PR #26609 完整报告

sgl-project/sglang

[CI] Clean DeepSeek V4 tests and installation scripts

合并时间：2026-05-29 06:22

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26609

执行摘要

- 一句话：清理 DeepSeek V4 测试和安装脚本
- 推荐动作：对于关注 CI 基础设施组织的读者值得精读，展示了如何通过删除冗余和统一命名来降低维护成本。

功能与动机

通过清理移除冗余安装脚本并统一测试注册，简化 DeepSeek V4 这类复杂模型的 CI 维护工作，便于后续扩展。

实现拆解

1. 删除旧安装脚本：移除 scripts/ci/cuda/ci_install_dsv4_dep.sh，其功能被合并到更通用的 ci_install_deepep.sh 中，减少重复维护。
2. 加强 DeepEP 安装脚本：在 ci_install_deepep.sh 中添加基于 CUDA 版本的计算能力选择逻辑，动态支持 Hopper (sm_90)、Blackwell (sm_100) 和未来 Blackwell (sm_103)，避免硬编码。
3. 统一测试注册键：将所有 DeepSeek V4 测试文件（如 test_deepseek_v4_flash_fp4_b200.py、test_deepseek_v4_flash_fp8_h200.py 等）中的注册键从 dsv4-* 改为 deepep-*，并将部分测试从 base-c 阶段移至 extra-b 阶段，降低常用 CI pipeline 负担。
4. 更新 CI 工作流：修改 .github/workflows/pr-test.yml 和 pr-test-extra.yml，添加新的 job 和 runner 配置，确保新注册的测试能被正确调度。
5. 调整 runner 配置与测试套件映射：更新 scripts/ci/runner_configs.yml 和 test/run_suite.py，使其与新测试阶段和 runner 名称保持一致。

关键文件：

- scripts/ci/cuda/ci_install_dsv4_dep.sh（模块 部署脚本；类别 infra；类型 deletion）：删除了整个旧版安装脚本，将功能合并到 ci_install_deepep.sh，是本次清理的核心
- scripts/ci/cuda/ci_install_deepep.sh（模块 部署脚本；类别 infra；类型 infrastructure）：添加了多架构构建支持（Hopper/Blackwell），是 CI 基础设施的核心改进
- .github/workflows/pr-test.yml（模块 CI 工作流；类别 infra；类型 infrastructure）：更新了 DeepSeek V4 测试的 runner 配置，更新 CI 流程

- .github/workflows/pr-test-extra.yml（模块 CI 工作流；类别 infra；类型 infrastructure）：添加了 extra-b 阶段的测试 job，支持额外 DeepSeek V4 测试
- test/registered/models_e2e/test_deepseek_v4_flash_fp4_b200.py（模块 测试套件；类别 test；类型 test-coverage）：作为 DeepSeek V4 测试的样本，展示了注册键从 dsv4 转为 deepep
- test/registered/models_e2e/test_deepseek_v4_flash_fp8_h200.py（模块 测试套件；类别 test；类型 test-coverage）：该测试从 base-c 阶段移到 extra-b 阶段，是阶段调整的实例
- scripts/ci/runner_configs.yml（模块 CI 配置；类别 infra；类型 infrastructure）：更新 runner 配置以反映新的 deepep 键，是 CI 配置的关键文件
- test/run_suite.py（模块 测试运行器；类别 test；类型 test-coverage）：添加了新的测试套件映射，支持 extra-b 阶段

关键符号：未识别

关键源码片段

scripts/ci/cuda/ci_install_deepep.sh

添加了多架构构建支持（Hopper/Blackwell），是 CI 基础设施的核心改进

```
# 构建 DeepEP 时根据 CUDA 版本动态选择计算能力
if [ -n "${NVCC_VER:-}" ]; then
  CUDA_VERSION="$NVCC_VER"
elif command -v nvcc >/dev/null 2>&1; then
  CUDA_VERSION=$(nvcc --version | grep -oP 'release \K[0-9]+\.[0-9]+')
else
  CUDA_VERSION=$(nvidia-smi | grep "CUDA Version" | head -n1 | awk '{print $9}' || true)
fi
if [ -z "${CUDA_VERSION:-}" ]; then
  echo "FATAL: 无法确定 CUDA 工具包版本"
  exit 1
fi
# 根据 CUDA 版本设置架构列表：sm_90=Hopper, sm_100=Blackwell, sm_103= 未来 Blackwell
if [ "$CUDA_VERSION" = "12.8" ]; then
  CHOSEN_TORCH_CUDA_ARCH_LIST='9.0;10.0'
elif awk -v ver="$CUDA_VERSION" 'BEGIN {exit !(ver > 12.8)}'; then
  CHOSEN_TORCH_CUDA_ARCH_LIST='9.0;10.0;10.3'
else
  CHOSEN_TORCH_CUDA_ARCH_LIST='9.0'
fi
TORCH_CUDA_ARCH_LIST="${CHOSEN_TORCH_CUDA_ARCH_LIST}" python3 setup.py install
```

评论区精华

该 PR 没有引起公开评论，由作者直接合并。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险在于 DeepEP 安装脚本的变更可能影响 a2a 后端的正确性，但已在 CI 中对相关测试做了 rerun 验证（见 Issue 评论）。测试阶段从 base-c 移至 extra-b 可能导致某些问题在基础 CI 中未被及时捕获，但额外 CI 仍覆盖了这些场景。整体风险较低。
- 影响：影响范围限于 CI 系统和 DeepSeek V4 测试维护者。对普通用户无直接影响；对 CI 维护者而言，配置更加简洁清晰，runner 名称统一为 deepep 前缀，降低了认知负担。
- 风险标记：测试阶段调整，安装脚本重构，CI 配置变更

关联脉络

- 暂无明显关联 PR