

PR #29338 完整报告

sgl-project/sglang

[Spec] Add DFLASH basic sanity CI test

合并时间: 2026-06-29 14:45

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/29338>

执行摘要

- 一句话: 为 DFLASH 添加基础 sanity CI 测试
- 推荐动作: 值得关注 DFLASH 测试基础设施的搭建方式。对于其他推测解码路径 (如 EAGLE3) 的测试有借鉴意义。fwd_occupancy_kit 的格式改进也有参考价值。

功能与动机

DFLASH 推测解码路径目前缺少 CI 测试, 需要添加基础 sanity 测试以确保核心功能在代码变更后仍正常工作, 类似已有 test_basic_sanity.py 和 test_basic_sanity_eagle3.py 的模式。

实现拆解

1. 创建测试文件 test/registered/core/test_basic_sanity_dflash.py, 定义 TestBasicSanityDFlash 类, 继承多个 mixin (BasicAPIContractMixin、BasicDecodeCorrectnessMixin、BasicSchedulerStressMixin、FwdOccupancyMixin、GSM8KMixin) 以获得 API、解码、调度、占用率及 GSM8K 验证能力。
2. 在 setUpClass 中使用 popen_launch_server 启动服务器, 配置 DFLASH 算法、triton attention backend、cuda-graph-max-bs=4、mem-fraction-static=0.7、启用指标等, 并设定环境变量 SGLANG_ENABLE_METRICS_DEVICE_TIMER=1。
3. 设定特定的占用率阈值 (97.5) 和 GSM8K 准确率阈值 (0.74), 并注册为 stage=base-a 的 CI 任务 (估计运行时间 200s)。
4. 修改 python/sglang/test/kits/fwd_occupancy_kit.py 中 test_fwd_occupancy 方法的打印部分, 为性能指标和占用率统计表格添加文本标题行 ([perf metrics] 和 [fwd_occupancy stats]), 防止 CI 时间戳前缀破坏表格格式。

关键文件:

- test/registered/core/test_basic_sanity_dflash.py (模块 DFLASH 测试; 类别 test; 类型 test-coverage; 符号 TestBasicSanityDFlash, setUpClass, tearDownClass): 新增 DFLASH 基础 sanity 测试的核心文件, 覆盖多种验证维度。
- python/sglang/test/kits/fwd_occupancy_kit.py (模块 前向占用; 类别 test; 类型 test-coverage): 改进输出表格格式, 添加标题行以避免 CI 日志中表格混淆。

关键符号: TestBasicSanityDFlash.setUpClass, TestBasicSanityDFlash.tearDownClass, FwdOccupancyMixin.test_fwd_occupancy

