

PR #37338 完整报告

sgl-project/sglang

[Fix][CPU] fix xeon ci failure by test_qwen35_flashinfer_fusion

合并时间: 2026-09-01 14:17

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/37338>

执行摘要

- 一句话: 修复 Qwen3.5 MoE 单元测试因使用错误模拟方式导致的 Xeon CI 失败。
- 推荐动作: 这是一个清晰、小规模且必要的测试修复 PR。它正确地解决了由前序 PR 引入的测试回归问题, 且不涉及生产代码。对于维护 CI 健康和测试准确性很重要, 但代码逻辑简单, 值得快速审阅以确认修复的正确性, 但不需要进行深入的代码逻辑分析。PR body 已充分解释了动机和修改, 是一个良好的实践。

功能与动机

这是一个针对 PR#32733 引入的测试 fixture 回归问题的修复。原测试在模拟 `server_args` 时, 为 `cutedsl_moe_max_num_tokens` 提供了一个 `lambda` 属性, 这与实际代码 (函数调用) 的行为不一致, 导致 CI 失败。PR body 明确指出: 'This is a test-fixture regression, not a production-code bug in #32733'。

实现拆解

1. 修改测试模拟方式: 在 `test_qwen35_flashinfer_fusion.py` 文件中, 对 `test_framework_capacity_is_maximum_of_all_sources` 函数进行修改。引入 `unittest.mock.patch` 装饰器, 在测试运行时将目标模块中的 `cutedsl_moe_max_num_tokens` 函数替换为返回固定值 8192 的 `mock` 对象, 以更准确地模拟生产环境中的函数调用行为。
2. 简化测试配置构造: 在移除模拟 `lambda` 后, 相应地简化了 `server_args` 的 `SimpleNamespace` 构造, 去除了对 `cutedsl_moe_max_num_tokens` 属性的直接设置, 使测试 fixture 更贴合实际的 `server_args` 结构。
3. 调整测试注册套件: 将文件的 CI 注册从 `base-c-test-cpu` 套件更改为 `base-a-test-cpu` 套件。根据 PR body 解释, 因为这是一个纯逻辑单元测试, 不依赖 Xeon/AMX 硬件特性, 因此应属于更基础的测试阶段。
4. 确认无生产代码变更: 本次 PR 所有变更均在测试文件内, `sglang.srt.layers.moe.qwen35_flashinfer_fusion` 等生产模块未做任何改动。

关键文件:

- `test/registered/unit/layers/moe/test_qwen35_flashinfer_fusion.py` (模块 MoE 测试; 类别 test; 类型 test-coverage; 符号 `test_framework_capacity_is_maximum_of_all_sources`): 这是本次修复的唯一文件, 包含导致 CI 失败的测试用例及其修复。

关键符号: test_framework_capacity_is_maximum_of_all_sources

关键源码片段

test/registered/unit/layers/moe/test_qwen35_flashinfer_fusion.py

这是本次修复的唯一文件，包含导致 CI 失败的测试用例及其修复。

```
# 修改前的测试函数定义和模拟方式
# def test_framework_capacity_is_maximum_of_all_sources():
#     graph = SimpleNamespace(...)
#     server_args = SimpleNamespace(
#         cuda_graph_config=graph,
#         ## 问题所在: 直接设置了一个 lambda 属性, 而非模拟函数调用
#         cutedsl_moe_max_num_tokens=lambda: 8192,
#     )
# ...

# 修改后的测试函数定义和模拟方式
@patch(
    "sglang.srt.layers.moe.qwen35_flashinfer_fusion.cutedsl_moe_max_num_tokens", # 正确
    patch 生产代码中的函数
    return_value=8192, # 模拟函数被调用时返回固定值 8192
def test_framework_capacity_is_maximum_of_all_sources(_cutedsl_moe_max_num_tokens): #
    mock 对象作为参数注入
    graph = SimpleNamespace(
        decode=SimpleNamespace(max_bs=512, bs=[1, 64, 256]),
        prefill=SimpleNamespace(max_bs=4096, bs=[1024, 2048, 4096]),
    )
    server_args = SimpleNamespace(cuda_graph_config=graph) # 简化 server_args
    构造, 移除了错误的 lambda 属性
    runner = SimpleNamespace(server_args=server_args, max_running_requests=2048)

    assert resolve_max_m(runner) == 8192 # 验证 resolve_max_m 能正确取到 mock 的最大值
```

评论区精华

本次 PR 无任何代码审查评论。PR body 中提供了清晰的变更说明和上下文，直接指出了问题的本质是测试 fixture 回归，而非生产代码 bug，并详细列出了具体修改点，起到了有效的自我审查作用。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。1. 回归风险：变更仅限于测试文件，且修复了现有测试，不会引入新的回归。2. 兼容性风险：将测试套件从 base-c 移至 base-a 属于 CI 配置调整，需确保 base-a-test-cpu 套件在 CI 环境中能正常运行此测试，从作者背景和标签看此调整是合理的。3. 隔离性风险：使用 mock 确保测试不依赖特定 ServerArgs 实现细节，增强了测试的健壮性。

- 影响：影响范围小且直接。 1. 对用户：无直接影响，仅修复了 CI 流程。 2. 对系统：确保了 Qwen3.5 MoE 相关单元测试在 Intel CPU CI 上能够稳定通过，维持了测试套件的可靠性。 3. 对团队：避免了 CI 红灯的干扰，使团队能更专注于核心功能开发。测试的正确模拟也为后续维护提供了更好的范例。
- 风险标记：CI 测试隔离

关联脉络

- PR #32733 【示例】此 PR 编号来自 PR body 提及，标题未知：PR body 明确指出当前修复的是 PR#32733 引入的测试 fixture 回归问题。