

PR #44969 完整报告

vllm-project/vllm

[ROCm][CI] Gating more ROCm tests

合并时间: 2026-08-19 06:45

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/44969>

执行摘要

- 一句话: 为 Stage D 门禁新增 15 个 ROCm mirror, 扩展 AMD CI 覆盖
- 推荐动作: 值得精读。这是学习 vLLM 大规模 CI 门禁设计的好样本: mirror 机制如何在保持 NVIDIA 步骤不动的前提下横向扩展硬件覆盖, 以及 `source_file_dependencies` 如何精细化控制触发面。重点留意两点: ① mamba 个人 fork 引用需要长期跟踪, 建议在官方 v2.3.1 验证通过后替换为正式 tag; ② 可以将本 PR 的硬件队列规划 (MI250/MI300/MI355 的分工) 与后续 ROCm CI 演进 PR 对照阅读, 理解 Stage D/E 门禁分层的迭代逻辑。

功能与动机

PR body 明确说明目标: "Add 15 AMD mirrors for existing CI test groups as the Stage D gating expansion.", 即把原本只在 NVIDIA 硬件上运行的测试组通过 mirror 机制扩展到 AMD 硬件, 让 ROCm 相关代码改动在合入前就受到同等力度的自动化测试保护。作者在评审中补充解释, AMD 侧没有专门维护 mamba 的团队, 因此选择用个人 fork 分支来控制 ROCm CI 中安装的构建来源。

实现拆解

1. 确定测试分组与硬件配额: 按 PR body 的规划映射 5 类硬件队列——mi250_2 (Plugin 测试 2 GPU)、mi300_1 (PyTorch Compilation、Speculators Correctness)、mi300_2 (Model Runner V2 Distributed、Distributed Model Tests 2 GPU)、mi300_4 (V1 E2E、LoRA TP Distributed)、mi355_1 (Entrypoints、Multi-Modal、Async Engine 等)。
2. 在测试步骤上追加 mirror 块: 每个变更文件都是给已有的 NVIDIA 步骤插入 mirror:
amd: 配置段, 统一设置 `dind: false`、`depends_on: image-build-amd`、独立的 `timeout_in_minutes` (全部按 5 分钟取整)。关键适配点是 `source_file_dependencies`: 除复用原 NVIDIA 依赖外, 额外纳入 `vllm/platforms/rocm.py`、`vllm/v1/attention/backends/`、`vllm/_aiter_ops.py` 等 ROCm 特有路径 (见 `models_distributed.yaml`、`model_runner_v2.yaml`), 确保 AITER、平台层变更也会触发 AMD mirror。
3. 命令层适配: 将 `TARGET_TEST_SUITE=L4` 替换为 `TARGET_TEST_SUITE=MI300`, `CUDA_VISIBLE_DEVICES` 替换为 `HIP_VISIBLE_DEVICES`, 其余 pytest marker (如 `distributed(num_gpus=2)`、`not core_model`) 与 NVIDIA 侧保持一致 (见

models_distributed.yaml、model_runner_v2.yaml、lora.yaml)。

4. 特殊依赖与门禁策略: models_language.yaml 的 MI355 mirror 将 mamba 安装源指向 AndreasKaratzas/mamba@fix-rocm-7.0-warp-size-constexpr 以绕过 ROCm 7.0 编译问题; engine.yaml 额外将 NVIDIA 步骤 label 从 "V1 E2E" 细化为 "V1 E2E Hybrid Chunked Prefill"。
5. 门禁节奏调整: 提交历史显示先后经历 "设置 optional → 正式纳入 gating"、将 6 个 MI250 mirror 挪至 Stage E、移除 Model Runner V2 Spec Decode mirror、移除 AMD compile passes mirror 等多轮收敛, 最终保留 15 个新 mirror。
6. 评审中涉及的配套修复未进入最终 diff: 作者曾在 issue 评论中说明修改了 engine output socket 路径 (避免复用可变的 msgpack bytearray 做 copy=False 的 ZMQ 零拷贝发送), 并尝试默认使用 HIP silu_and_mul 以规避 Inductor 数值漂移; 后者随后被 revert ("Reverted the silu part... until we debug the inductor part"), 最终提交只保留 YAML 配置变更。

