

PR #39013 完整报告

sgl-project/sglang

[CI] Trim DSV4 trtllm B200 tests

合并时间: 2026-09-11 16:04

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/39013>

执行摘要

- 一句话: 裁剪 DSV4 trtllm B200 CI: 删重复测试文件、4 个 recipe 收敛为 2 个
- 推荐动作: 值得一读, 尤其是维护 DSV4 后端 CI 与测试基础设施的同学: 这是一份罕见的把「删测试」写成完整覆盖映射的 PR。建议重点关注三点: (1) 被删文件独有能力与被保留类的映射是否真的完备, 特别是「TP4、非投机、非 DP」组合是否需要补一个最小 e2e (可以从 BCG 类去掉 --dp / --enable-dp-attention 派生); (2) test_long_prompt_chunked_prefill_recall 的跨 chunk needle 召回写法值得作为模板沉淀进 test kit; (3) est_time 与实测耗时的对齐方式 (per-file、~460s → 500) 以及合并前 CI 失败重跑的流程, 是否需要在团队内形成统一约定。

功能与动机

PR body 指出, 被裁剪的 CI 由 #30805 引入, 但「duplicated the shared test kits and itself»: test_dsv4_fp8_trtllm_backend.py 里的 ascii-ratio 检查 (_printable_ascii_ratio / _assert_sane) 重复了 BasicDecodeCorrectnessMixin.test_ascii_ratio, test_gsm8k_sanity 重复了 GSM8KMixin, test_cuda_graph_capture_replay_smoke 重复了 test_determinism_temp_zero; 同时 FP4 B200 文件里 4 个 recipe 类意味着每个 commit 要多次启动 4 卡 B200 服务。目标是在不损失独特覆盖的前提下压缩 B200 CI 成本与重复代码。

实现拆解

1. 删除重复测试文件: 整体移除 test/registered/e2e/dsv4/test_dsv4_fp8_trtllm_backend.py (-214 行)。该文件自带的 _is_sm100() skip、_printable_ascii_ratio()、_assert_sane() 与 900s 独立 CI 预算, 与其覆盖的「TP4 + DSV4-Flash + --dsv4-attn-backend trtllm」服务配置, 几乎完全被保留文件中的 LowLatency 类覆盖; 断言能力则由共享 test kit 承担。
2. 迁移唯一独有路径并升级断言: 在 test/registered/e2e/models/test_deepseek_v4_flash_fp4_b200_trtllm.py 的 TestDSV4FlashFP4B200Trtllm 中新增类常量 NEEDLE = "48173" 与 test_long_prompt_chunked_prefill_recall()。prompt 由 needle 句 + 220 段 filler + 末尾提问拼成, needle 落在第一个 chunk、问题落在最后一个 chunk, 因此只有多 chunk 的 _forward_trtllm_prefill 正确才能召回——这比原来的「输出非空 + ascii 比例」强得多。循环标签 ("cold", "cached-prefix") 使第二遍从 radix cache 前缀续跑, 额外覆盖 cached-prefix extend。

3. 裁剪 recipe 组合：删除 TestDSV4FlashFP4B200BalancedTrtllm 与 TestDSV4FlashFP4NonMTPB200Trtllm 两个类（-90 行）。body 给出的依据是：Balanced 只增加 MoE 侧（TP4 + DP4 + DeepEP + 1-step EAGLE）与 attention 正交的覆盖，NonMTP 是 BCG 的子集。保留的两个类是语义互补的 LowLatency（draft extend / target verify / 多步 backend，隐含 EAGLE 3-step）与 BCG（DP padding、breakable graph replay、mixed chunk），后者 docstring 补上了 "no spec" 说明；register_cuda_ci(est_time=700 → 500, stage="base-c", runner_config="4-gpu-b200")。
4. 单元测试与配置同步：test/registered/attention/unittests/dsv4/test_deepseek_v4.py 的 test_trtllm_semaphore_capacity_covers_configured_query_rows 中，schedule 桩由 SimpleNamespace(max_prefill_tokens=16384, chunked_prefill_size=4096, max_running_requests=256) 精简为只保留 max_prefill_tokens / max_running_requests (+1/-5)，仍与 max_prefill_buffer_tokens 的 patch 配合，16384 / 32768 两条断言不变。
5. 验证配套：作者在 8x B300（SM103）上执行 dsv4 单测 22 passed、LowLatency 10 passed（241s）、BCG 8 passed（219s）；随后用 /rerun-test -c 在 4-gpu-b200（2 tests，含 test_deepseek_v4.py 与 test_deepseek_v4_flash_fp4_b200_trtllm.py）与 1-gpu-h100（1 test）车道全部通过。

