

PR #48126 完整报告

vllm-project/vllm

Add XPU nightly and release image publishing to DockerHub

合并时间: 2026-07-10 15:56

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/48126>

执行摘要

- 一句话: 新增 XPU 镜像的 nightly 和 release 发布流程
- 推荐动作: 建议阅读此 PR 以了解 XPU 镜像发布流程的设计。如果团队计划增加其他硬件后端, 可以参考此模式。值得关注的点是 `push-nightly-builds-xpu.sh` 中使用了 `docker manifest` 操作, 以及如何在 pipeline 中组织依赖。

功能与动机

XPU (Intel GPU) 镜像之前仅发布到 ECR, 社区用户难以使用。本 PR 扩展了现有的 nightly 和 release 发布流程, 使 XPU 镜像能够自动发布到 DockerHub, 便于用户通过 `docker pull vllm/vllm-openai-xpu` 获取。

实现拆解

1. 发布流水线配置 (`.buildkite/release-pipeline.yaml`): 在 ROCm nightly 步骤之前新增了 Publish nightly XPU image to DockerHub 步骤, 依赖 `create-manifest-xpu`; 并在 release 发布的 block 中增加 `create-manifest-xpu` 依赖。
2. 新增 nightly 发布脚本 (`.buildkite/scripts/xpu/push-nightly-builds-xpu.sh`): 从 ECR 拉取 COMMIT-x86_64-xpu 镜像, 重新标记并推送 nightly-x86_64 标签, 然后清理旧 manifest 并创建 / 推送 nightly 和 nightly-COMMIT manifest。
3. 扩展 release 发布脚本 (`.buildkite/scripts/publish-release-images.sh`): 在 ROCm 和 CPU 处理之后添加了 XPU 镜像的拉取、标记为 latest-x86_64 和 `v${RELEASE_VERSION}-x86_64`、推送, 以及创建 / 推送 latest 和 `v${RELEASE_VERSION}` manifest。
4. 无测试、配置或架构配套改动: 本 PR 是纯 CI 流程变更。

关键文件:

- `.buildkite/release-pipeline.yaml` (模块 发布流水线; 类别 config; 类型 configuration): 发布流水线配置, 定义了 nightly XPU 发布步骤及其依赖关系, 是入口文件。
- `.buildkite/scripts/xpu/push-nightly-builds-xpu.sh` (模块 XPU 发布脚本; 类别 other; 类型 core-logic): 新增脚本, 负责从 ECR 拉取 XPU 镜像并推送 nightly 标签到 DockerHub。
- `.buildkite/scripts/publish-release-images.sh` (模块 镜像发布脚本; 类别 other; 类型 core-logic): 在现有 release 发布脚本中添加了对 XPU 镜像的处理, 包括拉取、tag、manifest 创建和推送。

关键符号：未识别

关键源码片段

[.buildkite/scripts/xpu/push-nightly-builds-xpu.sh](#)

新增脚本，负责从 ECR 拉取 XPU 镜像并推送 nightly 标签到 DockerHub。

```
#!/bin/bash

set -ex

ORIG_TAG_NAME="$BUILDKITE_COMMIT"
REPO="vllm/vllm-openai-xpu"

echo "Pushing original XPU tag ${ORIG_TAG_NAME}-xpu to nightly tags in ${REPO}"

# 登录 ECR 并拉取构建产物
aws ecr-public get-login-password --region us-east-1 | docker login --username AWS --password-stdin public.ecr.aws/q9t5s3a7
docker pull public.ecr.aws/q9t5s3a7/vllm-release-repo:"$ORIG_TAG_NAME"-x86_64-xpu

# 标记并推送架构特定 nightly 标签
docker tag public.ecr.aws/q9t5s3a7/vllm-release-repo:"$ORIG_TAG_NAME"-x86_64-xpu ${REPO}:nightly-x86_64
docker push ${REPO}:nightly-x86_64

# 清理可能存在的旧 manifest（忽略错误）
docker manifest rm ${REPO}:nightly || true
docker manifest rm ${REPO}:nightly-"$BUILDKITE_COMMIT" || true

# 创建并推送 multi-arch manifest（当前只有 x86_64）
docker manifest create ${REPO}:nightly ${REPO}:nightly-x86_64 --amend
docker manifest create ${REPO}:nightly-"$BUILDKITE_COMMIT" ${REPO}:nightly-x86_64 --amend
docker manifest push ${REPO}:nightly
docker manifest push ${REPO}:nightly-"$BUILDKITE_COMMIT"
```

