PR #19213 完整报告

sgl-project/sglang

[diffusion] Add cache-dit CI tests

合并时间：2026-05-10 13:38

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/19213

执行摘要

- 一句话：新增 cache-dit 的 1GPU 和 2GPU CI 测试用例
- 推荐动作：值得合并，建议后续缓存相关变更应确保该测试通过。可考虑进一步增加更多模型或配置的测试以覆盖更广场景。

功能与动机

PR #16662 引入了 cache-dit 功能，但缺乏 CI 测试覆盖。为防止后续修改导致回归，并捕获缓存并行配置构造等潜在问题（如 #19955、#19965），需要添加专用的 CI 测试用例。

实现拆解

1. 在 gpu_cases.py 头部添加 from dataclasses import replace 和 from pathlib import Path，并定义 _CACHE_DIT_CONFIG_DIR 常量指向 configs 目录，便于后续引用配置文件。
2. 在 ONE_GPU_CASES 列表中添加 qwen_image_t2i_cache_dit_scm_config_diffusers_1gpu 测试用例：使用 diffusers 后端，通过 --cache-dit-config 指定自定义 SCM 配置，并设置较小的输出尺寸和推理步数，禁用 run_perf_check 等所有非必要检查以保持轻量。
3. 在 TWO_GPU_CASES 列表中添加 wan2_1_t2v_1_3b_cache_dit_sp_only_2gpu 测试用例：仅启用 sequence parallelism（ulysses_degree=2），不启用 tensor parallelism，设置 enable_cache_dit=True 及环境变量 SGLANG_CACHE_DIT_WARMUP=2，同样禁用非必要检查。
4. 新增 configs/cache_dit_scm_config.yaml，定义缓存策略参数（如 max_warmup_steps、taylorseer_order、steps_computation_policy 等），供 diffusers 后端的 SCM 缓存测试使用。

关键文件：

- python/sglang/multimodal_gen/test/server/gpu_cases.py（模块 测试用例；类别 test；类型 test-coverage；符号 qwen_image_t2i_cache_dit_scm_config_diffusers_1gpu, wan2_1_t2v_1_3b_cache_dit_sp_only_2gpu）：主测试文件，添加了 cache-dit 相关的 1GPU 和 2GPU 测试用例
- python/sglang/multimodal_gen/test/server/configs/cache_dit_scm_config.yaml（模块 缓存配置；类别 test；类型 test-coverage）：新增的 SCM 缓存配置文件，定义了缓存策略参数，供 diffusers 后端测试使用

关键符号：未识别

关键源码片段

python/sglang/multimodal_gen/test/server/gpu_cases.py

主测试文件，添加了 cache-dit 相关的 1GPU 和 2GPU 测试用例

```
from dataclasses import replace
from pathlib import Path

# 定义配置目录常量
_CACHE_DIT_CONFIG_DIR = Path(__file__).parent / "configs"

# ONE_GPU_CASES 中添加 SCM 配置测试
DiffusionTestCase(
    "qwen_image_t2i_cache_dit_scm_config_diffusers_1gpu",
    DiffusionServerArgs(
        model_path=DEFAULT_QWEN_IMAGE_MODEL_NAME_FOR_TEST,
        extras=[
            "--backend",
            "diffusers",
            "--cache-dit-config",
            str(_CACHE_DIT_CONFIG_DIR / "cache_dit_scm_config.yaml"),
        ],
    ),
    # 使用 replace 继承默认参数并覆盖部分设置
    replace(
        T2I_sampling_params,
        output_size="512x512",
        extras={"num_inference_steps": 8, "seed": 0},
    ),
    # 禁用所有非必要检查以保持轻量
    run_perf_check=False,
    run_consistency_check=False,
    run_component_accuracy_check=False,
    run_models_api_check=False,
    run_t2v_input_reference_check=False,
),

# TWO_GPU_CASES 中添加 native SGLD 缓存 SP only 测试
DiffusionTestCase(
    "wan2_1_t2v_1_3b_cache_dit_sp_only_2gpu",
    DiffusionServerArgs(
        model_path=DEFAULT_WAN_2_1_T2V_1_3B_MODEL_NAME_FOR_TEST,
        ulysses_degree=2, # 仅 sequence parallelism
        enable_cache_dit=True,
        env_vars={"SGLANG_CACHE_DIT_WARMUP": "2"},
    ),
    replace(
```

```
            T2V_sampling_params,
            output_size="832x480",
            num_frames=5,
            extras={"num_inference_steps": 8, "seed": 0},
        ),
        # 同样禁用检查
        run_perf_check=False,
        run_consistency_check=False,
        run_component_accuracy_check=False,
        run_models_api_check=False,
        run_t2v_input_reference_check=False,
    ),
```

python/sglang/multimodal_gen/test/server/configs/cache_dit_scm_config.yaml

新增的 SCM 缓存配置文件，定义了缓存策略参数，供 diffusers 后端测试使用

```
cache_config:
  max_warmup_steps: 2 # 最大预热步数
  warmup_interval: 2 # 预热间隔
  max_cached_steps: -1 # 最大缓存步数（-1 表示不限）
  max_continuous_cached_steps: 2 # 最大连续缓存步数
  Fn_compute_blocks: 1 # Fn 计算块数
  Bn_compute_blocks: 0 # Bn 计算块数
  residual_diff_threshold: 0.12 # 残差差异阈值
  enable_taylorseer: true # 启用泰勒级数预测
  taylorseer_order: 1 # 泰勒级数阶数
  num_inference_steps: 8 # 推理步数
  steps_computation_mask: "medium" # 步计算掩码
  steps_computation_policy: dynamic # 动态计算策略
```

评论区精华

- SCM 测试必要性：@DefTruth 建议补充 SCM 相关测试，以避免缓存配置错误（#19955、#19965）。
- 测试轻量化：@mickqian 担心测试过重，@qimcis 随后移除了冗余用例并精简参数，仅保留关键检查。
- 配置文件组织：@mickqian 建议将分散的 YAML 配置统一放入 configs 目录，最终版本实现了这一建议。
  - SCM 配置测试的必要性 (testing): 作者 qimcis 接受了建议，新增了 qwen_image_t2i_cache_dit_scm_config_diffusers_1gpu 测试用例。
  - 测试用例重量优化 (testing): 最终版本仅包含两个测试，均为轻量化配置。
  - YAML 配置文件集中管理 (design): 作者将 cache_dit_scm_config.yaml 放置在 configs 目录下，并通过 _CACHE_DIT_CONFIG_DIR 引用。

风险与影响

- 风险：风险较低，仅为测试变更。2GPU 测试可能需要较多 CI 资源，可能延长 CI 总时间。SCM 配置参数若与缓存逻辑实现不同步更新，可能导致测试失效但不会影响生产。
- 影响：扩大缓存功能的测试覆盖，提高回归检测能力。对用户无直接影响，对开发者增加修改缓存逻辑时的安全性。CI 新增两个测试用例，总执行时间预计增加数分钟。
- 风险标记：测试资源消耗，低回归风险

关联脉络

- PR #16662 Add *cache*-dit support: 原始 *cache*-dit 功能 PR，本 CI 测试为其补充回归覆盖