PR #44761 完整报告

vllm-project/vllm

[ROCm][CI] Stabilizing teardown and timeout of flaky tests to prevent rare OOMs

合并时间：2026-06-08 14:11

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/44761

执行摘要

- 一句话：稳定 ROCm 测试清理流程，修复单位不匹配和超时问题
- 推荐动作：值得精读，尤其是 tests/utils.py 中关于 ROCm 内存清理的阈值的注释，以及对跨平台 GPU 内存查询单位不一致的修复模式。此 PR 展示了如何系统性解决测试环境中的 GPU 资源泄漏问题，是 CI 稳定性的良好实践。

功能与动机

PR body 中指出：'several hybrid/APC and pooling tests can leave enough residual VRAM on gfx942 to trip later test startup or teardown checks, even though the model allocations have already been released'，因此需要稳定测试的清理和 teardown 过程以防止罕见的 OOM。

实现拆解

1. 统一内存等待函数：在 tests/utils.py 中新增 wait_for_rocm_memory_to_settle 封装，内部调用 wait_for_gpu_memory_to_clear 并内置平台检查与合理的默认参数（4 GiB 最小阈值），避免各测试文件重复相同的等待逻辑。
2. 修复 MiB/bytes 单位不一致：_get_gpu_memory_used 中 ROCm 分支原直接返回 amdsmi 给出的 MiB 值，而 CUDA 分支返回 bytes，导致 _wait_for_gpu_memory_release 的字节比较永远提前满足。现在将 MiB 显式乘以 1024*1024 转为 bytes，使等待逻辑生效。
3. 增强 HfRunner/VllmRunner 的 teardown：在 tests/conftest.py 中，HfRunner.__exit__ 和 VllmRunner.__exit__ 于清理后调用 wait_for_rocm_memory_to_settle；VllmRunner.__exit__ 中为 engine_core.shutdown 设置了 60 秒超时（仅 ROCm），避免硬杀死导致的 VRAM 残留。同时将 VllmRunner._wait_for_rocm_memory_release 中的等待逻辑替换为对共享函数的调用。
4. 简化 hybrid 测试清理：删除 test_hybrid.py 中的本地 _wait_for_rocm_memory_to_settle 函数，将其集成到共享工具中；移除 _owned_vLLM_runner 的 finally 等待块，因为清理责任已上移至高层的 runner teardown。
5. ColBERT 测试增加上下文管理器：在 test_colbert.py 中新增 _hf_colbert_model 上下文管理器，确保 HF 模型和权重在测试退出时被显式删除并通过 cleanup_dist_env_and_memory 和 wait_for_rocm_memory_to_settle 释放 GPU 内存，

避免模块级 fixture 跨测试残留 VRAM。

关键文件：

- tests/utils.py（模块 测试工具；类别 test；类型 test-coverage；符号 wait_for_rocm_memory_to_settle, wait_for_gpu_memory_to_clear, _get_gpu_memory_used）：新增核心工具函数 wait_for_rocm_memory_to_settle 并修复单位转换 bug，是整个变更的基础。
- tests/conftest.py（模块 测试配置；类别 test；类型 test-coverage；符号 HfRunner.exit, VllmRunner.exit, VllmRunner._wait_for_rocm_memory_release）：修改了 HfRunner 和 VllmRunner 的 teardown 逻辑，增加统一的清理等待和更长的 shutdown 超时。
- tests/models/language/generation/test_hybrid.py（模块 混合测试；类别 test；类型 test-coverage；符号 _owned_vllm_runner）：删除了本地等待函数，统一使用共享工具函数，并简化了上下文管理器，展示了清理逻辑的上移。
- tests/models/language/pooling/test_colbert.py（模块 ColBERT 测试；类别 test；类型 test-coverage；符号 _hf_colbert_model）：新增 _hf_colbert_model 上下文管理器确保 HF 模型在测试退出时正确清理，避免跨 fixture VRAM 残留。

关键符号：wait_for_rocm_memory_to_settle, wait_for_gpu_memory_to_clear, RemoteOpenAIServer._get_gpu_memory_used, _hf_colbert_model, HfRunner.exit, VllmRunner.exit, VllmRunner._wait_for_rocm_memory_release

关键源码片段

tests/utils.py

新增核心工具函数 wait_for_rocm_memory_to_settle 并修复单位转换 bug，是整个变更的基础。

```
def wait_for_rocm_memory_to_settle(threshold_ratio: float = 0.01) -> None:
    """Wait until ROCm GPU memory usage drops below a threshold ratio.

    On ROCm, VRAM is freed asynchronously by the driver.  After a model is
    destroyed, the driver may keep the allocation resident for a short time.
    Calling this before starting a new test avoids an OOM on the startup
    guard.
    """
    from vllm.platforms import current_platform
    if not current_platform.is_rocm():
        return
    num_gpus = current_platform.device_count()
    if num_gpus == 0:
        return
    # Use a conservative 4 GiB minimum threshold because the ROCm driver
    # itself keeps ~2.5 GiB resident even when no model is loaded. A strict
    # 1% ratio would never be satisfied and would hang the test.
    wait_for_gpu_memory_to_clear(
        devices=list(range(num_gpus)),
        threshold_ratio=threshold_ratio,
```

```
        timeout_s=120,
    )
```

tests/conftest.py

修改了 HfRunner 和 VllmRunner 的 teardown 逻辑，增加统一的清理等待和更长的 shutdown 超时。

```
def __exit__(self, exc_type, exc_value, traceback):
    from tests.utils import wait_for_rocm_memory_to_settle
    # 显式删除模型，然后等待 ROCm 异步释放 VRAM，
    # 避免下一个 runner 启动时因残留内存而导致 OOM。
    del self.model
    cleanup_dist_env_and_memory()
    wait_for_rocm_memory_to_settle()
```

评论区精华

无实质性讨论；仅 claude[bot] 自动评论（因 fork 无法运行），DarkLight1337 直接批准。

- 暂无高价值评论线程

风险与影响

- 风险：所有变更仅限于测试基础设施，不影响产品代码。主要风险包括：
  - 新增的 60 秒 shutdown 超时可能让测试总耗时增加，但仅对 ROCm 生效；
  - 统一的内存等待函数如果阈值选择不当，可能导致测试超时或等待不足，但已有充分注释并基于经验值（4 GiB）设置；
  - 单位转换修复可能暴露之前被掩盖的其他问题，但这是正确的行为。
  - 影响：影响范围：仅 ROCm CI 上的 hybrid/APC/pooling 测试。影响程度：预期显著降低随机 OOM 导致的失败率，提升 CI 可靠性。对其他平台无影响。对用户无影响，团队可获得更稳定的测试反馈。
  - 风险标记：测试超时增加，阈值可能不适用所有 GPU

关联脉络

- 暂无明显关联 PR