

PR #52566 完整报告

vllm-project/vllm

[ROCm][CI] Restore Torch defaults and type DSV4 scratch buffers

合并时间: 2026-08-18 02:32

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/52566>

执行摘要

- 一句话: 恢复 Torch 默认值并锁定 DSV4 scratch 为 FP32, 修复 ROCm CI
- 推荐动作: 值得快速精读, 改动仅 22 行、逻辑直白, 但对测试编写者有实际借鉴价值。两个设计决策值得关注: (1) 用 autouse fixture 统一管理进程级单例状态 (默认 dtype、device、lru cache), 并用 try/finally 保证失败也能还原; (2) 治本与治标结合——既消除污染源, 又让下游用例对 dtype 显式化、不再依赖执行顺序。若仓库内还有其他测试存在裸改全局 torch 默认值的情况, 建议按同一模式收敛。

功能与动机

PR body 明确给出了根因与触发链: An earlier HIP MHA test left the process-wide default dtype as FP16, causing the newly added DSV4 test to overflow while constructing its sentinel scratch tensor in this MI355 Buildkite failure, 并注明失败发生在 kernel launch 之前。由于 torch.set_default_dtype 是进程级全局状态, 而 pytest 在同一进程中顺序执行用例, 前面用例的修改会泄漏到后面用例; DSV4 测试中 torch.full 未显式指定 dtype, 按默认 dtype (FP16) 推断后, FP32 最小值 (约 -3.4e38) 溢出。修复目标因此确定为两条: 恢复测试隔离, 以及让 scratch 的 dtype 与生产实现一致 (FP32)。

实现拆解

本 PR 的变更全部集中在测试层, 不涉及生产代码、schema 或部署配置。实现按以下步骤拆解:

1. 根因定位 (跨文件状态污染): tests/kernels/attention/test_mha_attn.py 中的 test_mha_attn_platform 为了验证 MHA/MMEncoder 在不同平台下的 backend 选择, 在用例内部调用 torch.set_default_dtype(torch.float16), 但用例结束后没有还原。原有的 autouse fixture clear_cache 只清空了 _cached_get_attn_backend 的 lru cache, 管不到 torch 的全局默认 dtype, 因此后续在同一 CI 任务中执行的 tests/kernels/attention/test_rocm_triton_attn_dsv4.py 继承了这个 FP16 默认值。
2. 修复一: 升级 autouse fixture 为 reset_test_state: fixture 在 yield 前先记录 torch.get_default_device() 与 torch.get_default_dtype(), 并在 finally 块中恢复这两个值、再次 _cached_get_attn_backend.cache_clear()。try/finally 保证即使断言失败也会还原全局状态, 同时消除了对用例执行顺序的隐式依赖。

3. 修复二：DSV4 scratch 显式指定 FP32：在 `test_sparse_attn_decode_gfx950_adaptive_reduce_ignores_stale_scratch` 中，`part_m`（-inf 哨兵）与 `part_acc`（nan 累加器）的 `torch.full` 调用补上 `dtype=torch.float32`。这样即使默认 dtype 再次被污染，FP32 最小值也不会再在构造时溢出，且与生产 reducer 的 FP32 scratch 缓冲语义一致。
4. CI 与格式配套：首次 push 后 pre-commit 检查失败（`mergify[bot]` 提示），作者修正后重新提交并先后触发 Buildkite CI #84144、#84224；review 由 `tjtanaa` APPROVED，无遗留技术争议。

关键文件：

- `tests/kernels/attention/test_mha_attn.py`（模块 注意力测试；类别 test；类型 test-coverage；符号 `reset_test_state`, `clear_cache`）：本 PR 的根治点：将 `autouse` fixture 从仅清 `selector` 缓存升级为完整测试状态重置，在 `finally` 中恢复 `torch` 默认设备与 `dtype`，杜绝跨用例全局状态泄漏。
- `tests/kernels/attention/test_rocm_triton_attn_dsv4.py`（模块 内核测试；类别 test；类型 test-coverage；符号 `test_sparse_attn_decode_gfx950_adaptive_reduce_ignores_stale_scratch`）：为 DSV4 reducer 的哨兵与累加器张量显式指定 FP32，消除对进程默认 dtype 的依赖，与生产 reducer 的 scratch 缓冲类型保持一致。

关键符号：`reset_test_state`, `clear_cache`, `test_mha_attn_platform`,
`test_sparse_attn_decode_gfx950_adaptive_reduce_ignores_stale_scratch`

关键源码片段

`tests/kernels/attention/test_mha_attn.py`

本 PR 的根治点：将 `autouse` fixture 从仅清 `selector` 缓存升级为完整测试状态重置，在 `finally` 中恢复 `torch` 默认设备与 `dtype`，杜绝跨用例全局状态泄漏。

```
# tests/kernels/attention/test_mha_attn.py
```

```
@pytest.fixture(autouse=True)
```

```
def reset_test_state():
```

```
    """清除缓存选择器并恢复进程级 torch 默认状态。"""
```

```
    # 先记录进入测试前的默认设备与默认 dtype，确保无论测试内部如何修改都能还原
```

```
    default_device = torch.get_default_device()
```

```
    default_dtype = torch.get_default_dtype()
```

```
    # 清空 attention backend 的 lru cache，避免用例之间复用选择结果
```

```
    _cached_get_attn_backend.cache_clear()
```

```
    try:
```

```
        yield
```

```
    finally:
```

```
        # 还原默认设备与默认 dtype：防止后续用例（如 DSV4 测试）继承 FP16，
```

```
        # 导致 FP32 哨兵值构造时发生溢出或截断
```

```
        torch.set_default_device(default_device)
```

```
        torch.set_default_dtype(default_dtype)
```

```
        # 再次清空缓存，避免本用例的选择结果污染下一次用例
```

```
        _cached_get_attn_backend.cache_clear()
```

