PR #40553 完整报告

vllm-project/vllm

test: add nan/inf clamp regression test for fused_topk_bias

合并时间：2026-04-22 08:46

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/40553

执行摘要

- 一句话：为 fused_topk_bias 添加 NaN/Inf 钳制回归测试，确保专家 ID 唯一性。
- 推荐动作：该 PR 值得快速浏览以了解测试模式和内核钳制的验证方式，但核心设计决策已在 PR #39391 中讨论。关注点在于测试参数化和对 fused_topk_bias 路径的覆盖。

功能与动机

动机源于 Issue #40457，该 issue 要求将 fused_topk_bias 纳入 CI 测试覆盖，作为 PR #39391 修复的后续跟进。PR body 明确表示 'Closes #40457'，并指出 fused_topk_bias 路由通过相同内核（topk_softmax_kernels.cu），钳制已生效，但需要测试验证以确保覆盖。

实现拆解

1. 识别变更文件：仅修改 tests/kernels/moe/test_fused_topk.py，无源码主路径改动。
2. 添加测试函数：新增 test_fused_topk_bias_nan_inf_clamp 函数，结构与现有 test_fused_topk_nan_inf_clamp 类似，但调用 fused_topk_bias 接口。
3. 参数化覆盖：通过 pytest 参数化覆盖 num_experts (6,8,16)、topk (3,4)、scoring_func (softmax,sigmoid)、bad_value (NaN,Inf)、dtype (bfloat16, half, float32)，共 144 个测试用例。
4. 测试逻辑：生成包含 NaN/Inf 的污染行，验证输出专家 ID 唯一性和权重有限性，同时确保正常行结果与参考一致。
5. 测试配套：无需修改 CI 配置，测试自动被现有 Kernels MoE Test 步骤收集。

关键文件：

- tests/kernels/moe/test_fused_topk.py（模块 MoE 融合 TopK；类别 test；类型 test-coverage；符号 test_fused_topk_bias_nan_inf_clamp）：新增 fused_topk_bias 的 NaN/Inf 钳制回归测试，确保内核修复覆盖偏差路径。

关键符号：test_fused_topk_bias_nan_inf_clamp

关键源码片段

tests/kernels/moe/test_fused_topk.py

新增 fused_topk_bias 的 NaN/Inf 钳制回归测试，确保内核修复覆盖偏差路径。

```
@pytest.mark.skipif(
    not current_platform.is_cuda(), reason="This test is skipped on non-CUDA platform."
)
@pytest.mark.parametrize("num_experts", [6, 8, 16])
@pytest.mark.parametrize("topk", [3, 4])
@pytest.mark.parametrize("scoring_func", ["softmax", "sigmoid"])
@pytest.mark.parametrize("bad_value", [float("nan"), float("inf")])
@pytest.mark.parametrize("dtype", [torch.bfloat16, torch.half, torch.float32])
def test_fused_topk_bias_nan_inf_clamp(
    num_experts: int,
    topk: int,
    scoring_func: str,
    bad_value: float,
    dtype: torch.dtype,
):
    """回归测试：当存在 e_score_correction_bias 时，门控 logits 中的 NaN/Inf 不得产生重复的专家
    ID 或非有限权重。

    与 test_fused_topk_nan_inf_clamp 相同场景，但练习偏差路径 (fused_topk_bias)，以便 topk_
    softmax_kernels.cu 中的修复也覆盖该入口点。
    """
    torch.manual_seed(0) # 固定随机种子确保测试可重复
    num_tokens = 4
    hidden_size = 1024
    hidden_states = torch.randn((num_tokens, hidden_size), dtype=dtype, device="cuda")
    e_score_correction_bias = torch.randn(
        (num_experts,), dtype=torch.float32, device="cuda"
    ) # 专家分数校正偏差，用于 DeepSeek 风格模型

    gating_output = torch.randn((num_tokens, num_experts), dtype=dtype, device="cuda")
    gating_output[1:, :] = bad_value # 污染除第一行外的所有行，模拟 NaN/Inf 输入

    topk_weights, topk_ids = fused_topk_bias(
        hidden_states=hidden_states,
        gating_output=gating_output,
        e_score_correction_bias=e_score_correction_bias,
        topk=topk,
        renormalize=False,
        scoring_func=scoring_func, # 软最大或 sigmoid 评分函数
    )

    # 正常行仍必须匹配 torch 参考，确保基础功能不受影响
    ref_weights, ref_ids = torch_topk(
        gating_output=gating_output[:1],
        topk=topk,
        renormalize=False,
        e_score_correction_bias=e_score_correction_bias,
        scoring_func=scoring_func,
    )
```

```
    torch.testing.assert_close(
        ref_weights.to(torch.float32), topk_weights[:1], atol=1e-2, rtol=1e-2
    )
    torch.testing.assert_close(ref_ids.to(torch.int32), topk_ids[:1], atol=0, rtol=0)

    # 污染行：ID 必须唯一（无重复），权重必须有限（防止 NaN/Inf 传播到下游 MoE 内核）
    for row in range(1, num_tokens):
        row_ids = topk_ids[row]
        assert row_ids.unique().numel() == topk, (
            f"Row {row} has duplicate expert IDs {row_ids.tolist()} "
            f"(bad_value={bad_value}, scoring_func={scoring_func})"
        )
        assert torch.isfinite(topk_weights[row]).all(), (
            f"Row {row} has non-finite weights {topk_weights[row].tolist()} "
            f"(bad_value={bad_value}, scoring_func={scoring_func})"
        )
```

评论区精华

review 中，vadiklyutiy 提问 'I think we also should add this test to CI. I don't see the tests name or corresponding dir in .buildkite'，作者 jhaotingc 回应已验证测试自动收集，并引用 .buildkite 文件。vadiklyutiy 确认后批准。讨论焦点是 CI 集成，结论是无需额外修改。

- CI 测试覆盖确认 (testing): 确认无需额外 CI 修改，测试已覆盖。

风险与影响

- 风险：风险极低，仅添加测试不修改生产代码。潜在风险是测试假阳性或覆盖不足，但参数化全面，且依赖已有内核修复，实际风险可忽略。
- 影响：对用户无直接影响，增强系统测试覆盖，提升对 MoE 内核稳定性的信心。对团队，提供回归保护，防止未来代码变更破坏钳制逻辑。
- 风险标记：无源码变更风险，测试覆盖全面

关联脉络

- PR #39391 fix: clamp NaN/Inf in topk_softmax to prevent duplicate expert IDs: 本 PR 是该修复的测试跟进，专门验证 fused_topk_bias 路径。