

PR #56528 完整报告

vllm-project/vllm

[Test][Determinism] Cover VLM batch invariance in default execution mode

合并时间: 2026-09-13 22:22

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/56528>

执行摘要

- 一句话: 新增 VLM 默认执行模式批次不变性用例并收紧判定断言
- 推荐动作: 值得精读, 但阅读重点不是 diff 体量而是测试设计: 它是纯测试变更且已被 yewentao256 以 LGTM 合并, 逐行 review 价值有限。建议关注三处决策——(1) `_assert_batch_invariant` 由 `skip` 改为 `fail-closed`, 把「静默通过」换成「显式失败」, 值得在仓库其他条件跳过型用例中复制; (2) 用 `enable_prefix_caching=False`、`mm_processor_cache_gb=0`、`VLLM_USE_FLASHINFER_SAMPLER=0`、`ignore_eos=True` 这组旋钮把变量收敛到「执行模式」单一维度; (3) 只加一格而非铺满 `image/video × backend` 全矩阵的取舍。若要继续扩展, 建议补显存 / 硬件门槛标记、把 Hopper/Blackwell 验证沉淀进 CI 矩阵而不是依赖作者本地复跑, 并明确 `prefill/encoder` 图捕获仍是空白。

功能与动机

PR body 给出的直接动因是覆盖缺口: 「#53531 added the current VLM batch-invariance test, but all of its cases set `enforce_eager=True`. This PR adds one deliberately narrow default-execution cell, which exercises CUDA Graph decode without multiplying the full `image/video × encoder-backend` matrix.」同时 body 指出「The maintainer also invited investigation of non-eager mode in #27433」, 即维护者此前已邀请社区排查非 `eager` 模式。第二个动机是消除假绿: 原 `_assert_batch_invariant` 在 `selected-token logprobs` 缺失时直接 `pytest.skip`, 用例可能「没校验任何东西却算通过」, `commit message` 里写明要把它换成显式失败。

实现拆解

1. 把判定函数从「可跳过」改成 `fail-closed`: `tests/v1/determinism/test_batch_invariance_vlm.py` 中的 `_assert_batch_invariant` 是唯一判定入口。原先任一侧取不到 `selected-token logprobs` 就直接 `pytest.skip`, 现在改为把该请求写入 `failures` 并 `continue`, 最后统一 `pytest.fail`; 同时新增两层前置断言——`llm.generate(inputs, sampling, use_tqdm=False)` 的返回条数必须等于输入条数, 以及每条结果 `token` 数必须等于 `sampling.max_tokens`, 先把截断 / 空结果挡在逐位比较之前。影响面不止新用例: 原有 4 个 `eager` 组合也一并变严。
2. 抽出共享 runner 并参数化执行模式: 新增 `_run_vlm_batch_invariance(input_type, mm_encoder_attn_backend, enforce_eager)`, 把 `_make_inputs` 造输入、构造

SamplingParams、起 LLM、跑断言、llm.shutdown() 收敛到一处；
test_vlm_batch_invariance_bs1_vs_bsN 退化为薄壳，继续保留 image/video x
FLASH_ATTN/TORCH_SDPA 的 4 个组合并统一传 enforce_eager=True。这样改的原因是原测试体在用例里硬编码 LLM(enforce_eager=True, ...)，无法在保住 eager 覆盖的同时再开默认执行模式。

3. 补齐默认执行模式用例：新增 test_vlm_batch_invariance_default_execution，刻意只跑 image + FLASH_ATTN 一格，传 enforce_eager=False，让 decode 阶段走 CUDA Graph，从而检验「单请求 vs 同批」在 graph replay 下是否仍逐 bit 一致。PR body 里的插桩验证称，四请求默认执行运行中 7 个目标 decode step 全部为 FULL CUDA Graph replay，且每步至少 3 个逻辑活跃请求，四张图都真正过了 encoder。
4. 用配置把干扰变量隔离干净：共享 runner 的 SamplingParams 增加 ignore_eos=True（保证固定 8 个 token，配合新长度断言）；LLM 增加 max_model_len=4096、max_num_batched_tokens=2048、enable_prefix_caching=False、mm_processor_cache_gb=0；两个用例都通过 monkeypatch.setenv("VLLM_USE_FLASH_INFER_SAMPLER", "0") 关掉 FlashInfer 采样器。目的都是让「执行模式」成为唯一被比较的变量，避免前缀缓存命中或采样内核差异污染结论。
5. 测试与验证配套：本 PR 不含 vllm/ 下任何源码、配置或 schema 改动，风险面集中在测试本身。验证方式是把用例在 RTX 4090 (SM89)、RTX 5090 (SM120, 32 GB)、H800 (SM90) 上分别跑通，并把结果写进 PR description 与评论；CI 由维护者 yewentao256 通过 /ci run 触发（fork PR 的自动审查被禁用）。

