

PR #50726 完整报告

vllm-project/vllm

[CI][ROCm] Export Helion benchmark script in test artifacts

合并时间: 2026-08-04 02:09

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/50726>

执行摘要

- 一句话: 修复 ROCm CI 缺 Helion 脚本导致的测试收集失败
- 推荐动作: 值得快速浏览, 尤其适合 CI 基础设施维护者阅读。它展示了一个容易被忽视的模式: 构建测试镜像时导出工件清单必须与测试代码的 import 依赖保持同步, 并用 `source_file_dependencies` 固化触发关系。改动极小, 无需精读。

功能与动机

PR body 明确指出: PR #48968 添加的 Helion 数值测试依赖 `scripts/benchmark_helion_kernels.py`, 而 ROCm CI 是从较小的导出工件 (`export_vllm / export_vllm_wheel_release` 阶段) 重建 `/vllm-workspace`, 并非完整签出, 导致 AMD 作业在测试收集阶段失败。本 PR 的目标就是补齐导出工件清单并建立触发关系, 防止类似问题再次发生。

实现拆解

1. 在 `docker/Dockerfile.rocm` 的 `export_vllm` 阶段 (约第 325 行, 位于 `benchmarks / tests / examples` 的 `COPY` 之后) 追加 `COPY --from=build_vllm ${COMMON_WORKDIR}/vllm/scripts/benchmark_helion_kernels.py /scripts/benchmark_helion_kernels.py`, 使常规导出镜像携带该脚本。
2. 在 `docker/Dockerfile.rocm` 的 `export_vllm_wheel_release` 阶段 (约第 641 行) 追加相同 `COPY`, 保证基于 `wheel release` 的导出镜像同样具备该文件, 避免两类 ROCm 测试环境行为不一致。
3. 在 `.buildkite/test-amd.yaml` 的 `Kernels Helion Test` 作业 `source_file_dependencies` 中追加 `scripts/benchmark_helion_kernels.py`, 使脚本后续变更能自动触发该作业, 形成“依赖即触发源”的闭环。
4. 保留 `tests/kernels/helion` 中既有的 `from scripts import benchmark_helion_kernels` 导入, 并明确不触碰 PR #48646 的 `ci_base` 缓存复用逻辑——即使基础镜像可复用, 导出阶段仍从当前 `checkout` 构建, 新增 `COPY` 总会生效。

关键文件:

- `docker/Dockerfile.rocm` (模块 镜像构建; 类别 `infra`; 类型 `infrastructure`): 修复的核心: 在 `export_vllm` 与 `export_vllm_wheel_release` 两个 ROCm 导出阶段各新增一条 `COPY`, 把 `scripts/benchmark_helion_kernels.py` 带入导出工件, 解决测试收集阶段找不到文件

的问题。

- .buildkite/test-amd.yaml (模块 CI 配置; 类别 config; 类型 configuration) : 辅助配置: 把脚本加入 Kernels Helion Test 作业的 source_file_dependencies, 使该文件变化时自动触发作业, 防止依赖与工件再次脱节。

关键符号: 未识别

关键源码片段

docker/Dockerfile.rocm

修复的核心: 在 export_vllm 与 export_vllm_wheel_release 两个 ROCm 导出阶段各新增一条 COPY, 把 scripts/benchmark_helion_kernels.py 带入导出工件, 解决测试收集阶段找不到文件的问题。

```
# 从构建阶段复制测试相关资源到导出阶段, 保证 ROCm CI 的 /vllm-workspace 是自洽的产物
COPY --from=build_vllm ${COMMON_WORKDIR}/vllm/benchmarks /benchmarks
COPY --from=build_vllm ${COMMON_WORKDIR}/vllm/tests /tests
COPY --from=build_vllm ${COMMON_WORKDIR}/vllm/examples /examples
```

```
# 新增: Helion 测试在收集阶段就导入该脚本, 缺少会导致 pytest 收集失败
COPY --from=build_vllm ${COMMON_WORKDIR}/vllm/scripts/benchmark_helion_kernels.py /
scripts/benchmark_helion_kernels.py
```

评论区精华

评论区没有实质技术争议。claude[bot] 仅自动提示该仓库采用手动 code review 流程; 审核人 dllehr-amd 直接 APPROVED, 评价“Thanks Andreas! Looks pretty straightforward!”。可视为无设计分歧、变更被一致认可通过。

- 变更内容确认与审批 (other): 无实质技术争议, 变更被认定为直接明了的 CI 修复, 批准合并。

风险与影响

- 风险:
 1. 回归风险低: 两条 COPY 均为新增指令, 不修改既有导出内容; 若未来脚本被删除, Docker build 会提前 fail-fast, 问题会在更早阶段暴露。
 2. 兼容性: 只影响 ROCm 镜像导出与 CI 触发, 对运行时推理、模型加载无任何影响。
 3. CI 触发面: 新增 source_file_dependencies 后, 脚本每次改动都会触发 Helion 作业; 该作业标记为 optional, 且改动频率低, 额外开销可忽略。
 4. 不确定点: PR body 未说明 CUDA 侧为何未出现同类问题, 推断是 CUDA 从完整 checkout 运行、ROCm 从导出工件运行所致, 但缺少显式证据。- 影响: 对最终用户无影响; 对 ROCm CI 是修复性变更, 恢复 Kernels Helion Test 的可运行性, 解除 AMD Buildkite 作业在测试收集阶段的持续失败。对团队而言, 该 PR 建立了“测试新增导入的脚本必须同步加入导出工件清单与 source_file_dependencies”的维护约定, 为后续 Helion 测试迭代提供触发保障。- 风险标记: 低风险 CI 修复, 导出依赖需与测试同步

关联脉络

- PR #48968 [CI][ROCm] Add Helion numerical tests (依据 PR body 推断) : 该 PR 引入 Helion 数值测试及对 scripts/benchmark_helion_kernels.py 的导入, 是本 PR 要补齐的依赖源头; 其未把脚本加入 ROCm 导出工件导致 AMD CI 失败。
- PR #48646 [CI] Reuse ci_base cache (依据 PR body 提及) : PR body 明确说明本 PR 的导出依赖要与 ci_base 缓存解耦, 即使复用基础镜像, 导出阶段仍从当前 checkout 重建, 因此新增 COPY 始终生效。