PR #27156 完整报告

sgl-project/sglang

[XPU CI] Expand stage-a and consolidate stage-b tests into stage-a

合并时间：2026-06-04 12:00

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27156

执行摘要

- 一句话：整合 XPU CI 测试至单个 stage 以消除重复构建
- 推荐动作：对于 CI 维护者，此整合策略值得关注，但需注意后续启用 stage-b 时务必实现 Docker 镜像缓存（如 registry push/pull），以避免重复构建。本次更改不涉及核心逻辑，可安全合并。

功能与动机

PR body 指出每个 stage 从零构建 Docker 镜像，导致 CI 壁钟时间严重膨胀。通过合并到一个 stage-a，消除了重复构建。一旦 Docker 拉取 / 推送从注册表到位，stage-b 将恢复。

实现拆解

1. 修改 .github/workflows/pr-test-xpu.yml：注释掉 stage-b 作业，将 finish 作业改为仅依赖 stage-a。
2. 在 6 个现有测试文件中，将 register_xpu_ci 的 suite 参数从 stage-b-test-1-gpu-xpu 改为 stage-a-test-1-gpu-xpu。涉及文件：test_chunk_gated_delta_rule.py、test_deepseek_ocr.py、test_deepseek_ocr_triton.py、test_intel_xpu_backend.py、test_topk.py。
3. 在 3 个新测试文件中添加 register_xpu_ci 调用，注册到 stage-a：test_sampling_params.py（66 测试）、test_lora_eviction_policy.py（12 测试）、test_adaptive_spec_params.py（11 测试）。

关键文件：

- .github/workflows/pr-test-xpu.yml（模块 CI 流水线；类别 infra；类型 infrastructure）：CI 流水线核心配置，注释掉 stage-b 并调整作业依赖，实现 stage 合并。
- test/registered/unit/sampling/test_sampling_params.py（模块 采样参数；类别 test；类型 test-coverage）：新增 XPU CI 注册，覆盖 66 个采样参数测试，是本次新加入的最大测试文件。
- test/registered/lora/test_lora_eviction_policy.py（模块 LoRA 驱逐策略；类别 test；类型 test-coverage）：新增 XPU CI 注册，覆盖 12 个 LoRA 驱逐策略测试，是 LoRA 模块首次加入 XPU CI。
- test/registered/unit/spec/test_adaptive_spec_params.py（模块 适应推测参数；类别 test；类型 test-coverage）：新增 XPU CI 注册，覆盖 11 个自适应推测参数测试。

- test/registered/attention/test_chunk_gated_delta_rule.py（模块 注意层 Delta 规则；类别 test；类型 test-coverage）：将 XPU 注册从 stage-b 迁移到 stage-a。
- test/registered/xpu/test_deepseek_ocr.py（模块 DeepSeek OCR；类别 test；类型 test-coverage）：将 XPU 注册从 stage-b 迁移到 stage-a。
- test/registered/xpu/test_deepseek_ocr_triton.py（模块 DeepSeek OCR Triton；类别 test；类型 test-coverage）：将 XPU 注册从 stage-b 迁移到 stage-a（该测试因 Triton-XPU 升级未完成而禁用）。
- test/registered/xpu/test_intel_xpu_backend.py（模块 XPU 后端；类别 test；类型 test-coverage）：将 XPU 注册从 stage-b 迁移到 stage-a。
- test/registered/xpu/test_topk.py（模块 TopK；类别 test；类型 test-coverage）：将 XPU 注册从 stage-b 迁移到 stage-a。

关键符号：未识别

关键源码片段

.github/workflows/pr-test-xpu.yml

CI 流水线核心配置，注释掉 stage-b 并调整作业依赖，实现 stage 合并。

```
# 将 finish 作业改为仅依赖 stage-a
finish:
  needs: [check-changes, pr-gate, stage-a-test-1-gpu-xpu]
  ...

# Stage B 被整体注释掉，等待 Docker 缓存机制就绪再恢复
# stage-b-test-1-gpu-xpu:
# needs: [check-changes, pr-gate, wait-for-stage-a]
# if: needs.check-changes.outputs.main_package == 'true'
# runs-on: intel-bmg
# steps:
# ...（原 Docker 构建 + 测试步骤全部注释）
```

test/registered/lora/test_lora_eviction_policy.py

新增 XPU CI 注册，覆盖 12 个 LoRA 驱逐策略测试，是 LoRA 模块首次加入 XPU CI。

```
# 导入列表中增加 register_xpu_ci
from sglang.test.ci.ci_register import (
    register_amd_ci,
    register_cpu_ci,
    register_cuda_ci,
    register_xpu_ci, # 新增
)
...
register_cpu_ci(est_time=6, suite="base-b-test-cpu")
register_xpu_ci(est_time=10, suite="stage-a-test-1-gpu-xpu") # 将 LoRA 测试加入 XPU stage-a
```

评论区精华

Gemini-code-assist 机器人在多个测试文件上建议使用结构化的 stage 和 runner_config 参数替代旧的 suite 参数，但维护者 arathi-hlab 回复指出“stage-name-test-number of gpu-gpu-xpu”是目前遵循的命名模式，未采纳该建议。另外，mingfeima 询问 stage-a 的超时是否需要更新（因为预期 37 分钟），arathi-hlab 回应实际 CI 运行仅 14 分钟，本地估算不准确。

- 使用结构化 stage/runner_config 参数替代 suite 参数 (style): 维护者 arathi-hlab 回复解释当前命名模式是 'stage-name-test-number of gpu-gpu-xpu'，未采纳建议。mingfeima 也指出机器人评论正确，但最终决定保持现有风格。
- stage-a 超时是否需要更新 (question): arathi-hlab 回应实际 CI 运行仅 14 分钟，本地估时不准确，无需更新超时。

风险与影响

- 风险：主要风险是将所有测试集中到 stage-a，一旦该 stage 失败则无备用 stage-b 兜底。当前注释掉 stage-b 但未删除，后续需 Docker 缓存机制才能安全恢复两阶段流程。此外，如果 stage-a 运行时间过长，可能影响 CI 整体效率，但实际运行 14 分钟在可接受范围内。
- 影响：对 XPU CI 开发者而言：测试现在都在 stage-a 运行，消除了重复 Docker 构建，总体壁钟时间缩短。新增的 3 个测试文件（采样参数、自适应推测参数、LoRA 驱逐策略）纳入 XPU CI 覆盖，提高了对 Intel XPU 平台的质量保障。stage-b 被注释但保留，为未来恢复两阶段流水线提供基础。
- 风险标记：单点故障：若 stage-a 失败则无备份，注释的 stage-b 代码需要后续清理或恢复

关联脉络

- 暂无明显关联 PR