PR #32649 完整报告

sgl-project/sglang

add NPU GSM8K accuracy tests for 7 models

合并时间：2026-08-01 14:22

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/32649

执行摘要

- 一句话：NPU 平台新增 6 模型 GSM8K 精度回归测试与 CI 注册
- 推荐动作：值得关注的是 NPU CI 注册机制（register_npu_ci 与 workflow matrix 的双向注册）和精度测试基类 TestNpuAccuracyTestCaseBase 的模板化用法；对于需要在 NPU 上新增模型测试的工程师，本 PR 提供了可直接复用的范式。代码本身逻辑简单，无需精读推理路径。

功能与动机

PR body 明确提出目标：Add NPU GSM8K accuracy test cases under test/registered/ascend/accuracy/，覆盖 glm4_7_flash、qwen3_vl_30b_a3b、glm5_top64_pruned、moonshotai_moonlight_16b_a3b、qwen3_5_9b、qwen3_vl_8b 等模型。动机是补全 NPU 侧精度回归覆盖：ascend 注意力后端、采样后端等 NPU 特有实现与 CUDA 路径差异较大，缺少 GSM8K 基线会导致算子改动后精度回退难以被及时发现，需要为 NPU 推理栈建立可自动触发的精度基线。

实现拆解

1. 模型路径常量（python/sglang/test/ascend/e2e/test_npu_performance_utils.py）：新增 QWEN3_5_9B_MODEL_PATH、MOONLIGHT_16B_A3B_MODEL_PATH、GLM5_TOP64_PRUNED_GSM8K_MODEL_PATH 三个常量，统一采用 modelscope 本地缓存路径格式，与既有 GLM_4_7_FLASH_MODEL_PATH 等常量管理方式一致。
2. 新增精度测试用例（test/registered/ascend/accuracy//）：6 个测试文件均继承 TestNpuAccuracyTestCaseBase，通过类属性声明 model、envs、other_args、accuracy 阈值、datasets、few_shot_num、generation_config、eval_batch_size，test_gsm8k 调用 run_accuracy() 执行完整评估流程；再用 register_npu_ci(est_time=3600, nightly=True, disabled="accuracy testcase") 注册进夜间 CI 套件。多模态模型（Qwen3-VL-8B、Qwen3-VL-30B-A3B）额外配置 --enable-multimodal 与 ascend_attn 后端；GLM5-top64-pruned 使用 --tp-size 16 与 deepep a2a 后端，是规模最大的用例。
3. CI 工作流注册（.github/workflows/pr-test-npu.yml）：在测试矩阵中新增 6 个 test_type: 'accuracy' 条目，按模型规模分配 linux-aarch64-a3-2（双卡）、linux-aarch64-a3-4（四卡）、linux-aarch64-a3-16（十六卡）runner。

4. 修正与精简：提交历史显示修复了基类导入名（TestAscendAccuracyTestCaseBase → TestNpuAccuracyTestCaseBase），更新精度阈值基线；按 review 意见删除 register_npu_ci 中重复的 full 套件注册，并移除 qwen3_vl_30b_a3b_thinking 工作流条目（该用例最终未合入）。qwen3_5_9b 与 glm5_top64_pruned 通过 limit = 100 限制评估样本数以控制 CI 时长。