关键文件：

- test/registered/e2e/models/test_deepseek_v4_flash_fp4_b200_trtllm.py（模块 端到端测试；类别 test；类型 test-coverage；符号 TestDSV4FlashFP4B200Trtllm, test_long_prompt_chunked_prefill_recall, setUpClass, tearDownClass）：本 PR 的核心载体：4 个 recipe 收敛为 LowLatency + BCG 两个类，删除 Balanced / NonMTP，新增 NEEDLE 常量与跨 chunked-prefill 的 needle 召回测试，并把 est_time 由 700 校正为 500。
- test/registered/e2e/dsv4/test_dsv4_fp8_trtllm_backend.py（模块 端到端测试；类别 test；类型 deletion；符号 TestDSV4Fp8TrtllmBackend, _make_long_prompt, _is_sm100, _greedy_generate）：被整体删除的测试文件（-214 行）。其 _printable_ascii_ratio / _assert_sane / test_gsm8k_sanity / test_cuda_graph_capture_replay_smoke 分别与 BasicDecodeCorrectnessMixin.test_ascii_ratio、GSM8KMixin、test_determinism_temp_zero 重复，是本次 CI 裁剪的主要收益来源。
- test/registered/attention/unittests/dsv4/test_deepseek_v4.py（模块 注意力单测；类别 test；类型 test-maintenance；符号 test_trtllm_semaphore_capacity_covers_configured_query_rows）：随裁剪同步的精简：test_trtllm_semaphore_capacity_covers_configured_query_rows 的 schedule 桩去掉 chunked_prefill_size 字段，保持对 _trtllm_query_row_capacity 的两条容量断言不变。

关键符号：test_long_prompt_chunked_prefill_recall,

TestDSV4FlashFP4B200Trtllm.setUpClass, TestDSV4FlashFP4BreakableCudaGraphB200Trtllm.setUpClass, test_trtllm_semaphore_capacity_covers_configured_query_rows, TestDSV4Fp8TrtllmBackend.test_long_prompt_varlen_prefill, TestDSV4Fp8TrtllmBackend._assert_sane

关键源码片段

test/registered/attention/unittests/dsv4/test_deepseek_v4.py

随裁剪同步的精简：test_trtllm_semaphore_capacity_covers_configured_query_rows 的 schedule 桩去掉 chunked_prefill_size 字段，保持对 _trtllm_query_row_capacity 的两条容量断言不变。

```
# test/registered/attention/unittests/dsv4/test_deepseek_v4.py
def test_trtllm_semaphore_capacity_covers_configured_query_rows(self):
    from sglang.srt.layers.attention import deepseek_v4_trtllm_backend as trtllm

    # 桩对象只保留容量计算真正读取的字段；chunked_prefill_size 已属冗余，
    # 移除后不改变任何断言结果，也让桩与 max_prefill_buffer_tokens 的 patch 更对称
    schedule = SimpleNamespace(max_prefill_tokens=16384, max_running_requests=256)
    spec = SimpleNamespace(
        speculative_algorithm="EAGLE", speculative_num_draft_tokens=4
    )
    model_runner = SimpleNamespace()
    with (
        mock.patch.object(trtllm, "get_schedule", return_value=schedule),
        mock.patch.object(trtllm, "get_spec", return_value=spec),
        mock.patch.object(trtllm, "max_prefill_buffer_tokens", return_value=4096),
    ):
        # prefill chunk / max_prefill_tokens 占主导
        self.assertEqual(trtllm._trtllm_query_row_capacity(model_runner), 16384)
        # decode 行数 = 请求数 x draft tokens 占主导
        schedule.max_running_requests = 8192
        self.assertEqual(trtllm._trtllm_query_row_capacity(model_runner), 32768)

    # 持久化 semaphore 行数不足时必须显式报错，而不是静默截断
    with mock.patch.object(trtllm, "_trtllm_semaphore_rows", 64):
        trtllm._check_trtllm_query_rows(64)
        with self.assertRaisesRegex(RuntimeError, "exceeds the persistent"):
            trtllm._check_trtllm_query_rows(65)
```

评论区精华

本 PR 没有任何行内 review 评论：reviewer Fridge003 直接给出空正文的 APPROVED，唯一的人工互动是作者 mmangkad 在 body 的 pr-states 显示三条车道失败（pr-test / pr-test-extra / amd-rocm720）后发出 `/rerun-test -c`，bot 回报 4-gpu-b200 与 1-gpu-h100 车道^②。因此全部技术论证都集中在 PR body 里，实质上是一份「覆盖映射 + 成本账」的自证材料：重复项逐条对到共享 kit，保留项逐条对到 recipe 语义（Balanced 属 MoE 正交覆盖、NonMTP 属 BCG 子集），并附上 SM103 实测耗时与单测通过数。没有出现反对意见，也没有人为「纯 TP4、非投机、非 DP」路径的覆盖缺口提出质询——这是唯一未被讨论触及的点。

- 删除整个 FP8 trtllm 测试文件是否会丢失覆盖 (testing)：覆盖被判定为等价或更强，删除文件并通过迁移保留独有能力。

- Balanced 与 NonMTP 两个 recipe 的下线依据 (design): 按「与 attention 后端正交」的标准裁剪 recipe, 接受低成本换取更集中的覆盖。
- est_time 语义修正: 700 → 500 (performance): 预算按实测值校正, 写成 per-file 口径。
- 合并前 CI 车道失败与重跑 (testing): 以重跑通过作为合入依据, 但未记录原始失败原因。
- 单测 schedule 桩字段精简 (testing): 纯桩对象清理, 不改变语义; 若源码重新读取该字段会立即以 AttributeError 暴露。