[.buildkite/scripts/publish-release-images.sh](#)

在现有 release 发布脚本中添加了对 XPU 镜像的处理，包括拉取、tag、manifest 创建和推送。

```
# ---- XPU ----

# 从 ECR 拉取预构建的 XPU 镜像
docker pull public.ecr.aws/q9t5s3a7/vllm-release-repo:${COMMIT}-x86_64-xpu

# 标记为 latest 和 release 版本的 x86_64 架构标签
docker tag public.ecr.aws/q9t5s3a7/vllm-release-repo:${COMMIT}-x86_64-xpu vllm/vllm-openai-xpu:latest-x86_64
docker tag public.ecr.aws/q9t5s3a7/vllm-release-repo:${COMMIT}-x86_64-xpu vllm/vllm-openai-
```

```
xpu:v${RELEASE_VERSION}-x86_64
docker push vllm/vllm-openai-xpu:latest-x86_64
docker push vllm/vllm-openai-xpu:v${RELEASE_VERSION}-x86_64

# 创建并推送 multi-arch manifest (当前只有 x86_64, 未来可扩展 ARM)
docker manifest rm vllm/vllm-openai-xpu:latest || true
docker manifest rm vllm/vllm-openai-xpu:v${RELEASE_VERSION} || true
docker manifest create vllm/vllm-openai-xpu:latest vllm/vllm-openai-xpu:latest-x86_64 --amend
docker manifest create vllm/vllm-openai-xpu:v${RELEASE_VERSION} vllm/vllm-openai-xpu:
v${RELEASE_VERSION}-x86_64 --amend
docker manifest push vllm/vllm-openai-xpu:latest
docker manifest push vllm/vllm-openai-xpu:v${RELEASE_VERSION}
```

评论区精华

本 PR 的 review 由两位项目维护者 (jikunshang 和 khluu) 批准, 没有提出实质性修改建议。Claude 自动化检查因来自 fork 而未执行。整体流程比较简单, 无重大争议。

- 无实质性讨论 (other): 无需修改。

风险与影响

- 风险: 风险较低。主要风险包括: 1) XPU nightly 步骤依赖 create-manifest-xpu 步骤, 如果该步骤失败则 nightly 发布不会触发; 2) 脚本中硬编码了镜像仓库名 vllm/vllm-openai-xpu, 后续更名需手动同步; 3) 新增的 manifest 清理逻辑 (docker manifest rm ... || true) 可能导致极短时间内标签不可用。但由于发布流程本身已具备类似模式 (如 ROCm), 风险可控。
- 影响: 对用户: XPU 用户现在可以从 DockerHub 获取 nightly 和稳定版镜像。对团队: 新增一个脚本需要维护, 但逻辑与现有发布脚本类似。对系统: 在 release pipeline 中增加了一些步骤, 只影响 XPU 相关路径, 不增加其他构建负担。
- 风险标记: 新增 CI 步骤, 依赖上游 manifest 步骤, 脚本硬编码仓库名

关联脉络

- PR #47180 [CI] Add TORCH_NIGHTLY=1 build mode (run full suite on torch nightly): 本 PR 使用了相同的 nightly 构建机制 (NIGHTLY 环境变量), 并在此基础上扩展了 XPU nightly 步骤。