tests/kernels/attention/test_rocm_triton_attn_dsv4.py

为 DSV4 reducer 的哨兵与累加器张量显式指定 FP32，消除对进程默认 dtype 的依赖，与生产 reducer 的 scratch 缓冲类型保持一致。

```
# tests/kernels/attention/test_rocm_triton_attn_dsv4.py

@requires_gfx950
@torch.inference_mode()
def test_sparse_attn_decode_gfx950_adaptive_reduce_ignores_stale_scratch() -> None:
    device = torch.device("cuda")
    # 显式指定 FP32: 即使进程级默认 dtype 意外被其他用例改成 FP16,
    # FP32 最小值 (约 -3.4e38) 也不会在构造时溢出为 -inf 或 nan
    part_m = torch.full(
        (1, 8, 1),
        torch.finfo(torch.float32).min,
        dtype=torch.float32,
        device=device,
    )
    part_l = torch.zeros_like(part_m)
    # 与生产 reducer 的 scratch 语义一致: 累加器必须使用 FP32, 不能跟随默认 dtype
    part_acc = torch.full(
        (1, 8, 1, HEAD_DIM),
        float("nan"),
        dtype=torch.float32,
        device=device,
    )
    # 模拟只对前两行做在线 softmax 更新的场景: 其余位置保持 -inf / nan 哨兵
    part_m[:, :2] = 0
    part_l[:, :2] = 1
    part_acc[:, 0] = 1
    part_acc[:, 1] = 3

    actual = _launch_sparse_decode_reduce(part_m, part_l, part_acc, True)
    # 期望结果: 所有头汇聚输出为 2 (即 1 与 3 的加权平均), 且不受陈旧 scratch 影响
    assert torch.isfinite(actual).all()
    assert torch.equal(actual, torch.full_like(actual, 2))
```

评论区精华

该 PR 的 review 讨论密度很低，没有围绕实现方案的技术辩论。mergify[bot] 提示 pre-commit 失败并给出修复命令 (`uv pip install pre-commit>=4.5.1` 后 `pre-commit run --all-files`)，作者随后重新 push 并再次触发 CI。claude[bot] 仅提示本仓库配置为手动 review。tjtanaa 直接 APPROVED，无文字点评。整体决策主要由作者 Andreaskaratzas 一人完成，评审未提出替代方案或质疑。

- pre-commit 检查失败触发重新 CI (style): 作者修正格式后重新 push，并触发 Buildkite CI #84224; 后续无同类告警。
- claude[bot] 手动 review 提示 (other): 未实际触发额外 review，不影响合并流程。

- tjtanana APPROVED (other): 评审通过, PR 由作者本人合并。

风险与影响

- 风险:

1. 改动仅限测试文件, 不触碰 kernel、注意力后端或任何生产路径, 生产代码回归风险极低。
2. `reset_test_state` 依赖 `try/finally` 恢复, 若测试进程被硬杀 (如 `segfault`、`OOM killer`) 则无法执行恢复逻辑; 这是 `pytest` 测试隔离的固有边界, 可接受但值得知晓。
3. `fixture` 只保护 `test_mha_attn.py` 自身; 仓库内若还有其他测试裸改 `torch.set_default_dtype/set_default_device`, 仍可能污染后续用例。本次材料中未核实全仓同类用法, 建议后续统一收敛。
4. DSV4 测试中仍有大量 `torch.randn(...)` 未显式给出 `dtype`, 本次只修复了观察到的两个溢出构造点; 若未来默认 `dtype` 再被污染, 这些张量仍会跟随变化, 属于残留薄弱点。
 - 影响: 改动面仅两个 ROCm 相关测试文件, 不影响任何生产推理路径、CLI、模型行为或外部 API, 用户侧完全无感知。对团队的主要影响是: 消除 ROCm CI (尤其是 MI350/gfx950) 上因用例执行顺序而随机出现的 DSV4 测试失败, 提升 CI 稳定性; 同时为本仓库提供了可复用的 `reset_test_state` 测试隔离模式, 可作为后续测试编写规范参考。对 DSV4 注意力在 ROCm 上的回归保护也因显式 FP32 构造而得到补强。
 - 风险标记: 仅测试层变更, `try/finally` 无法覆盖硬崩溃, 同类 `dtype` 泄漏在其他文件仍可能残留, DSV4 测试部分张量仍依赖默认 `dtype`

关联脉络

- PR #52608 [Bugfix][CI] Release the shared ColBERT engine before `test_colbert_hf_comparison`: 同属测试状态隔离类 CI 修复: `class` 作用域 `fixture` 未及时释放显存导致后续用例失败, 与本 PR 的全局 `dtype` 泄漏是同一类跨用例污染问题。
- PR #52492 [Bugfix][DSV4] Keep indexer scoring in breakable graphs: DSV4 注意力相关 bugfix, 说明 DSV4 注意力 (含 ROCm/gfx950 路径) 是近期活跃的修复与测试线。
- PR #52550 [Config] Unify indexer cache `dtype` under `attention_config.indexer_kv_dtype`: 同为 `dtype` 管理主题: 统一 DSV4 indexer 缓存 `dtype` 配置, 与本 PR 显式锁定 `scratch` 为 FP32 的意图一致。