

PR #49913 完整报告

vllm-project/vllm

[ROCm] Make vllm_c RMSNorm output contiguous

合并时间: 2026-07-27 12:59

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/49913>

执行摘要

- 一句话: 修复 ROCm vllm_c RMSNorm 输出非连续问题
- 推荐动作: 本 PR 为定位精准的小范围 bugfix, 变更清晰, 值得阅读
vllm/kernels/vllm_c.py 中的输出创建改进和 test_layernorm.py 中的转置输入测试用例。

功能与动机

ROCm 上的 vllm_c RMSNorm 内核 (libtorch-stable) 要求输出张量连续, 但原代码使用 `torch.empty_like(x)` 创建输出, 当输入为 transposed view 时会保留非连续步幅, 导致内核 crash。PR body 中引用了 CI 失败日志 (<https://buildkite.com/vllm/amd-ci/builds/11266>) 作为复现证据。

实现拆解

1. 修改输出创建方式 (vllm/kernels/vllm_c.py) : 在 ROCm 分支中将 `output = torch.empty_like(x)` 改为 `torch.empty(x.shape, device=x.device, dtype=x.dtype)`, 确保输出张量总是连续的, 满足 libtorch-stable 内核要求。
2. 新增转置输入回归测试 (tests/kernels/ir/test_layernorm.py) : 新增 `test_vllm_c_rms_norm_accepts_transposed_input` 测试函数, 构造 shape (1, 320, 120) 的转置张量并验证输出与 native 实现一致。

关键文件:

- vllm/kernels/vllm_c.py (模块 内核层; 类别 source; 类型 core-logic; 符号 rms_norm) :
修复核心: 将 `torch.empty_like` 改为 `torch.empty` 以确保 ROCm 分支输出连续。
- tests/kernels/ir/test_layernorm.py (模块 层归一化; 类别 test; 类型 test-coverage; 符号 test_vllm_c_rms_norm_accepts_transposed_input) : 新增回归测试, 覆盖转置输入场景。

关键符号: rms_norm

关键源码片段

vllm/kernels/vllm_c.py

修复核心: 将 `torch.empty_like` 改为 `torch.empty` 以确保 ROCm 分支输出连续。

vllm/kernels/vllm_c.py (关键修改)

```

@ir.ops.rms_norm.register_impl(
    "vllm_c", supports_args=rms_no_var_size, supported=CUDA_ALIKE
)
def rms_norm(
    x: Tensor, weight: Tensor | None, epsilon: float, variance_size: int | None = None
) -> Tensor:
    assert variance_size is None
    # ROCm's vLLM C RMSNorm kernel operates on contiguous 2D tensors.
    # Higher-rank callers still normalize over the last dimension, so flatten
    # all leading dims. reshape handles strided views from q/k/v splits.
    if IS_ROCM and (x.dim() > 2 or not x.is_contiguous()):
        original_shape = x.shape
        x = x.reshape(-1, original_shape[-1])
        # 修复前: output = torch.empty_like(x) 会保留 transposed input 的 strides
        # 修复后: 使用 torch.empty 显式指定 shape, 确保 output 连续
        output = torch.empty(x.shape, device=x.device, dtype=x.dtype)
        torch.ops._C.rms_norm(output, x, weight, epsilon)
        return output.reshape(original_shape)

    output = torch.empty(x.shape, device=x.device, dtype=x.dtype)
    torch.ops._C.rms_norm(output, x, weight, epsilon)
    return output

```

tests/kernels/ir/test_layernorm.py

新增回归测试，覆盖转置输入场景。

```

# tests/kernels/ir/test_layernorm.py ( 新增测试 )

@pytest.mark.skipif(
    not current_platform.is_rocm(),
    reason="ROCm vllm_c RMSNorm needs a contiguous output for strided inputs",
)
def test_vllm_c_rms_norm_accepts_transposed_input():
    impl = ir.ops.rms_norm.impls["vllm_c"]
    if not impl.supported:
        pytest.skip("vllm_c impl not supported on this platform")

    # 构造一个类似 VibeVoice 的转置输入 (1, 320, 120) -> transpose(1,2) -> (1, 120, 320)
    x = torch.randn(
        1, 320, 120, dtype=torch.float16, device=current_platform.device_type
    ).transpose(1, 2)
    # 确认最后维度的 stride 不为 1 (非连续)
    assert x.reshape(-1, x.shape[-1]).stride(-1) != 1
    weight = torch.randn(320, dtype=torch.float16, device=current_platform.device_type)
    epsilon = 1e-5

    output = impl.impl_fn(x, weight, epsilon)
    ref_output = rms_norm_native(x, weight, epsilon)

    assert output.shape == x.shape

```

```
assert_close(ir.ops.rms_norm, output, ref_output)
```

评论区精华

无 review 讨论。

- 暂无高价值评论线程

风险与影响

- 风险：变更范围极小（仅修改一行代码），且仅在 ROCm 平台的非连续输入路径生效，CUDA 等其他平台不受影响。新增测试覆盖了关键转置布局，回归风险低。
- 影响：影响范围仅限于 ROCm 平台上使用 vllm_c RMSNorm 且输入为转置张量的场景（如 VibeVoice 单 batch 布局）。修复后这些场景不再 crash，数值结果与 native 实现一致。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR