PR #50905 完整报告

vllm-project/vllm

[ROCm][CI] Add aiter per-token FP8 quant roundtrip and RMSNorm determinism tests

合并时间：2026-08-05 10:32

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/50905

执行摘要

- 一句话：ROCm 新增 AITER per-token FP8 量化与 RMSNorm 确定性测试
- 推荐动作：作为纯测试补强 PR，价值主要在于防止 ROCm AITER 量化和归一化内核后续优化引入精度退化或非确定性。建议浏览测试断言方式（roundtrip 反量化对比、torch.equal 位级比较），不必精读；若后续要在 CI 中扩大覆盖，可考虑增加多 shape 参数化和多 GPU 确定性验证。对于维护者，可以关注宽松阈值是否足以捕获实际退化。

功能与动机

PR body 说明：Add test_per_token_quant_matches_native: verifies per-token FP8 quantization roundtrip accuracy (per-tensor version existed, per-token did not)；Add test_rms_norm_determinism: verifies AITER RMSNorm produces bitwise-identical results across repeated calls。即补齐 per-token 量化测试缺口，并验证 RMSNorm 的重复调用确定性，防止后续内核重构引入非确定性或精度退化。

实现拆解

1. 在 tests/rocm/aiter/test_quant_op_schema.py 中新增 test_per_token_quant_matches_native：调用 rocm_aiter_ops.per_token_quant 对随机输入做 per-token FP8 量化，断言输出 shape、dtype 与 scale 维度，再通过 scale.view(-1, 1) 反量化回 float32 后与输入对比（rtol=0.07, atol=5e-2）。
2. 新增 test_rms_norm_determinism：先 import vllm.kernels.aiter_ops 确保算子注册，再对 bfloat16 输入连续调用 torch.ops.vllm_aiter.rms_norm 三次，用 torch.equal 断言每次结果与首次调用完全一致（bitwise）。
3. 删除原有一段关于已移除 test_per_tensor_quant_torch_compile 的历史注释，该测试因 fp8-safe opcheck 支持而冗余，避免误导后续维护者。
4. 无配置、schema、部署或生产代码配套改动，纯测试文件变更。

关键文件：

- tests/rocm/aiter/test_quant_op_schema.py（模块 量化算子；类别 test；类型 test-coverage；符号 test_per_token_quant_matches_native, test_rms_norm_determinism）：本 PR 唯一变更文件，新增 per-token FP8 量化 roundtrip 精度测试和 RMSNorm 确定性测试，补齐 ROCm AITER 量化算子测试缺口。

关键符号：test_per_token_quant_matches_native, test_rms_norm_determinism

关键源码片段

tests/rocm/aiter/test_quant_op_schema.py

本 PR 唯一变更文件，新增 per-token FP8 量化 roundtrip 精度测试和 RMSNorm 确定性测试，补齐 ROCm AITER 量化算子测试缺口。

```
# tests/rocm/aiter/test_quant_op_schema.py（本次新增的两个测试）

def test_per_token_quant_matches_native():
    '''Per-token quant output dequantizes back to the input within FP8 error.'''
    torch.manual_seed(0)
    x = _x()

    # 调用 ROCm AITER 的 per-token 量化：返回 FP8 输出与每个 token 对应的 scale
    out, scale = rocm_aiter_ops.per_token_quant(x, FP8_DTYPE)

    # 基本契约：输出 shape/dtype 与输入一致，scale 按第一维（token 维）对齐
    assert out.shape == x.shape
    assert out.dtype == FP8_DTYPE
    assert scale.shape[0] == x.shape[0]

    # roundtrip 校验：反量化后与原始输入对比，允许 FP8 舍入误差（rtol=0.07, atol=5e-2）
    deq = out.to(torch.float32) * scale.view(-1, 1)
    torch.testing.assert_close(deq, x.to(torch.float32), rtol=0.07, atol=5e-2)


def test_rms_norm_determinism():
    '''AITER RMSNorm produces bitwise-identical results across repeated calls.'''
    # 显式导入确保 vllm_aiter 算子完成注册
    import vllm.kernels.aiter_ops # noqa: F401

    torch.manual_seed(0)
    M, N = 32, 512
    x = torch.randn(M, N, dtype=torch.bfloat16, device='cuda')
    weight = torch.ones(N, dtype=torch.bfloat16, device='cuda')
    eps = 1e-5

    # 以第一次调用结果为位级基准，后续每次调用都必须 bitwise 一致（torch.equal）
    reference = torch.ops.vllm_aiter.rms_norm(x, weight, eps)
    for i in range(3):
        result = torch.ops.vllm_aiter.rms_norm(x, weight, eps)
        assert torch.equal(reference, result), f'Run {i + 1} differs from reference'
```

评论区精华

无实质技术讨论。claude[bot] 提示该 PR 来自 fork，自动审查被禁用，维护者可评论 @claude review 触发一次性审查；AndreasKaratzas 直接以 LGTM 批准。没有发现设计权衡或未解决疑虑。

- fork PR 自动审查策略 (other): 维护者未触发额外审查，AndreasKaratzas 直接以 LGTM 批准合并；无实质技术讨论。

风险与影响

- 风险：仅改测试文件，生产代码零改动，回归风险非常低。主要风险点： 1) 测试依赖 ROCm AITER 环境，在非 ROCm 或未安装 AITER 的 CI 上需要正确跳过； 2) per-token 量化测试的 FP8 舍入阈值（rtol=0.07, atol=5e-2）相对宽松，轻度精度退化可能漏检； 3) RMSNorm 确定性测试只覆盖单一 shape（32x512）和单位 weight，未覆盖多 GPU 或其他 shape，确定性结论的泛化性有限。
  - 影响：影响范围：仅 ROCm CI 测试套件中的 tests/rocm/aiter/test_quant_op_schema.py。为运行在 ROCm 上的 AITER per-token 量化与 RMSNorm 提供回归保护，防止后续内核优化引入精度退化或非确定性；对最终用户无直接可见影响；对团队是低成本高收益的测试覆盖补强。
  - 风险标记：纯测试变更，依赖 ROCm 环境，FP8 阈值宽松

关联脉络

- 暂无明显关联 PR