风险与影响

- 风险:
 1. 覆盖缺口 (中): 删除 test_dsv4_fp8_trtllm_backend.py 后, B200 上不再有一个「TP4、非 DP、非投机」的独立服务实例。保留的 BCG 类同样不开 spec, 但带 --dp 4 --enable-dp-attention --enable-mixed-chunk --cuda-graph-backend-prefill breakable, 与纯 TP4 路径的 kernel 选择、DP padding、graph 准入条件并不相同; 若 trtllm 后端在非投机且非 DP 的组合上回归 (例如 semaphore 容量、decode graph bucket 选择, 参考 #38992 修过的 DSA 变体准入问题), B200 e2e 可能捕获不到。
 2. 断言强度与稳定性 (中低): 新的 test_long_prompt_chunked_prefill_recall 依赖模型在跨 chunk 预填充后复述 "48173"。断言从「非空 + ascii 比例」变成内容召回, 检测力显著提升, 但对 max_new_tokens (sanity_max_new_tokens_short) 与采样温度更敏感, 长 prompt (220 段 filler) 也会增加单例耗时, 需要观察后续 lane 是否出现随机失败。
 3. CI 时间预算 (低): est_time 由 700 降为 500, 而实测为 241s + 219s ≈ 460s, 余量约 40s, 接近入门阈值; B200 队列抖动或 server 冷启动变慢时可能超时, 应关注后续 lane 的实际耗时统计。
 4. 合入前的失败记录 (低): body 中 pr-test / pr-test-extra / amd-rocm720 三条车道显示失败, 仅通过重跑相关车道后合并, 未记录失败根因; 如果是与裁剪无关的基础设施抖动则无碍, 否则可能掩盖真实问题。
 5. 单测 mock 精简 (很低): 移除 chunked_prefill_size 字段本身不改变断言, 但若后续 deepseek_v4_trtllm_backend 重新读取该字段, 测试会因为 SimpleNamespace 缺属性直接报 AttributeError——属于可接受的白盒约束。- 影响: 对 CI/ 团队: 4 卡 B200 资源占用与每个 commit 的服务启动次数明显下降 (body 表述 Three 4-GPU B200 server launches per commit become one), DSV4 trtllm 的回归网从「多个近似重复 recipe 的浅断言」转为「两个语义互补 recipe 的深断言」, 同时为后续 DSV4 (V4.1) 相关改动腾出 base-c 预算。对工程质量: 删除自研 helper 后, 新增 DSV4 trtllm 测试必须复用 BasicDecodeCorrectnessMixin / GSM8KMixin / SpecDecodingMixin, 减少同类重复再次出现。对用户: 几乎零影响, 本 PR 不触碰 python/sglang 运行时代码, 只影响 CI 覆盖与建库信心。对后续维护者: needle 召回这种「把关键信息放在首 chunk、把问题放在末 chunk」的写法成为跨 chunked-prefill 验证的范式, 可复制到其他长上下文分块预填充测试。- 风险标记: B200 非投机 TP4 路径 e2e 覆盖减少, CI est_time 与实测耗时余量偏紧, 合并前 CI 失败根因未记录, 新召回断言依赖模型数字复述能力

关联脉络

- PR #30805 DSV4 trtllm attention backend CI (本 PR body 中引用的来源 PR, 标题未在上下文中提供): PR body 明确说明本次裁剪的对象就是 #30805 为 DSV4 trtllm attention 后端引入的 CI, 本 PR 是其覆盖与成本的反向修正。
- PR #38947 [Refactor] Clarify DeepSeek V4 metadata names for V4.1: 同样改动 test/registered/attention/unittests/dsv4/test_deepseek_v4.py 与 DSV4 attention 元数据符号, 属于同一 DSV4 注意力测试面的并发演进, 本 PR 的单测精简建立在其重命名后的符号之上。
- PR #38954 [Refactor] Generalize DeepSeek V4 compressed pool management: 同一 DSV4 V4.1 准备序列中的重构, 涉及 deepseek_v4_backend / dsv4 metadata 与压缩池, 与本次精简覆盖的 trtllm/ 稀疏 MLA 测试对象处于同一模块线。
- PR #36655 [SM120] Use exact query-head widths for DeepSeek-V4 sparse MLA decode: 同样围绕 DSV4 稀疏 MLA decode 的测试与 Blackwell 系 (SM120/SM100/SM103) 后端行为, 说明该后端的测试面正在被持续重整。
- PR #38992 [Bug] Include DSA variant in exact-bucket graph admission: 修的是 DeepSeek 注意力变体在 decode CUDA graph 准入上的回退问题; 本 PR 删掉的 test_cuda_graph_capture_replay_smoke 与保留的 BCG 类都在覆盖同类 graph 行为, 构成「graph 准入回归需要足够 CI 覆盖」的对照。