

PR #6521 完整报告

verl-project/verl

[ci] chore: add npu sglang nightly ci

合并时间: 2026-05-29 11:57

原文链接: <http://prhub.com.cn/verl-project/verl/pull/6521>

执行摘要

- 一句话: 新增 NPU SGLang 夜间 CI 测试
- 推荐动作: 该 PR 主要涉及 CI 配置和测试脚本, 核心逻辑在 verl 框架内未变化。建议 nightly CI 维护者关注 review 中提出的两个高优先级问题并考虑后续修复。对于其他开发者, 可以了解 nightly 测试的编写模式, 但不必精读。

功能与动机

为了保障 Megatron + SGLang 在 Ascend NPU 上的功能稳定性, 需要在 nightly CI 中增加自动化测试覆盖。此前 NPU 上只有 vLLM 相关的 nightly 测试, 缺少 SGLang 的 nightly 验证。

实现拆解

1. 新增测试脚本 (`tests/special_npu/nightly_ci_ascend/run_grpo_qwen3_30b_megatron_sglang_npu.sh`): 这是一个 244 行的 bash 脚本, 用于配置和运行 GRPO 训练。它包含了环境变量设置、模型和路径配置、并行度参数、训练超参等, 最后调用 verl 训练命令。脚本开头还包含对 SGLang 临时修复 (注释掉某些行, transpose 权重) 以规避上游问题。
2. 更新 CI 工作流 (`.github/workflows/nightly_ascend.yml`): 在已有的 nightly 工作流中新增一个 job `nightlyCI_grpo-qwen3-30b-megatron-sglang_ascend`。该 job 在 `linux-aarch64-a3-16` runner 上运行, 使用特定容器镜像 (`verl-sglang-8.5.0-a3-...`), 设置环境变量, 执行 checkout、准备权重, 最后运行测试脚本。超时设为 180 分钟。

关键文件:

- `tests/special_npu/nightly_ci_ascend/run_grpo_qwen3_30b_megatron_sglang_npu.sh` (模块 测试脚本; 类别 test; 类型 test-coverage): 核心测试脚本, 定义了 GRPO 训练的完整配置和执行流程
- `.github/workflows/nightly_ascend.yml` (模块 CI 配置; 类别 infra; 类型 infrastructure): CI 工作流定义, 新增 job 来触发测试脚本

关键符号: 未识别

关键源码片段

tests/special_npu/nightly_ci_ascend/run_grpo_qwen3_30b_megatron_sglan_g_npu.sh

核心测试脚本，定义了 GRPO 训练的完整配置和执行流程

```
#!/bin/bash
set -x

# 并行度配置
train_tp=4
train_ep=4
train_etp=4
train_pp=1
train_cp=1
gen_tp=4
gen_dp=1
gen_ep=1

# 数据配置数组
DATA_CONFIG=(
    data.train_files="${TRAIN_FILE}"
    data.val_files="${TEST_FILE}"
    data.prompt_key=prompt
    data.train_batch_size=${train_prompt_bsz}
    data.max_prompt_length=${max_prompt_length}
    data.max_response_length=${max_response_length}
    data.filter_overlong_prompts=False
    data.truncation='left'
)

# 模型配置数组
MODEL_CONFIG=(
    actor_rollout_ref.model.path="${MODEL_PATH}"
    actor_rollout_ref.model.use_remove_padding=True
)

# 算法配置数组
ALGORITHM_CONFIG=(
    algorithm.adv_estimator=${adv_estimator}
    algorithm.use_kl_in_reward=${use_kl_in_reward}
    algorithm.kl_ctrl.kl_coef=${kl_coef}
)

# 最终将配置数组传递给训练命令
```

评论区精华

Review 中 gemini-code-assist[bot] 指出了两个高优先级问题：

- 脚本使用 `set -x` 而非 `set -xeuo pipefail`，可能导致训练命令通过管道 `tee` 后退出码被吞没，CI 误报成功。
- 配置项 `actor_rollout_ref.actor.profiler.tool_config.npu.contents=['npu','cpu']` 中的方括号未加引号，可能被 `bash glob` 展开，导致参数错误。最终 reviewer wucong25 批准了 PR，上述问题未在合并前修复，可能存在风险。
- 使用 `set -xeuo pipefail` 确保 CI 正确检测失败 (correctness): 未在 PR 中直接修正，但 reviewer 仍批准合并。
- 方括号未加引号可能导致 `glob` 扩展 (correctness): 未修正即合并。

风险与影响

- 风险：
 1. CI 假阳性风险：如 review 指出的，脚本未使用 `set -euo pipefail`，若训练失败可能因为管道 `tee` 导致退出码为 0，CI 显示成功但实际失败。
 2. 环境依赖风险：测试依赖特定的容器镜像、模型权重路径和数据集文件，若镜像更新或路径变化可能导致测试失败。
 3. 超时风险：GRPO 训练耗时较长，超时 180 分钟可能不足，需根据实际运行调整。
 4. 配置参数硬编码：测试脚本中许多参数（如 TP/EP、offload 等）硬编码，若未来默认配置变化可能导致失效。
 - 影响：对用户：无直接影响，但夜间测试可提前发现 NPU SGLang 集成问题，提升发布质量。对系统：新增一个 CI job，在特定 runner 上运行，增加 CI 资源消耗。对团队：需要维护测试脚本和 CI 配置，确保镜像、路径等与基础设施同步。
- 风险标记：CI 假阳性风险，管道错误掩盖，硬编码配置，环境依赖风险

关联脉络

- PR #6374 [megatron] feat: ascend bump into megatron 016: 该 PR 升级了 Megatron 以支持 Ascend，为本 PR 的 nightly 测试提供了基础版本依赖。
- PR #6435 [docker] chore: upgrade slang to 0.5.12: 该 PR 升级了 SGLang 版本，并且本夜测使用 SGLang，版本兼容依赖该升级。