PR #52005 完整报告

vllm-project/vllm

[Bugfix] Fix .../mrope.py::apply_interleaved_rope() when torch.compile is used in torch==2.13

合并时间：2026-08-14 06:08

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/52005

执行摘要

- 一句话：重写 MRoPE 交错重排实现，修复 torch.compile 2.13 输出错误
- 推荐动作：值得精读。该 PR 展示了一个针对编译后端回归的等价重写手法（用索引掩码 + torch.where 替代切片赋值），以及用 eager-vs-torch.compile 对拍测试锁定数值一致性的方法。建议关注 PyTorch 上游 issue 193183 的修复，并在代码中保留 workaround 注释以便将来回退。

功能与动机

PR body 指出 vLLM 最新版本使用的 torch==2.13 下，torch.compile 对一个“只做索引和赋值”的简单函数产生了错误输出，而 torch==2.11 正常；在 524288 元素中最多 98229 个元素不匹配（18.7%），最大绝对误差 6.25。对应 PyTorch 上游 issue pytorch/pytorch#193183（2.12 开始回归，2.13 仍存在）。本 PR 通过“换一种 torch.compile 不会破坏的等价计算方式”来规避上游编译 bug，且新方式只包含索引和赋值语义，编译不应引入错误。

实现拆解

1. 定位与替换实现：在 vllm/model_executor/layers/rotary_embedding/mrope.py 中重写 apply_interleaved_rope。旧实现 x_t = x[0].clone() 后做两次切片赋值；新实现先生成 channels 索引，构造 height（余 1）与 width（余 2）的布尔掩码，再两次 torch.where 合并，避免原地切片赋值被 Inductor 错误编译。同时修正 docstring 中交错布局的描述（从 [THTHWHTHW...TT] 改为 [THWTHWTHW...TT]）。
2. 新增语义测试：tests/kernels/core/test_mrope.py 新增 test_apply_interleaved_rope，使用固定小张量（3×2×5、mrope_section=[3,1,1]）验证新实现的精确输出。
3. 新增编译一致性测试：test_apply_interleaved_rope_torch_compile 在 8192 token、bf16、CUDA/ROCm 上对比 eager 与 torch.compile(backend="inductor", fullgraph=True) 的输出，要求 rtol=0, atol=0，确保回归被锁死。
4. 端到端验证：配合 PR#51989 的 Cosmos3-Edge 修复用 vllm serve 实测，修复前模型将两图均识别为兔子，修复后正确输出“蜜蜂 + 兔子”；Buildkite CI #83713 通过。
5. 无配置、部署或 schema 配套改动。

关键文件：

- vllm/model_executor/layers/rotary_embedding/mrope.py（模块 旋转编码；类别 source；类型 core-logic；符号 apply_interleaved_rope）：核心修复文件：apply_interleaved_rope 由 clone + 切片赋值改为掩码索引 + torch.where，规避 PyTorch 2.12+ Inductor 编译回归，影响所有 MRoPE 多模态模型的编译路径。
- tests/kernels/core/test_mrope.py（模块 单元测试；类别 test；类型 test-coverage；符号 test_apply_interleaved_rope, test_apply_interleaved_rope_torch_compile）：新增两个针对性测试：固定张量的语义测试与 eager-vs-torch.compile 对拍测试，确保新实现与旧实现数值严格一致并防止回归。

关键符号：apply_interleaved_rope, test_apply_interleaved_rope, test_apply_interleaved_rope_torch_compile

