

PR #29693 完整报告

sgl-project/sglang

[AMD] Register ltx2_ada_values JIT kernel test for AMD nightly CI

合并时间: 2026-07-01 16:13

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/29693>

执行摘要

- 一句话: 为 AMD 注册 ltx2_ada_values JIT 内核测试
- 推荐动作: 可直接批准。此 PR 展示了如何为已有 NVIDIA 测试添加 AMD 覆盖, 可作为后续类似操作的范例。

功能与动机

缩小 AMD 与 NVIDIA 在 JIT 内核测试覆盖上的差距, 确保 Triton 编写的 diffusion 相关内核在 ROCm 上同样经过验证。PR body 中明确说明该内核是纯 Triton 实现, 不依赖 CUDA C++ 或 flashinfer/sgl_kernel, 因此可移植。

实现拆解

1. 修改导入: 在 test/registered/jit/diffusion/test_ltx2_ada_values.py 中将 from sglang.test.ci.ci_register import register_cuda_ci 改为同时导入 register_amd_ci。
2. 添加注册: 在 register_cuda_ci(...) 之后新增一行 register_amd_ci(est_time=8, suite="nightly-amd-kernel-1-gpu", nightly=True), 将测试加入 AMD 夜间 kernel 套件。
3. 无需 workflow 变更: 因为 nightly-amd-kernel-1-gpu 套件已在 nightly-test-amd.yml 和 nightly-test-amd-rocm720.yml 中定义, 所以只需注册即可。
4. 解决冲突: 第二个 commit 合入上游 main 分支, 保留了原有的 register_cuda_ci 格式并添加了新的注册。

关键文件:

- test/registered/jit/diffusion/test_ltx2_ada_values.py (模块 测试注册; 类别 test; 类型 test-coverage) : 唯一变更文件, 为 ltx2_ada_values 测试添加 AMD 夜间 CI 注册

关键符号: 未识别

关键源码片段

[test/registered/jit/diffusion/test_ltx2_ada_values.py](#)

唯一变更文件, 为 ltx2_ada_values 测试添加 AMD 夜间 CI 注册

```
import sys
```

```
import pytest
```

```

import torch

from sglang.jit_kernel.diffusion.triton.ltx2_ada_values import ltx2_ada_values9
# 同时导入 register_amd_ci 以支持 AMD 夜间 CI
from sglang.test.ci.ci_register import register_amd_ci, register_cuda_ci

# NVIDIA 注册保持不变
register_cuda_ci(est_time=8, stage="base-b-kernel-unit", runner_config="1-gpu-large")
# 新增 AMD 注册, 加入 nightly-amd-kernel-1-gpu 套件
register_amd_ci(est_time=8, suite="nightly-amd-kernel-1-gpu", nightly=True)

DEVICE = "cuda"

@pytest.fixture(autouse=True)
def cuda_setup():
    if not torch.cuda.is_available():
        pytest.skip("CUDA required")
    torch.cuda.manual_seed(0)

def _reference(scale_shift_table: torch.Tensor, timestep: torch.Tensor) -> tuple[torch.Tensor, ...]:
    batch, seq, _ = timestep.shape
    hidden = scale_shift_table.shape[1]
    return (
        scale_shift_table.to(device=timestep.device, dtype=timestep.dtype)
        .view(1, 1, 9, hidden)
        .add(timestep.reshape(batch, seq, 9, hidden))
        .unbind(dim=2)
    )

@torch.no_grad()
@pytest.mark.parametrize("batch,seq,hidden", [(1, 1, 4096), (2, 3, 2048)])
@pytest.mark.parametrize("table_dtype", [torch.bfloat16, torch.float32])
def test_ltx2_ada_values9(batch: int, seq: int, hidden: int, table_dtype: torch.dtype):
    scale_shift_table = torch.randn(1, hidden, 9, dtype=table_dtype, device=DEVICE)
    timestep = torch.randn(batch, seq, 9 * hidden, dtype=torch.float32, device=DEVICE)
    ref = _reference(scale_shift_table, timestep)
    out = ltx2_ada_values9(scale_shift_table, timestep, batch, seq, hidden)
    for r, o in zip(ref, out):
        torch.testing.assert_close(o, r, atol=0, rtol=0)

```

评论区精华

讨论集中在合并冲突上：由于上游 main 分支已更新了 `register_cuda_ci` 的参数形式（新增 `stage` 和 `runner_config`），导致冲突。作者 michaelzhang-ai 手动解决冲突，保留上游新格式并添加 AMD 注册，HaiShaw 在解决冲突后重新批准。

- 暂无高价值评论线程

风险与影响

- 风险：无技术风险。变更仅为测试注册，不影响任何生产逻辑。由于测试本身已在 ROCm 上验证通过（PR body 附有 CI 结果），不会引入回归。
- 影响：对 AMD 平台用户：ltx2_ada_values 内核将在夜间 CI 中得到验证，确保 Triton 内核在 ROCm 上的正确性。对 NVIDIA 平台无影响。
- 风险标记：暂无

关联脉络

- PR #29789 chore: clean diffusion dead code: 同为 diffusion 模块相关的清理或测试增强
- PR #29824 [diffusion] CI: tighten multimodal-gen consistency thresholds: 同为 diffusion 相关的 CI 改进