

PR #49945 完整报告

vllm-project/vllm

[Test] Skip ROCm AITER MLA prefill tests on non-ROCm platforms

合并时间: 2026-07-28 16:26

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/49945>

执行摘要

- 一句话: 跳过非 ROCm 平台上的 ROCm AITER MLA 测试
- 推荐动作: 该 PR 逻辑简单, 可以直接合并。它修复了一个明确的跨平台 CI 问题, 代码改动小且安全。值得注意的设计决策是使用 `current_platform.is_cuda_alike()` 而非特定平台检查 (如 `is_rocm`), 这更符合未来的扩展性。建议团队在编写依赖于特定硬件后端 (如 ROCm 特有模块) 的测试时, 统一采用类似的跳过策略。

功能与动机

`TestROCMaIterFAPrefillSelection` 测试在非 ROCm 平台 (如 Intel XPU) 上执行时, 因为 `vllm.platforms.rocm` 模块导入会触发 `torch.cuda.get_device_properties`, 而该平台的 PyTorch 未与 CUDA 一起构建, 导致测试崩溃。PR 描述明确指出该问题, 且 CI 结果验证了修复前 XPU CI 会因该测试挂起。

实现拆解

1. 新增导入语句: 在文件 `tests/v1/attention/test_mla_prefill_selector.py` 中, 新增 `from vllm.platforms import current_platform`, 用于获取当前运行平台信息。
2. 添加条件跳过装饰器: 在 `TestROCMaIterFAPrefillSelection` 类上方添加 `@pytest.mark.skipif(not current_platform.is_cuda_alike(), reason="...")`, 当平台不属于 CUDA 类 (即非 ROCm 或 CUDA) 时跳过整个测试类, 避免因平台不匹配导致的模块导入错误。

关键文件:

- `tests/v1/attention/test_mla_prefill_selector.py` (模块 预填充; 类别 test; 类型 test-coverage): 唯一修改的文件, 通过添加条件跳过装饰器和导入语句, 解决了非 CUDA/ROCm 平台上因测试导入 ROCm 模块导致的 CI 失败。

关键符号: 未识别

关键源码片段

`tests/v1/attention/test_mla_prefill_selector.py`

唯一修改的文件, 通过添加条件跳过装饰器和导入语句, 解决了非 CUDA/ROCm 平台上因测试导入 ROCm 模块导致的 CI 失败。

```

# tests/v1/attention/test_mla_prefill_selector.py
"""Tests for MLA prefill backend selector."""

from unittest.mock import MagicMock, patch

import pytest
import torch

from vllm.config import AttentionConfig, ModelConfig, VllmConfig
from vllm.platforms import current_platform # 新增导入, 用于判断当前平台
from vllm.platforms.interface import DeviceCapability
from vllm.v1.attention.backends.mla.prefill.base import MLADimensions
from vllm.v1.attention.backends.mla.prefill.registry import MLAPrefillBackendEnum
from vllm.v1.attention.backends.mla.prefill.selector import (
    MLAPrefillSelectorConfig,
    _auto_select_mla_prefill_backend,
    _get_mla_prefill_backend_priorities,
    get_mla_prefill_backend,
)

# ... 其他测试类 ...

# 添加条件跳过: 非 CUDA/ROCm 平台 (如 XPU、CPU、TPU) 跳过这个 ROCm 专用测试类
@pytest.mark.skipif(
    not current_platform.is_cuda_alike(),
    reason="Imports vllm.platforms.rocm, whose module init requires a CUDA or "
    "ROCm torch build; not importable on XPU/CPU/TPU.",
)

class TestROCmAiterFAPrefillSelection:
    """Tests for the ROCm AITER FlashAttention MLA prefill backend."""

    def test_rocm_priorities_prefer_aiter_fa(self):
        """On ROCm, ROCM_AITER_FA is tried first, FLASH_ATTN as fallback."""
        with patch("vllm.platforms.current_platform") as mock_platform:
            mock_platform.is_rocm.return_value = True
            priorities = _get_mla_prefill_backend_priorities(
                DeviceCapability(major=9, minor=5),
                MLADimensions(
                    qk_nope_head_dim=128,
                    qk_rope_head_dim=64,
                    v_head_dim=128,
                ),
            )
            assert priorities == [
                MLAPrefillBackendEnum.ROCM_AITER_FA,
                MLAPrefillBackendEnum.FLASH_ATTN,
            ]

    def test_supported_dtypes_are_fp16_bf16_only(self):

```

```
from vllm.v1.attention.backends.mla.prefill.aiter_flash_attn import (
    AiterFlashAttnPrefillBackend,
)
assert AiterFlashAttnPrefillBackend.supports_dtype(torch.bfloat16)
assert AiterFlashAttnPrefillBackend.supports_dtype(torch.float16)
# FP8 is served by the separate AITER ASM backend, not this one.
assert not AiterFlashAttnPrefillBackend.supports_dtype(torch.float8_e4m3fn)
```

评论区精华

代码审查中无实质讨论。合并者 [tjtanaa](#) 在 PR 评论中询问 Intel CI 测试结果，[jikunshang](#) 随后添加了 [Intel-gpu](#) 标签并提供了 CI 构建链接，确认了该修复在 Intel 平台上的验证通过。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。变更仅涉及测试代码，通过 `@pytest.mark.skipif` 安全跳过，不影响生产代码。若 `current_platform.is_cuda_alike()` 在 ROCm 平台上意外返回 `False`，会导致有效测试被跳过，但从代码看 `is_cuda_alike()` 在 ROCm 上返回 `True`，因此无实际风险。
- 影响：直接影响是 Intel XPU、CPU、TPU 等非 CUDA/ROCM 平台上的 CI 不会再因该测试失败。对 ROCm 和 CUDA 平台无影响，测试正常执行。对整个项目而言，这是一个最小的测试兼容性补丁，提高了跨平台 CI 稳定性。
- 风险标记：无风险，仅测试代码

关联脉络

- PR #49947 [Bugfix][KV Offload][OBJ] Preserve job completion during cleanup: 同为近期合并的测试兼容性修复，涉及 CI 稳定性改进。