关键文件：

- tests/v1/determinism/test_batch_invariance_vlm.py（模块 确定性测试；类别 test；类型 test-coverage；符号 _assert_batch_invariant, test_vlm_batch_invariance_bs1_vs_bsN, test_vlm_batch_invariance_default_execution, _run_vlm_batch_invariance）：唯一变更文件：抽出共享 runner、新增默认执行模式用例，并把 logprobs 缺失由 skip 改为 fail-closed，是本次全部行为的承载点。

关键符号：_run_vlm_batch_invariance, test_vlm_batch_invariance_default_execution, test_vlm_batch_invariance_bs1_vs_bsN, _assert_batch_invariant

关键源码片段

tests/v1/determinism/test_batch_invariance_vlm.py

唯一变更文件：抽出共享 runner、新增默认执行模式用例，并把 logprobs 缺失由 skip 改为 fail-closed，是本次全部行为的承载点。

判定函数：_assert_batch_invariant (fail-closed 化 + 长度断言)

```
def _assert_batch_invariant(llm: LLM, inputs: list[dict], sampling) -> None:
    # bs1: 逐条单独跑，作为「单请求」基准
    bs1 = [
        _extract_step_logprobs(llm.generate([inp], sampling, use_tqdm=False)[0])
        for inp in inputs
    ]
    # bsN: 四条请求一次性同批跑，作为「批内」对照
```

```

bsN_outputs = llm.generate(inputs, sampling, use_tqdm=False)
assert len(bs1) == len(inputs)
assert len(bsN_outputs) == len(inputs)
bsN = [_extract_step_logprobs(out) for out in bsN_outputs]

failures = []
for i, ((lp1, t1), (lpN, tN)) in enumerate(zip(bs1, bsN)):
    if lp1 is None or lpN is None:
        # 关键改动：原来是 pytest.skip，用例会在「没校验任何东西」的情况下算通过，
        # 现在改成记为失败并继续检查其余请求，保证断言真正生效
        failures.append(f"req {i}: selected-token logprobs are unavailable")
        continue
    if len(t1) != sampling.max_tokens or len(tN) != sampling.max_tokens:
        # 新增：先卡生成长度，避免请求被提前截断后仍被当作「不变」
        failures.append(
            f"req {i}: expected {sampling.max_tokens} tokens, "
            f"got bs1={len(t1)} bsN={len(tN)}"
        )
        continue
    if t1 != tN:
        failures.append(f"req {i}: token mismatch bs1={t1} bsN={tN}")
    elif not torch.equal(lp1, lpN):
        # torch.equal 要求逐 bit 相同，而不是只看误差阈值，这正是 batch invariance 的定义
        d = (lp1 - lpN).abs().max().item()
        failures.append(f"req {i}: logprob mismatch max_diff={d:.3e}")

if failures:
    pytest.fail(
        f"Batch invariance violated for {len(failures)}/{len(inputs)} "
        f"requests:\n" + "\n".join(failures)
    )

```

评论区精华

讨论量小但方向明确：维护者只提了一条实质要求，即跨硬件验证。

1. [yewentao256](#) (COMMENTED) : 「Thanks for the work, I think you might also want to test on Hopper/Blackwell to make sure it works」。作者随后在 RTX 5090 (SM120, 32 GB, driver 595.71.05) 复跑两个目标用例，[test_vlm_batch_invariance_default_execution](#) 156.02s 通过、[test_vlm_batch_invariance_bs1_vs_bsN\[FLASH_ATTN-image\]](#) 36.10s 通过，并解释耗时差异来自「first-use Triton/[ptxas-blackwell](#) compilation for SM120」；之后又补了 H800 (SM90) 结果：115.29s 与 49.91s 通过，并指出「The 15 warnings in each run were deprecation warnings and did not affect the results」。
2. [claude\[bot\]](#): 「This pull request is from a fork — automated review is disabled.」最终由维护者给出 **APPROVED**: 「LGTM, thanks for the work!」
3. 没有出现针对测试设计本身的反对意见：skip→fail-closed 的语义收紧、只加一格而不铺满矩阵的取舍、以及关闭两级缓存与 FlashInfer sampler 的隔离手法均在 PR body 中自证，

