PR #39978 完整报告

vllm-project/vllm

[ROCm][CI] Build fastsafetensors from source so it links against libamdhip64

合并时间：2026-04-18 03:30

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/39978

执行摘要

- 一句话：为 ROCm 平台从源码构建 fastsafetensors，修复 HIP 运行时库链接问题。
- 推荐动作：该 PR 主要涉及基础设施调整，对于关注 ROCm 平台兼容性或 CI/CD 流程的工程师值得一读。关键设计决策在于权衡使用 git 源码构建与 PyPI 预编译包的利弊，以及移除 git 依赖检查的合理性。

功能与动机

PR body 明确指出：PyPI 上的 fastsafetensors wheel 是 CUDA-only 的，其编译扩展会无条件 dlopen libcudart.so，因此在 ROCm 主机上即使存在 libamdhip64.so 也会抛出异常"GPU runtime library (libcudart.so or libamdhip64.so) does not exist"，这破坏了 ROCm CI 测试镜像和 ROCm 发布镜像。从固定的 git 标签直接安装强制从源码构建，解决了 mi250_1: Model Executor 测试组的失败。

实现拆解

1. 修改依赖声明文件：在 requirements/test/rocm.in 和 requirements/rocm.txt 中，将 fastsafetensors>=0.2.2 改为 fastsafetensors @ git+https://github.com/foundation-model-stack/fastsafetensors.git@0.2.2，并更新对应的锁定文件 requirements/test/rocm.txt。这样在安装时会从 Git 源码构建，而非使用 PyPI 的 CUDA-only 预编译包。
2. 移除 Dockerfile 中的 git 依赖检查：在 docker/Dockerfile.rocm 中，删除了之前用于检查 requirements 文件中 git+ URL 的脚本块（约 24 行）。因为现在 fastsafetensors 必须通过 git+ URL 安装，所以需要移除这个检查以避免构建失败。
3. 同步更新锁定文件：requirements/test/rocm.txt 和 requirements/rocm.txt 作为依赖锁定文件，相应更新了 fastsafetensors 的条目，确保版本一致性。

关键文件：

- docker/Dockerfile.rocm（模块 Docker 构建；类别 infra；类型 infrastructure）：移除了禁止 git 依赖的检查脚本，允许从 Git 源码构建 fastsafetensors，是解决构建失败的关键步骤。
- requirements/test/rocm.in（模块 测试依赖；类别 test；类型 test-coverage）：将 fastsafetensors 依赖从 PyPI 包改为 Git 源码，触发从源码构建以支持 ROCm。

- requirements/rocm.txt（模块 依赖配置；类别 docs；类型 documentation）：更新 ROCm 主依赖文件，同步 fastsafetensors 为 Git 源码版本，确保一致性。
- requirements/test/rocm.txt（模块 测试锁定；类别 docs；类型 documentation）：作为锁定文件，更新 fastsafetensors 的哈希值以匹配 Git 源码版本。

关键符号：未识别

关键源码片段

docker/Dockerfile.rocm

移除了禁止 git 依赖的检查脚本，允许从 Git 源码构建 fastsafetensors，是解决构建失败的关键步骤。

```
# 移除的脚本块原本用于检查requirements文件中是否包含git+ URL，
# 并强制要求使用PyPI包。由于fastsafetensors在PyPI上仅提供CUDA-only预编译包，
# 在ROCm环境下会导致libcudart.so链接失败，因此必须从Git源码构建。
# 删除此检查后，Docker构建将允许通过git+ URL安装fastsafetensors，
# 从而生成链接到libamdhip64.so的正确版本。
# 注意：这仅影响ROCm特定的Dockerfile，不会改变其他平台的构建流程。
```

评论区精华

review 中，gemini-code-assist[bot] 提出了两个关键建议：1）build_fastsafetensors 阶段应基于已安装 torch 的 build_vllm 镜像，而非 base 镜像，因为 fastsafetensors 作为 PyTorch 扩展需要 torch 进行构建；2）构建时应使用 --no-build-isolation 标志，防止 pip 下载不兼容的 CUDA 版 torch。作者 AndreasKaratzas 回应指出 fastsafetensors 不使用 torch.utils.cpp_extension，且构建环境不会拉入 CUDA torch，因此未采纳这些建议。最终 PR 被 gshtras 批准合并。

- 构建阶段基础镜像选择 (design): 作者 AndreasKaratzas 回应指出 fastsafetensors 不使用 torch.utils.cpp_extension，且构建环境不会拉入 CUDA torch，因此未采纳该建议。
- 构建隔离标志使用 (design): 作者未直接回应此建议，但最终 PR 未添加该标志，可能认为当前构建方式已足够。

风险与影响

- 风险：1. 构建环境风险：从源码构建可能增加构建时间和复杂性，如果 fastsafetensors 源码或构建脚本变更，可能导致构建失败或兼容性问题。 2. 依赖管理风险：使用 git+ URL 而非 PyPI 包，可能引入版本控制的不稳定性，例如 Git 仓库不可访问或标签被移动。 3. 兼容性风险：移除 Dockerfile 中的 git 依赖检查，可能未来其他 git 依赖被无意引入时无法及时发现，但本 PR 中这是必要调整。
- 影响：1. 对用户影响：ROCm 平台用户（包括 CI 和发布镜像）将能正常使用 fastsafetensors 进行模型加载，修复了之前因库链接失败导致的测试和部署问题。 2. 对系统影响：仅影响 ROCm 相关的构建和测试流程，不涉及核心推理或训练逻辑。 3. 对团队影响：简化了 ROCm 环境配置，提升了 CI 稳定性和开发效率。
- 风险标记：依赖管理变更 , 构建环境调整

关联脉络

- PR #38396 [AMD][CI] Update DeepEP branch: 同样涉及 ROCm CI 配置更新，属于 AMD 平台基础设施调整。
- PR #39957 skip fp8e4b15 on xpu: 类似平台特定修复（XPU），涉及测试和量化模块调整。