关键文件:

- .buildkite/test_areas/models_distributed.yaml (模块 分布式模型; 类别 config; 类型 configuration): 本 PR 覆盖最重的测试组 mirror: 在 MI300 (mi300_2) 上运行 2-GPU 分布式模型测试, 命令层完整替换为 TARGET_TEST_SUITE=MI300/HIP_VISIBLE_DEVICES, 并在依赖中纳入 attention 后端、_aiter_ops、rocm.py 等 ROCm 特有路径。
- .buildkite/test_areas/models_language.yaml (模块 语言模型; 类别 config; 类型 configuration): 唯一产生实质评审交锋的文件: MI355 mirror 将 mamba 安装源指向作者个人 fork 分支, 被 bot 标记为 MEDIUM 供应链风险, 作者以缺少官方维护为由拒绝修改。
- .buildkite/test_areas/model_runner_v2.yaml (模块 模型运行器; 类别 config; 类型 configuration): 将 V2 Model Runner 分布式测试 (basic_correctness、async_llm_dp、eagle_dp) 纳入 MI300 门禁, 并保留 VLLM_USE_V2_MODEL_RUNNER/NCCL_CUMEM_HOST_ENABLE 等环境变量配置。
- .buildkite/test_areas/entrypoints.yaml (模块 入口测试; 类别 config; 类型 configuration): MI355 上新增 4 个 Entrypoints mirror (Unit、Responses API、Multimodal、多模态 processor 相关), 覆盖前端 API 层在 AMD 侧的回归。
- .buildkite/test_areas/lora.yaml (模块 LoRA 测试; 类别 config; 类型 configuration): LoRA TP 分布式测试 (含 chatglm3/llama/qwen3/olmoe/gptoss/gemma4 多 TP 用例) 在 mi300_4 上获得 mirror, 补齐多卡 TP 场景的 AMD 覆盖。
- .buildkite/test_areas/engine.yaml (模块 引擎测试; 类别 config; 类型 configuration): V1 E2E Hybrid Chunked Prefill 在 mi300_4 上获得 mirror, 同时 NVIDIA 侧 label 细化以区分测试内容。
- .buildkite/test_areas/models_multimodal.yaml (模块 多模态模型; 类别 config; 类型 configuration): 多模态 processor、Extended Generation 2/3 等多个步骤分别进入 mi355_1, 扩展多模态生成在 AMD 的覆盖。
- .buildkite/test_areas/plugins.yaml (模块 插件测试; 类别 config; 类型 configuration): Plugin 集成测试进入 mi250_2 队列, 是 MI250 硬件在本次门禁扩展中的主要落点。

- `.buildkite/test_areas/pytorch.yaml` (模块 编译测试; 类别 config; 类型 configuration) : PyTorch Compilation 单测进入 `mi300_1`, 依赖 `csrc/` 与 `rocm.py`, 保护编译链路在 AMD 的可用性。
- `.buildkite/test_areas/misc.yaml` (模块 杂项测试; 类别 config; 类型 configuration) : Async Engine、Inputs、Utils、Worker 等基础组件测试进入 `mi355_1`, 补全非模型类逻辑的 AMD 回归。
- `.buildkite/test_areas/spec_decode.yaml` (模块 推测解码; 类别 config; 类型 configuration) : Spec decode 相关测试组获得 AMD mirror, 与作者在评论中提到的 ngram speculative logprob 数值问题形成呼应。

关键符号: 未识别

评论区精华

核心讨论集中在 `models_language.yaml:126` 的供应链风险上:

