

PR #55453 完整报告

vllm-project/vllm

[CI] Restore clean GPU state before OAI Triton MoE tests

合并时间: 2026-09-08 16:15

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/55453>

执行摘要

- 一句话: 将 `test_modular_oai_triton_moe.py` 加入 GPU 内存清理白名单, 修复 H100 CI 测试隔离失败。
- 推荐动作: 这是一个简单、明确且必要的 CI 修复。由于其直接解决了影响多天的 CI 失败问题, 且变更微小、风险极低, 建议快速合入。该 PR 适合作为代码规范或测试最佳实践 (如何维护测试清理白名单) 的参考案例。

功能与动机

H100 FP8 MoE 测试车道在 13 次主分支告警中失败。具体表现为

`test_modular_oai_triton_moe.py` 的 14 个测试用例在等待 GPU 使用率低于 10% 时失败, 而内存使用率稳定在 14.40 GiB (79.65 GiB GPU)。PR#54954 在为 MoE 测试添加清理白名单时, 遗漏了此文件。本 PR 旨在恢复该文件的清理入口, 修复此 CI 回归。

实现拆解

1. 变更入口: 修改 `tests/kernels/moe/conftest.py` 文件。
2. 核心逻辑: 在 `NEEDS_CLEAN_ENTRY` frozenset 中添加字符串 "`test_modular_oai_triton_moe.py`"。此集合是 `pytest conftest` 中的一个配置, 用于标识哪些测试文件在开始执行前需要进行全局 GPU 内存和分布式环境的清理。
3. 变更原因: PR#54954 引入了清理白名单机制, 但遗漏了 `test_modular_oai_triton_moe.py`。该文件中的测试函数显式调用了 `wait_for_gpu_memory_to_clear`, 其正确性依赖于从干净的 GPU 状态开始。将其加入白名单, 可以恢复该文件测试前的清理行为。
4. 影响: 此变更仅影响 CI 测试的执行流程, 不涉及任何生产代码逻辑。它确保了该测试文件在 CI 环境中的稳定性和正确性。

关键文件:

- `tests/kernels/moe/conftest.py` (模块 测试配置; 类别 `test`; 类型 `test-coverage`; 符号 `NEEDS_CLEAN_ENTRY`): 这是本次变更的唯一文件。其中定义的 `NEEDS_CLEAN_ENTRY` 集合控制着 MoE 测试目录下各测试文件的 GPU 内存清理行为。将 `test_modular_oai_triton_moe.py` 加入此集合是修复 CI 回归的核心动作。

关键符号: 未识别

关键源码片段

tests/kernels/moe/conftest.py

这是本次变更的唯一文件。其中定义的 `NEEDS_CLEAN_ENTRY` 集合控制着 MoE 测试目录下各测试文件的 GPU 内存清理行为。将 `test_modular_oai_triton_moe.py` 加入此集合是修复 CI 回归的核心动作。

```
# 测试配置文件，控制哪些 MoE 测试文件在开始前需要执行全局 GPU 内存清理。
# PR#54954 引入此白名单，但遗漏了 test_modular_oai_triton_moe.py，
# 导致其在 H100 CI 上因内存未释放而失败。
NEEDS_CLEAN_ENTRY = frozenset(
    {
        # Fail unless they start from a clean distributed and memory state,
        # which they used to get only as a side effect of the preceding test
        # cleaning up after itself.
        "test_moe_layer.py",
        # 本次修复：将遗漏的文件加入白名单，恢复其测试前的清理行为。
        "test_modular_oai_triton_moe.py",
        "test_ocp_mx_moe.py",
        "test_zero_expert_moe.py",
        # Execute no test on ROCm, where this was measured, so nothing is known
        # about how they behave without the per-test cleanup. Cleaned on entry
        # as a precaution; drop once they have been seen green without it.
        "test_batched_deepgemm.py",
        "test_deepep_deepgemm_moe.py",
        "test_deepep_moe.py",
        "test_deepep_v2_moe.py",
        "test_deepgemm.py",
        "test_grouped_topk.py",
        "test_mxfp4_moe.py",
        "test_shared_fused_moe_routed_transform.py",
        "test_silu_mul_per_token_group_quant_fp8_colmajor.py",
        "test_situ_mul_fp8_quant.py",
    }
)
```

评论区精华

PR body 详细说明了问题背景（H100 FP8 MoE 车道失败）、根本原因（PR#54954 遗漏）和验证方法（本地 H200 测试复现与修复）。评论区主要为 CI 触发和自动化机器人消息，未出现关于设计或实现的争议性讨论。

- CI 测试失败修复与验证 (testing): 通过将目标文件加入 `NEEDS_CLEAN_ENTRY` 白名单恢复清理，本地验证成功，等待 H100 CI 车道确认。

风险与影响

- 风险：风险极低。变更仅在测试配置文件中添加一个字符串到白名单集合。不改变任何测试逻辑、断言或生产代码。最坏情况是 CI 清理流程运行不必要的清理（但这是预期行为），不会导致功能回归或性能下降。
- 影响：用户影响：无直接影响。这是纯 CI 修复，确保 MoE 测试在 H100 GPU 上稳定运行，从而间接保障了 FP8 MoE 内核的质量。系统影响：修复了 CI 流水线中的测试隔离问题，提升了构建可靠性和开发体验。团队影响：解决了持续 13 次的 CI 失败告警，减少了团队排查 CI 问题的负担。
- 风险标记：CI 测试隔离

关联脉络

- PR #54954 [CI] Per-test cleanup opt-out for MoE tests: 本 PR 是直接修复 PR#54954 引入的遗漏。PR#54954 为 MoE 测试目录添加了清理白名单机制，但未将 `test_modular_oai_triton_moe.py` 纳入其中，导致了本次需要修复的 CI 回归。
- PR #54202 [MoE] Re-enable TRITON_UNFUSED in the gpt-oss MXFP4 dispatch list: PR#54202 的测试计划中包含了 `pytest tests/kernels/moe/test_modular_oai_triton_moe.py`，表明其功能测试也依赖于此测试文件在干净的 GPU 环境下稳定运行。本 PR 的修复间接保障了该 PR 功能验证的可靠性。