

PR #46414 完整报告

vllm-project/vllm

[ROCm] Fix AITER FP8 quantization schema tests

合并时间: 2026-06-24 22:29

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/46414>

执行摘要

- 一句话: 修复 AITER FP8 量化 schema 测试失败
- 推荐动作: 建议合并。该 PR 解决了一个实际的 CI 问题, 且修改审慎, 通过使用共享工具和调整测试策略提高了测试的健壮性和可维护性。

功能与动机

AITER FP8 量化 schema 测试在 MI300 上持续失败, 影响 ROCm 平台的 CI 稳定性。具体表现为: `test_per_tensor_quant_torch_compile` 因 PyTorch 版本差异 (`torch >= 2.12` 移除 `error_on_custom_op_aliasing`) 报错; `test_per_tensor_quant_matches_native[dynamic]` 因 AITER 与原生 FP8 动态 scale 的 max 值不同 (240 vs 224) 导致数值比较失败。需要修复这些测试以恢复对 AITER FP8 量化的验证。

实现拆解

1. 使用共享 `opcheck` 辅助函数: 将 `tests/rocm/aiter/test_quant_op_schema.py` 中所有直接调用 `torch.library.opcheck` 的地方替换为从 `tests/kernels/utils.py` 导入的共享 `opcheck` 函数。该函数内部通过 `torch.allclose` 的 `atol/rtol` 容忍 FP8 精度, 从而让 `test_schema` 能够正确运行在 FP8 操作上 (否则会因 `"mul_cuda" is unimplemented for fp8` 而失败)。同时移除了不再需要的 `_INPLACE_OPCHECK_UTILS` 配置。
2. 移除 `test_per_tensor_quant_torch_compile` 测试: 该测试原本用于覆盖 in-place 操作的 aliasing 契约, 但依赖 `error_on_custom_op_aliasing` 配置 (仅在 `torch < 2.12` 存在)。现在 `test_schema` 已能直接通过共享 `opcheck` 检查 aliasing 契约, 因此该测试被删除。
3. 修改动态数值测试逻辑: `test_per_tensor_quant_matches_native[dynamic]` 不再直接比较 AITER 输出与原生 `scaled_fp8_quant` 的输出 scale (因网格不同), 改为验证 AITER 自己的 `(out, scale)` 能够反量化回原始输入。静态测试保持不变 (scale 相同, 必须匹配原生)。

关键文件:

- `tests/rocm/aiter/test_quant_op_schema.py` (模块 量化; 类别 test; 类型 test-coverage; 符号 `test_per_tensor_quant_torch_compile`, `fn`): 唯一变更文件, 集中了所有修复: 导入共享 `opcheck`、移除不可用的 `torch_compile` 测试、修改动态数值测试为自洽验证。

关键符号: `test_per_tensor_quant_static_schema`,

`test_per_tensor_quant_dynamic_schema`, `test_per_token_quant_dynamic_schema`,

test_group_fp8_quant_schema, test_per_tensor_quant_matches_native

关键源码片段

tests/rocm/aiter/test_quant_op_schema.py

唯一变更文件，集中了所有修复：导入共享 opcheck、移除不可用的 torch_compile 测试、修改动态数值测试为自洽验证。

```
# tests/rocm/aiter/test_quant_op_schema.py
# 变更后使用共享 opcheck 辅助函数，该函数内部通过 atol/rtol 容忍 FP8 精度
import importlib.util
import pytest
import torch

# 导入共享的 opcheck，它已处理 FP8 精度问题
from tests.kernels.utils import opcheck
from vllm._aiter_ops import rocm_aiter_ops
from vllm.platforms import current_platform

aiter_available = importlib.util.find_spec("aiter") is not None

@pytest.mark.skipif(
    not (current_platform.is_rocm() and aiter_available),
    reason="AITER ops are only available on ROCm with aiter package installed",
)
FP8_DTYPE = current_platform.fp8_dtype()

def _x(M=128, N=4096):
    return torch.randn((M, N), dtype=torch.float16, device="cuda")

def test_per_tensor_quant_static_schema():
    x = _x()
    out = torch.empty_like(x, dtype=FP8_DTYPE)
    scale = torch.ones(1, dtype=torch.float32, device="cuda")
    # 使用共享 opcheck，不再需要 test_utils 参数
    opcheck(torch.ops.vllm.rocm_aiter_per_tensor_quant, (out, x, scale, False))

def test_per_tensor_quant_dynamic_schema():
    x = _x()
    out = torch.empty_like(x, dtype=FP8_DTYPE)
    scale = torch.empty(1, dtype=torch.float32, device="cuda")
    opcheck(torch.ops.vllm.rocm_aiter_per_tensor_quant, (out, x, scale, True))

def test_per_token_quant_dynamic_schema():
    x = _x()
    opcheck(torch.ops.vllm.rocm_aiter_per_token_quant, (x, FP8_DTYPE, None))

def test_group_fp8_quant_schema():
    x = _x()
```

```
opcheck(torch.ops.vllm.rocm_aiter_group_fp8_quant, (x, 128))
```

```
@pytest.mark.parametrize("dynamic", [True, False])
def test_per_tensor_quant_matches_native(dynamic):
    from vllm import _custom_ops as ops
    torch.manual_seed(0)
    x = _x()
    scale_in = None if dynamic else torch.tensor([0.5], dtype=torch.float32, device="cuda")
    out, scale = rocm_aiter_ops.per_tensor_quant(x, FP8_DTYPE, scale_in)
    ref_out, ref_scale = ops.scaled_fp8_quant(x, scale_in)
    assert out.shape == x.shape
    assert out.dtype == FP8_DTYPE
    # 静态测试比较 scale 和反量化结果；动态测试仅验证反量化一致性
    # 详见 patch 中的额外断言（此处省略以保持简洁）
```

评论区精华

无 reviewer 评论。

- 暂无高价值评论线程

风险与影响

- 风险：低风险。变更仅涉及测试文件，不修改生产代码。共享 opcheck 函数已在其他测试中使用，行为稳定。移除 test_per_tensor_quant_torch_compile 不会损失测试覆盖，因为 aliasing 契约已由 test_schema 覆盖。动态数值测试改为自洽验证，虽放宽了与原生实现的直接比较，但保留了核心正确性检查。
- 影响：正面影响：修复了 ROCm 平台上 AITER FP8 量化 schema 测试的失败，恢复 CI 验证能力。影响范围仅限于测试文件，无用户可见行为变化。对于维护者，减少了 ROCm 测试的噪音。
- 风险标记：测试专用变更

关联脉络

- 暂无明显关联 PR