- `depthfirst-app[bot]` (MEDIUM 严重度) : `MAMBA_FORCE_BUILD=TRUE` 会编译并执行来自 个人 GitHub fork 的分支引用, 分支可被 `force-push` 注入任意代码; 上游修复已在 `state-spaces/mamba#831` 合入并发布为正式版 `v2.3.1`, 建议改用官方 tag 以消除供应链风险。
- AndreasKaratzas 回应: "that's my branch ② the reason was because we do not have a team maintaining mamba and we want to be in control of what we are installing in CI for ROCm."——即选择个人 fork 是有意为之, 是为了在没有官方维护团队的情况下控制 ROCm CI 的安装内容。该分歧最终未在本 PR 内解决, 维持个人分支引用。

此外, 作者的两条长评论披露了测试失败根因分析过程: ZMQ 零拷贝发送复用 `msgpack` 缓冲区导致接收端偶发 `malformed msgpack` 错误; ROCm 基础镜像升级后 Inductor 原生 `SiLU-and-mul` 路径出现数值漂移, 足以让 `ngram speculative logprob` 检查失败, 本地复现确认 HIP custom op 可通过, 但最终因修复复杂度选择 `revert` 并回退旧基础镜像。

- mamba 个人 fork 分支的供应链风险 (security): 作者回复 "that's my branch ②", 解释 AMD 侧没有团队维护 mamba, 选择个人 fork 是为了控制 ROCm CI 的安装来源; 最终保留个人分支引用, 该问题未在本 PR 内解决。
- ZMQ 零拷贝发送复用 `msgpack` 缓冲区的正确性 (correctness): 修改为每个编码缓冲区存活到 ZMQ 发送完成; 大 tensor payload 仍可零拷贝。此修复为 PR 推进过程中的伴随改动, 最终未出现在最终 diff 中。
- ROCm Inductor `SiLU-and-mul` 数值漂移 (performance): 作者最初默认切到 HIP custom op, 随后 `revert` ("Reverted the silu part... until we debug the inductor part"), 回退旧基础镜像待后续调试, 避免在本 PR 中带入复杂修复。

风险与影响

- 风险:
 1. 供应链风险 (高) : `models_language.yaml` 使用个人 fork 的可变分支 `fix-rocm-7.0-warp-size-constexpr` 安装 mamba, 配合

MAMBA_FORCE_BUILD=TRUE 会在 CI 中编译执行任意代码；该分支可被 force-push，触发路径为任何涉及 vllm/ 的改动。author 已明确不接受修改建议，需要持续跟踪上游 v2.3.1 在 ROCm 7.0 的可用性。

2. CI 资源与稳定性风险：新增 15 个 gating mirror 将显著提升 AMD 队列负载；若 MI355/MI250 队列调度不稳定，可能拖慢 ROCm 相关 PR 的合入速度。
3. 触发覆盖偏差：mirror 的 source_file_dependencies 与 NVIDIA 步骤不完全一致（如 models_multimodal.yaml 的多模态步骤依赖 vllm/ 全量，而 processor 步骤依赖较窄），存在 " 改动应触发未触发 " 的缺口，例如仅改 vllm/platforms 时部分镜像不会运行。
4. 门禁误报风险：TARGET_TEST_SUITE 由 L4 改为 MI300，若上游测试代码对该 env 有硬编码行为（如路径或 marker 判断），可能出现 NVIDIA 通过但 AMD 误失败的情况；长跑 68 个 commit 的 merge 历史也增加了最终合入时配置漂移的不确定性。- 影响：对 开发者：ROCm 相关改动（attention 后端、AITER、platforms、模型代码）会触发更多 AMD 硬件上的自动化测试，回归发现更早；对 CI 系统：新增 15 个 Buildkite mirror，覆盖 mi250_2/mi300_1/mi300_2/mi300_4/mi355_1 五种队列，首次将 MI355 大规模引入门禁；对 最终用户：无任何运行时影响（纯 CI 配置）。整体影响半径集中在 vLLM 开发与 CI 基础设施侧。- 风险标记：供应链依赖，CI 资源消耗，触发覆盖缺口，门禁误报风险，长 merge 历史

关联脉络

- 暂无明显关联 PR