关键源码片段

vllm/model_executor/layers/rotary_embedding/mrope.py

核心修复文件：apply_interleaved_rope 由 clone + 切片赋值改为掩码索引 + torch.where，规避 PyTorch 2.12+ Inductor 编译回归，影响所有 MRoPE 多模态模型的编译路径。

```
def apply_interleaved_rope(x: torch.Tensor, mrope_section: list[int]) -> torch.Tensor:
    """Apply interleaved MRoPE to 3D rotary embeddings.
    Reorganizes frequency layout from chunked [TTT...HHH...WWW] to
    interleaved [THWTHWTHW...TT], preserving frequency continuity.

    旧实现基于切片赋值，在 torch.compile + PyTorch 2.12+ 下会生成错误数值；
    这里改用索引掩码 + torch.where 的组合，编译路径与 eager 语义等价。
    """
    # 按通道位置构造布尔掩码：height 通道落在余 1 位置，width 通道落在余 2 位置
    channels = torch.arange(x.shape[-1], device=x.device)
    is_height = (channels % 3 == 1) & (channels < mrope_section[1] * 3)
    is_width = (channels % 3 == 2) & (channels < mrope_section[2] * 3)

    # 第一次 where 合并 height（其余保留 x[0]），第二次合并 width，
    # 两步合并避免原地赋值产生的跨通道依赖
    result = torch.where(is_height, x[1], x[0])
    return torch.where(is_width, x[2], result)
```

tests/kernels/core/test_mrope.py

新增两个针对性测试：固定张量的语义测试与 eager-vs-torch.compile 对拍测试，确保新实现与旧实现数值严格一致并防止回归。

```
def test_apply_interleaved_rope_torch_compile():
    # 使用接近真实 MRoPE 的配置：8192 token、bf16、CUDA/ROCm 设备
    mrope_section = [24, 20, 20]
    num_tokens = 8192
    rotary_dim = sum(mrope_section) * 2
    cache = torch.randn(3, num_tokens, rotary_dim, device=device, dtype=torch.bfloat16)
    x = cache[..., : rotary_dim // 2]

    # eager 结果作为参考基准，与 torch.compile(inductor, fullgraph) 结果严格一致
```

```
expected = apply_interleaved_rope(x, mrope_section)
compiled_fn = torch.compile(apply_interleaved_rope, backend="inductor", fullgraph=True)
result = compiled_fn(x, mrope_section)

torch.testing.assert_close(result, expected, rtol=0, atol=0)
```

评论区精华

> yzong-rh：Thanks for the fix! Was able to reproduce and confirm. pre-commit passes locally —— 维护者实测复现并确认修复有效。 claude[bot]：PR 来自 fork，自动 review 被禁用，需维护者触发人工 review —— 最终由 yzong-rh、Isotr0py 与 PyTorch 团队 zou3519 批准。 PR 评论仅有 /ci run 触发指令与社区成员的 thanks!，无设计争议或未解决疑虑。

- 修复有效性与复现确认 (testing): 修复真实有效，*eager* 与 torch.compile 输出一致，测试与 pre-commit 均通过。
- fork PR 的自动 review 策略 (other): 改由维护者（yzong-rh、Isotr0py、zou3519）人工 review 并批准，流程无阻塞。

风险与影响

- 风险：
  - 上游依赖：这是对 PyTorch 2.12+ Inductor 回归的 workaround；若上游修复（issue 193183），建议评估是否回退为更直观的切片赋值实现，代码中应保留注释便于回溯。
  - 数值一致性：新实现必须与旧 *eager* 实现严格一致。测试覆盖了 8192 token bf16 和固定小张量两种场景，但未穷举所有 mrope_section 组合与 dtype。
  - 性能影响：每次调用新增一个 *arange* 与两个布尔掩码、两次 torch.where 全量广播，相较切片赋值有少量额外计算；rotary_dim 相对 head 维度较小，预计影响可忽略，但未附 *eager* 性能对比数据。
  - 影响面：所有启用 MRoPE 且走 torch.compile/CUDA graph 路径的多模态模型（Qwen2-VL、Qwen2.5-VL、Qwen3-VL、GLM-4.1V、Cosmos3-Edge 等）均会受益；*eager* 路径行为不变。
  - 影响：对用户与系统：修复了编译路径下多模态模型输出错误 token 的问题，显著提升数值正确性；对 *eager* 用户行为无变化。对团队：需要跟踪 pytorch/pytorch#193183 上游修复进展，后续可能回收 workaround。改动集中在 MRoPE 单函数与对应测试，无 API/ 配置变化，影响范围可控，风险低。
  - 风险标记：核心路径变更，上游编译器回归 workaround，多模态模型数值影响

关联脉络

- PR #51989 Cosmos3-Edge fixes（PR body 提及，标题未提供）：PR body 明确提到端到端测试需要先应用 PR 51989 的 Cosmos3-Edge 修复，二者属于同一模型功能线的配套改动。