers/fused_moe/experts/triton_moe.py

唯一修改的文件，修复 Triton 编译器错误的根源，在 scale 为 None 时 fallback 到 self.a1_scale。

```
# 向 Triton 内核传递 activation scale
# 如果外部传入的 a1q_scale 为 None（例如在 dp_ep 路径中），则使用
```

```
# 实例变量 self.a1_scale 作为 fallback，避免 None 导致 Triton 编译失败
invoke_fused_moe_triton_kernel(
    hidden_states,
    w1,
    intermediate_cache1,
    a1q_scale if a1q_scale is not None else self.a1_scale,
    self.w1_scale,
    None, # topk_weights
    sorted_token_ids,
    expert_ids,
    num_tokens_post_padded,
    False, # mul_routed_weights
    top_k_num,
    config,
    compute_type=compute_type,
    use_fp8_w8a8=self.quant_config.use_fp8_w8a8,
    use_int8_w8a8=self.quant_config.use_int8_w8a8,
    use_int8_w8a16=self.quant_config.use_int8_w8a16,
    use_int4_w4a16=self.quant_config.use_int4_w4a16,
    per_channel_quant=self.per_act_token_quant,
    block_shape=self.block_shape,
    B_bias=self.w1_bias,
)
```

评论区精华

- gemini-code-assist建议在 fp8_utils.py 中使用 .view(1) 代替 torch.tensor([...]) 以避免 host-device 同步（该建议最终未纳入合并版本）。
- rasmith回复使用 unsqueeze 实现，但最终合并版本只修改了 triton_moe.py。
- yewentao256审批通过（LGTM）。
- 使用 .view(1) 优化 scale 维度转换 (performance): 作者回复使用 unsqueeze 实现，但最终合并版本未包含该文件修改

风险与影响

- 风险：风险极低：仅修改一行，在 a1q_scale 为 None 时 fallback 到 self.a1_scale，这假设 self.a1_scale 存在且含义一致。已在 CUDA 和 ROCm 上验证，未发现回归。但需注意此修改依赖模型配置中 self.a1_scale 的正确初始化，若未初始化可能引入新问题。
- 影响：影响范围小：仅影响 ROCm 上使用 Triton 内核的 MoE 推理，修复了 test_moe_layer 的编译错误，对已有功能无负面影响。
- 风险标记：低风险，单行修复

关联脉络

- 暂无明显关联 PR