

PR #35238 完整报告

sgl-project/sglang

Exclude multimodal-gen NPU jobs from fast-fail cascade

合并时间: 2026-08-18 23:48

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/35238>

执行摘要

本 PR 是 NPU CI 流水线的防御性修正: 在 `_npu-single-node-test-stage.yml` 与 `_npu-pr-test-stage.yml` 的 `Check PR test health` 过滤逻辑中, 将 `multimodal-gen-test-1-npu-a3` / `multimodal-gen-test-2-npu-a3` 两个 job 排除出 `root cause` 失败判定, 使其失败不再级联导致 `base-c-test-perf-*-npu-a3` / `base-c-test-acc-*-npu-a3` 快速失败。改动仅涉及 GitHub Actions 工作流, 不影响任何运行时逻辑, 属于低风险、高性价比的 CI 稳定性改进。

功能与动机

NPU PR 测试工作流 `pr-test-npu.yml` 并行运行 `multimodal-gen` 测试与 `perf/accuracy` 测试, 后者通过可复用工作流 `_npu-single-node-test-stage.yml` 实现。PR body 明确指出: 该工作流内联的 `Check PR test health` 步骤会在同一 run 中其他 job 已因 `root cause` 失败时让当前 job 快速失败 (`fast-fail`)。由于 `multimodal-gen` 测试与 `perf/accuracy` 无依赖关系, 此前 `multimodal-gen` 的偶发失败会污染 `perf/accuracy` 的 CI 信号, 因此需要从过滤器中剔除这两个 job。

实现拆解

1. 定位过滤入口: `_npu-single-node-test-stage.yml` 与 `_npu-pr-test-stage.yml` 中内联的 `Check PR test health` 步骤是 `fast-fail` 级联的核心实现, 它遍历当前 run 内其他 job 的失败状态, 判断是否为 `root cause` 失败。
2. 新增过滤分支: 在过滤函数中, 于既有 `h20 job` 过滤逻辑之后新增分支——当 job 基名匹配 `multimodal-gen-test-1-npu-a3` 或 `multimodal-gen-test-2-npu-a3` 时, 打印 `[health-check] Filtered out multimodal-gen NPU job` 日志并返回 `false`, 将该 job 排除出 `root cause` 判定。
3. 同步双份工作流: 由于两个工作流各维护一份相同逻辑, 两处均加上完全一致的 5 行分支, 保证行为一致, 避免同一 run 内不同入口出现差异。
4. 配套注释修正: `pr-test-npu.yml` 中 `gsm8k` 数据集拷贝步骤的注释由 `# copy gsm8k dataset` 改为 `# copy gsm8k dataset from cache`, 明确数据来自 `~/.cache/modelscope/hub/datasets` 缓存目录, 无行为变化。
5. 验证方式: 纯 CI 配置变更无单元测试配套; 合并前由 `sglang-npu-bot` 通过 `/tag-and-rerun-ci` 触发 CI 复跑, 最终 PR Test 通过、PR Test (Extra) 失败后接入 `main`。

[.github/workflows/_npu-single-node-test-stage.yml](#)

可复用 workflow 中的 **Check PR test health** 步骤是 fast-fail 级联的核心实现，本 PR 在此新增 multimodal-gen NPU job 过滤分支，是行为变更的主要落点。

```
// Check PR test health 步骤的 root cause 过滤核心逻辑（重构示意）
// 内嵌于 NPU 测试 workflow，负责从当前 run 的失败 job 中剔除
// 与当前阶段无依赖关系的失败，避免 fast-fail 级联误伤

function shouldFilterJob(j) {
  const baseName = j.name.split('/')[0];

  // 既有规则：h20 相关 job 失败属于已知环境问题，不当作 root cause
  if (baseName.includes('h20')) {
    core.info(`[health-check] Filtered out h20 job: ${j.name}`);
    return true;
  }

  // 新增规则：multimodal-gen NPU 测试与 perf/accuracy 并行且互不依赖，
  // 其失败不应 cascading 到 `base-c-test-perf-*-npu-a3` / `base-c-test-acc-*-npu-a3` 阶段
  if (baseName === 'multimodal-gen-test-1-npu-a3' ||
      baseName === 'multimodal-gen-test-2-npu-a3') {
    core.info(`[health-check] Filtered out multimodal-gen NPU job: ${j.name}`);
    return true;
  }

  // 失败步骤本身是 health check 时，说明是级联失败，跳过
  const failedStep = (j.steps || []).find(s => s.conclusion === 'failure');
  if (failedStep &&
      (failedStep.name.includes('check-pr-test-health') ||
       failedStep.name.includes('Check PR test health'))) {
    return true;
  }

  return false;
}
```

评论区精华

cherryblo: Modify the comment content.

这是本 PR 唯一的 review 评论，针对新增注释的措辞提出风格建议，未涉及技术设计争议。结合最终合并版本保留了过滤逻辑与注释，可推断作者按建议调整了注释表述。

风险与影响

- job 名称硬编码：过滤条件直接匹配 multimodal-gen-test-1-npu-a3 / multimodal-gen-test-2-npu-a3 字符串，若未来重命名或新增 job，过滤会静默失效。
- 逻辑双份维护：两个 workflow 各有一份相同过滤代码，后续调整需同步修改，存在漂移风险。
- 保护范围收窄：multimodal-gen 的真实 root cause 失败不再触发 perf/accuracy 快速失败，需要人工关注，但换来更稳定的 perf/accuracy 信号。

- 影响面限定在 NPU PR 测试流水线，对最终用户无行为影响。

关联脉络

本 PR 是 NPU CI 稳定性治理的一部分。近期 diffusion / multimodal-gen 功能迭代频繁（如 Cosmos3 rollout、per-section LoRA、encoder 并行解耦、safetensors 分片过滤修复），这些改动持续触碰 `multimodal-gen-test-*-npu-a3` 测试面，失败偶发；本 PR 通过 health-check 白名单机制将 multimodal-gen 失败与 perf/accuracy 结果解耦，避免噪声污染核心性能信号。后续若扩展过滤清单，建议收敛为单一常量配置，减少双份维护成本。