关键文件：

- test/registered/ascend/accuracy/qwen3_vl_8b/test_npu_qwen3_vl_8b_bf16_2p_gsm8k.py（模块 精度测试；类别 test；类型 test-coverage；符号 TestNPUQwen3_VL_8B_GSM8K, test_gsm8k）：最完整的模板化精度测试用例：配置多模态、ascend 注意力后端、tp-size 4 等参数，并注册到 stage-b 夜间套件，是其他 5 个用例的参照。
- test/registered/ascend/accuracy/glm4_7_flash/test_npu_glm4_7_flash_1p_gsm8k.py（模块 精度测试；类别 test；类型 test-coverage；符号 TestNPUGlm4_7Flash_1P_GSM8K, test_gsm8k）：GLM4.7-Flash 的 GSM8K 精度测试，配置 tp-size 2、chunked prefill 16384 与 glm47 工具调用解析器，覆盖 Dense 模型在 NPU 上的精度基线。
- test/registered/ascend/accuracy/qwen3_5_9b/test_npu_qwen3_5_9b_bf16_1p_gsm8k.py（模块 精度测试；类别 test；类型 test-coverage；符号 TestNPUQwen3_5_9B_GSM8K, test_gsm8k）：Qwen3.5-9B 的 GSM8K 精度测试，开启 dp-attention，并通过 limit = 100 限制评估样本数控制时长，展示大规模模型的降采样策略。
- test/registered/ascend/accuracy/qwen3_vl_30b_a3b/test_npu_qwen3_vl_30b_a3b_bf16_2p_gsm8k.py（模块 精度测试；类别 test；类型 test-coverage；符号 TestNPUQwen3_VL_30B_A3B_GSM8K, test_gsm8k）：Qwen3-VL-30B-A3B 多模态 MoE 模型的 GSM8K 精度测试，tp-size 4、max-prefill-tokens 102400，覆盖多模态大模型配置。
- test/registered/ascend/accuracy/moonshotai_moonlight_16b_a3b/test_npu_moonlight_16b_a3b_bf16_1p_gsm8k.py（模块 精度测试；类别 test；类型 test-coverage；符号 TestNPUMoonlight16B_A3B_GSM8K, test_gsm8k）：Moonlight-16B-A3B 的 GSM8K 精度测试，tp-size 2，为月之暗面 MoE 模型在 NPU 上建立精度基线。
- test/registered/ascend/accuracy/glm5_top64_pruned/test_npu_glm5_top64_pruned_bf16_8p_gsm8k.py（模块 精度测试；类别 test；类型 test-coverage；符号 TestNPUGLM5_Top64_Pruned_GSM8K, test_gsm8k）：GLM5-top64-pruned 的 GSM8K 精度测试，tp-size 16 并使用 deepep a2a 后端，是本次矩阵中规模最大、资源占用最高的用例。
- python/sglang/test/ascend/e2e/test_npu_performance_utils.py（模块 测试工具；类别 test；类型 test-coverage）：集中管理 NPU 测试模型路径，本次新增 QWEN3_5_9B_MODEL_PATH、MOONLIGHT_16B_A3B_MODEL_PATH、GLM5_TOP64_PRUNED_GSM8K_MODEL_PATH，是测试用例引用的依赖入口。
- .github/workflows/pr-test-npu.yml（模块 CI 配置；类别 infra；类型 infrastructure）：NPU PR 测试矩阵入口，新增 6 个 accuracy 测试条目并按模型规模分配 runner，是 CI

实际执行的依据。

关键符号：test_gsm8k, run_accuracy, register_npu_ci

关键源码片段

test/registered/ascend/accuracy/qwen3_vl_8b/test_npu_qwen3_vl_8b_bf16_2p_gsm8k.py

