PR #49510 完整报告

vllm-project/vllm

[CI] Isolate cudagraph tests in child processes

合并时间：2026-07-24 01:16

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/49510

执行摘要

- 一句话：将 cudagraph 测试隔离到子进程，删除手动 GPU 内存清理
- 推荐动作：此 PR 值得阅读以了解测试隔离的最佳实践，尤其当测试涉及 GPU 资源时。设计决策可复用于其他类似测试文件。

功能与动机

PR body 指出：cudagraph 测试用例复用 pytest 进程生命周期，导致 VRAM 释放依赖非确定性的循环引用收集。通过将每个测试标记为 fork，子进程退出成为确定性的 GPU 内存所有权边界。这稳定了 Cudagraph 测试组。

实现拆解

1. 替换 GPU 内存清理机制：在文件 tests/v1/cudagraph/test_cudagraph_mode.py 中，删除 import weakref、from tests.utils import wait_for_gpu_memory_to_clear 以及测试函数中对应的 weakref.proxy、del llm、wait_for_gpu_memory_to_clear 代码块。
2. 引入子进程隔离：为两个测试函数 test_backend_and_cudagraph_mode_combo 和 test_cudagraph_compilation_combo 添加 @create_new_process_for_each_test("spawn") 装饰器，使每个测试在独立子进程中启动，确保 GPU 内存随进程退出而释放。
3. 简化清理逻辑：移除异常处理中 try-except 和 finally 子句中对 llm 变量的弱引用处理，因为在子进程退出后，所有 GPU 资源会自动回收。

关键文件：

- tests/v1/cudagraph/test_cudagraph_mode.py（模块 测试；类别 test；类型 test-coverage）：唯一修改的文件，核心变更：引入子进程隔离并删除手动 GPU 内存清理代码。

关键符号：未识别

关键源码片段

tests/v1/cudagraph/test_cudagraph_mode.py

唯一修改的文件，核心变更：引入子进程隔离并删除手动 GPU 内存清理代码。

```
# SPDX-License-Identifier: Apache-2.0
# SPDX-FileCopyrightText: Copyright contributors to the vLLM project
```

```
from contextlib import ExitStack

import pytest

# 新导入：子进程隔离装饰器
from tests.utils import create_new_process_for_each_test
from tests.v1.attention.utils import full_cg_backend_configs as backend_configs
from vllm import LLM
from vllm.config import CompilationConfig, CompilationMode
from vllm.platforms import current_platform

# 测试 attention backend 与 cudagraph_mode 组合
# (backend_name, cudagraph_mode, supported)
if current_platform.is_rocm():
    combo_cases_1 = [
        ("RocmAttn", "FULL", True),
        ("RocmAttn", "FULL_AND_PIECEWISE", True),
        ("TritonAttn", "FULL", True),
        ("TritonAttn", "FULL_AND_PIECEWISE", True),
    ]
else:
    combo_cases_1 = [
        ("FA3", "FULL", True),
        ("FA3", "FULL_AND_PIECEWISE", True),
        ("FA2", "FULL", True), # Should fallback to FULL_AND_PIECEWISE
        ("FA2", "FULL_AND_PIECEWISE", True),
        ("FlashInfer", "FULL", True), # Should fallback to FULL_AND_PIECEWISE
        ("FlashInfer", "FULL_AND_PIECEWISE", True),
    ]


@pytest.mark.parametrize("backend_name, cudagraph_mode, supported", combo_cases_1)
# 新增：每个测试在独立子进程中以 "spawn" 模式运行
@create_new_process_for_each_test("spawn")
def test_backend_and_cudagraph_mode_combo(backend_name, cudagraph_mode, supported):
    # ... ( 测试体保持不变，但删除了 weakref 和 GPU 清理代码 )
    ...
```

评论区精华

无 review 评论或讨论。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。变更仅涉及测试文件，通过子进程隔离替代手动内存清理，理论上更安全。需要确保 create_new_process_for_each_test 装饰器在 CI 环境（如 ROCm）中正常工作，但该助手已在其他测试中使用。

- 影响：影响范围仅限于 cudagraph 测试组。测试变得更稳定，不再依赖非确定性 GC，有利于 CI 一致性。对用户和生产系统无影响。
- 风险标记：暂无

关联脉络

- PR #48399 [Core] Simplify KVBlockZeroer index tensor handling: 同为 V1 模块的测试清理相关 PR，类似地删除了冗余代码。