评论区未提出反驳或补充要求。

- 跨硬件验证要求 (Hopper / Blackwell) (testing): 跨硬件验证要求被满足: SM89/SM90/SM120 三个代际均有通过记录, 维护者随后以 LGTM 批准。
- fork PR 的自动审查被禁用, 依赖维护者手动触发 CI (other): 机器人审查缺位由维护者人工 review 加 CI 触发补上, 流程闭环, 无技术争议。
- 判定语义收紧: logprobs 缺失由 skip 改为 fail-closed (testing): 收紧后的断言连同新增的 `len(t) == sampling.max_tokens` 检查一并合并, 原有 eager 组合同步受益。

风险与影响

- 风险: CI 时长与硬件门槛: 默认执行模式用例需要 CUDA Graph 捕获加首次 Triton 编译, H800 上 115s、RTX 5090 上 156s; 该用例只受 `@skip_if_not_cuda` 约束, 会在任何 CUDA 设备上运行, 而 `gpu_memory_utilization=0.85 + 4 条 256px 图像请求在显存更小的卡上是否有余量`, 材料中没有验证记录。

测试严格化带来的连带失败: `_assert_batch_invariant` 把 logprobs 缺失从 `pytest.skip` 改为失败, 若上游改动使 `RequestOutput` 不再返回 selected-token logprobs, 原有 4 个 eager 组合会一起变红; 新增的 `len(t) == sampling.max_tokens` 断言依赖 `ignore_eos=True` 生效, 采样默认值或 stop 逻辑变化同样会误报。

逐 bit 断言对内核非确定性天然敏感: 判定使用 `torch.equal` 而非误差阈值, 这正是 batch invariance 的定义, 但任何非确定性 kernel (attention、采样、cudagraph 相关) 都可能让测试间歇性失败; 本次通过关闭 FlashInfer 采样器与两级缓存压低了误报面, 但隔离手法本身依赖这些开关仍被尊重。

覆盖声明容易被高估: PR body 明确 target prefill 与 vision encoder 仍是 eager, 因此默认执行模式下的 prefill/encoder 图捕获并未被覆盖, 解读结果时需限定在 decode 路径。

环境变量注入时机: `monkeypatch.setenv("VLLM_USE_FLASHINFER_SAMPLER", "0")` 只有在 `LLM()` 构造时读取该 env 才有效; 若该开关未来被提升到 import 期解析, 隔离会静默失效而测试仍显示通过。

- 影响: 对用户与线上服务零影响: 不修改 vllm/ 下任何源码、配置或默认行为, 模型输出与部署方式不变, PR body 也说明这是 test-only change、no model-quality evaluation is applicable。

对工程团队的影响集中在 CI 与回归能力: `tests/v1/determinism/` 新增一条默认执行模式 (CUDA Graph decode) 的批次不变性守卫, 未来 cudagraph、采样或 VLM 执行路径若引入 batch 相关数值分歧, 可在 CI 阶段暴露; 代价是 CI 增加一条约 2-3 分钟的 GPU 用例。

对验证标准的影响是跨硬件矩阵被显式扩展为 SM89/SM90/SM120 三个代际, 为后续 CUDA Graph 相关改动提供了可复现的对照环境; 同时 fail-closed 的判定语义会被同目录其他用例继承。

- 风险标记: 测试 -only 变更, CI 耗时增加, 逐 bit 断言易受内核非确定性影响, 覆盖范围仅限 decode, fail-closed 语义收紧影响既有用例

关联脉络

- PR #53531 引入当前 VLM batch-invariance 测试的 PR (PR body 引用, 标题未在上下文中提供): PR body 明确说明现有 VLM batch-invariance 测试由 #53531 引入, 且其全部用例都设了 `enforce_eager=True`; 本 PR 正是在其基础上抽出共享 runner 并补上默认执行模式的一格。
- PR #56349 [ROCm] Auto-enable breakable CUDA graphs for DeepseekV41ForCausalLM: 同属 CUDA Graph / 非 eager 执行路径的开关与默认行为调整; 本 PR 为默认执行模式下 CUDA Graph decode 补上确定性回归守卫, 与仓库近期在图捕获路径上的加固趋势一致。
- PR #55508 [Bugfix][Benchmark] Make streaming TTFT/E2E latency accounting consistent across endpoints: 同是纯测试补强的「口径一致性」类改动: 一个统一三端点的计时口径, 一个统一 bs1 与 bsN 的数值判定口径, 在「让比较标准唯一、避免指标解释漂移」的方法论上可互相参照。