最完整的模板化精度测试用例：配置多模态、ascend 注意力后端、tp-size 4 等参数，并注册到 stage-b 夜间套件，是其他 5 个用例的参照。

```
import unittest

# 复用 NPU 精度测试基类，统一负责服务启动、数据集评估与阈值比对
from sglang.test.ascend.e2e.test_npu_accuracy_utils import TestNpuAccuracyTestCaseBase
# 模型路径常量集中在 performance_utils 中，供多套测试复用
from sglang.test.ascend.e2e.test_npu_performance_utils import QWEN3_VL_8B_MODEL_PATH
from sglang.test.ci.ci_register import register_npu_ci

# 注册进夜间 CI 的 stage-b 套件，预计耗时 1 小时
register_npu_ci(
    est_time=3600,
    suite="stage-b-test-4-npu-a3",
    nightly=True,
    disabled="accuracy testcase",
)

# 环境变量：开启 CPU 亲和与 expandable segments，稳定 NPU 显存分配
QWEN3_VL_8B_ENVS = {
    "SGLANG_SET_CPU_AFFINITY": "1",
    "PYTORCH_NPU_ALLOC_CONF": "expandable_segments:True",
    "STREAMS_PER_DEVICE": "32",
    "HCCL_BUFFSIZE": "1536",
    "HCCL_OP_EXPANSION_MODE": "AIV",
}

# 启动参数：多模态 + ascend 注意力后端，tp-size 4，bf16，关闭 radix cache
QWEN3_VL_8B_OTHER_ARGS = [
    "--enable-multimodal",
    "--mm-attention-backend", "ascend_attn",
    "--attention-backend", "ascend",
    "--device", "npu",
    "--trust-remote-code",
    "--tp-size", 4,
    "--mem-fraction-static", 0.8,
    "--disable-radix-cache",
    "--chunked-prefill-size", -1,
    "--sampling-backend", "ascend",
    "--tool-call-parser", "qwen",
```

```
    "--reasoning-parser", "qwen3",
    "--cuda-graph-bs",
    8, 16, 32, 64, 128, 256,
    "--dtype", "bfloat16",
]


class TestNPUQwen3_VL_8B_GSM8K(TestNpuAccuracyTestCaseBase):
    """Qwen3-VL-8B 在 NPU 上的 GSM8K 精度回归测试。"""

    model = QWEN3_VL_8B_MODEL_PATH
    envs = QWEN3_VL_8B_ENVS
    other_args = QWEN3_VL_8B_OTHER_ARGS
    # 精度阈值来自基线数据，低于该值即判定回归
    accuracy = 0.9553
    datasets = ["gsm8k"]
    few_shot_num = 5
    # 关闭 thinking 的生成配置，与 GSM8K 评测约定对齐
    generation_config = {
        "max_tokens": 32768,
        "temperature": 1.0,
        "top_p": 1.0,
        "top_k": 40,
        "repetition_penalty": 1.0,
        "presence_penalty": 2.0,
        "extra_body": {"chat_template_kwargs": {"enable_thinking": False}},
    }
    eval_batch_size = 64

    def test_gsm8k(self):
        self.run_accuracy()


if __name__ == "__main__":
    unittest.main()
```

评论区精华

review 全程由 cherryblo 给出，讨论非常简短：在 qwen3_vl_8b、qwen3_5_9b 测试文件的 register_npu_ci 块上分别评论 "delete"，指向 full-4-npu-a3、full-2-npu-a3 的重复注册；在 pr-test-npu.yml 中同样对 qwen3_vl_30b_a3b_thinking_1p_gsm8k 条目评论 "delete"。核心诉求是避免同一用例在 full 套件与 stage-b 套件中重复执行，并移除最终未落地的 thinking 变体条目。提交 cleanup: delete gsm8k test, keep only stage-b registrations for dual-reg files 落实了清理。

- 删除测试文件内重复的 full 套件注册 (design): 提交 cleanup: delete gsm8k test, keep only stage-b registrations for dual-reg files 已删除重复注册，仅保留 stage-b 条目。

- pr-test-npu.yml 中移除 qwen3_vl_30b_a3b_thinking 条目 (design): 已删除该条目，最终矩阵不含 thinking 变体用例。

风险与影响

- 风险：精度阈值硬编码：accuracy 字段（如 0.9553、0.8370）是基线快照，模型权重更新、采样后端或 ascend kernel 实现变化都可能触发阈值误报。模型路径硬编码为 /root/.cache/modelscope/...，依赖 CI runner 镜像预置权重缓存，缓存缺失会导致用例直接失败。CI 资源开销：6 个用例各 est_time=3600（约 1 小时），其中 glm5_top64_pruned 占用 16 卡 runner，矩阵扩展明显增加 NPU CI 资源消耗与排队时间。覆盖局限：qwen3_5_9b、glm5_top64_pruned 通过 limit = 100 仅评估 100 条样本，统计功效有限；qwen3_vl_30b_a3b_thinking 未合入意味着带 thinking 推理路径暂无 GSM8K 基线。合入时 PR Test 状态为失败（:x:），需确认是否为资源或基线问题。
- 影响：对线上服务无运行时影响。影响集中在 NPU CI 维护者（矩阵资源、模型权重准备）与后续 NPU 算子 / 后端改动者（必须保持 6 条 GSM8K 精度基线）。对团队而言，NPU 平台首次建立覆盖多种规模（双卡到十六卡）与多种架构（Dense、MoE、多模态）的精度回归入口，后续 NPU 内核优化可用其作为回归闸门。
- 风险标记：CI 资源开销增加，精度阈值硬编码，模型路径硬编码，样本数受限

关联脉络

- 暂无明显关联 PR