关键源码片段

test/registered/core/test_basic_sanity_dflash.py

新增 DFLASH 基础 sanity 测试的核心文件，覆盖多种验证维度。

```
"""Stage-a basic sanity with DFLASH spec decoding enabled. Mirrors
test_basic_sanity.py / test_basic_sanity_eagle3.py with the DFLASH path active
(overlap scheduling on by default)."""

import unittest

from sglang.srt.utils import kill_process_tree
from sglang.test.ci.ci_register import register_cuda_ci
from sglang.test.kits.basic_api_contract_kit import BasicAPIContractMixin
from sglang.test.kits.basic_decode_correctness_kit import BasicDecodeCorrectnessMixin
from sglang.test.kits.basic_scheduler_stress_kit import BasicSchedulerStressMixin
from sglang.test.kits.eval_accuracy_kit import GSM8KMixin
from sglang.test.kits.fwd_occupancy_kit import FwdOccupancyMixin
from sglang.test.test_utils import (
    DEFAULT_DRAFT_MODEL_DFLASH, # DFLASH 默认 draft 模型路径
    DEFAULT_TARGET_MODEL_DFLASH, # DFLASH 默认 target 模型路径
    DEFAULT_TIMEOUT_FOR_SERVER_LAUNCH,
    DEFAULT_URL_FOR_TEST,
    CustomTestCase,
    popen_launch_server,
)

register_cuda_ci(est_time=200, stage="base-a", runner_config="1-gpu-small")

class TestBasicSanityDFlash(
    BasicAPIContractMixin,
    BasicDecodeCorrectnessMixin,
    BasicSchedulerStressMixin,
    FwdOccupancyMixin,
    GSM8KMixin,
    CustomTestCase,
):
    """使用多个 mixin 组合验证 DFLASH 路径的基本功能。"""

    # DFLASH 默认模型和配置
    served_model_name = DEFAULT_TARGET_MODEL_DFLASH
    fwd_occupancy_threshold = 97.5 # 前向占用率阈值
    fwd_occupancy_max_new_tokens = 4096 # 占用率测试的最大生成长度
    # DFLASH 每次验证接受一个完整块，其接受长度通常高于 EAGLE3
    fwd_occupancy_acc_length_threshold: float = 2.0

    model = DEFAULT_TARGET_MODEL_DFLASH
    gsm8k_num_questions = 1400
```

```

gsm8k_accuracy_thres = 0.74
gsm8k_accept_length_thres = 2.8

attention_backend = "triton"
draft_attention_backend = "triton"

@classmethod
def setUpClass(cls):
    """启动一个为 DFLASH 配置的 SGLang 服务器进程。"""
    cls.base_url = DEFAULT_URL_FOR_TEST
    cls.process = popen_launch_server(
        DEFAULT_TARGET_MODEL_DFLASH,
        cls.base_url,
        timeout=DEFAULT_TIMEOUT_FOR_SERVER_LAUNCH,
        other_args=[
            "--trust-remote-code",
            "--attention-backend", cls.attention_backend,
            "--speculative-draft-attention-backend", cls.draft_attention_backend,
            "--speculative-algorithm", "DFLASH",
            "--speculative-draft-model-path", DEFAULT_DRAFT_MODEL_DFLASH,
            "--cuda-graph-max-bs", "4",
            "--mem-fraction-static", "0.7",
            "--enable-metrics",
            "--disable-piecewise-cuda-graph",
        ],
        env={"SGLANG_ENABLE_METRICS_DEVICE_TIMER": "1"},
    )

@classmethod
def tearDownClass(cls):
    """终止服务器进程。"""
    kill_process_tree(cls.process.pid)

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

```

评论区精华

主要讨论围绕 attention backend 的选择：最初指定 trtllm_mha 作为 target backend，后尝试 fa4，最终改为 triton 以兼容 RTX 5090 (sm120) 硬件。作者通过多次 CI rerun 确认 triton 组合能够通过测试。

- Attention backend 选择 (design): 采用 triton 作为 target 和 draft attention backend。

风险与影响

- 风险：测试覆盖范围有限：仅覆盖基础 sanity 场景，未测试长序列、多批次、压力等复杂情况。对硬件有依赖：当前配置针对 sm120 (RTX 5090)，可能不兼容其他 GPU 架构。增加

CI 负担：测试估计运行 200s，属于较长测试，可能影响 CI 队列效率。

- 影响：对用户无直接影响。对开发团队：DFLASH 路径将获得 CI 保护，防止回归。对 CI 系统：新增一个 stage=base-a 的测试，运行时长约 200s，可能会略微增加整体 CI 完成时间。
- 风险标记：测试覆盖有限，硬件依赖，CI 负担增加

关联脉络

- PR #29343 [dflash] fa3/fa4: device-side page table; drop seq_lens_cpu D2H sync: 优化了 DFLASH 的 attention 实现，与本 PR 测试的 DFLASH 路径直接相关。
- PR #29541 [Spec] Publish DFLASH verify read-done event for fine-grained WAR barrier: 改进了 DFLASH verify 阶段的同步机制，本 PR 的 sanity 测试可覆盖此功能。