

PR #45073 完整报告

vllm-project/vllm

[Bugfix] Fix missing sequence_lengths in EXAONE-4.5 vision encoder

合并时间: 2026-06-10 22:44

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/45073>

执行摘要

- 一句话: 修复 EXAONE-4.5 视觉编码器 missing sequence_lengths
- 推荐动作: 建议快速合并, 属于典型的数据契约同步修复。可关注类似模式——继承 Qwen2_5_VisionTransformer 的模型 (如刚修复的 qwen3_omni_moe_thinker) 是否也有遗漏。

功能与动机

PR #42787 为 Qwen2.5-VL 视觉骨干引入 `sequence_lengths` 参数, 但 EXAONE-4.5 的重载类未更新签名, 导致继承 `Qwen2_5_VisionTransformer.forward()` 的多模态请求抛出 `TypeError: Exaone4_5_VisionBlock.forward() got an unexpected keyword argument 'sequence_lengths'`。关联 Issue #45071 报告了此问题。

实现拆解

1. `EXAONE4_5_VisionAttention.forward()` 添加参数: 在 `vllm/model_executor/models/exaone4_5.py` 的 `EXAONE4_5_VisionAttention.forward()` 签名中新增 `sequence_lengths: torch.Tensor | None = None`, 并将其传递给 `self.attn()` 调用。
2. `Exaone4_5_VisionBlock.forward()` 添加参数: 在 `Exaone4_5_VisionBlock.forward()` 签名中新增 `sequence_lengths` 参数, 并在调用 `self.attn()` 时透传。
3. 注册 `dynamic_arg_dims`: 在 `Exaone4_5_VisionBlock` 的 `@support_torch_compile` 装饰器中添加 `"sequence_lengths": 0`, 确保 `torch.compile` 正确追踪该张量的动态维度。
4. 无测试 / 配置变更: 仅修改源码文件 1 个, 无配套测试或配置改动; 作者通过预提交和手动部署验证了修复。

关键文件:

- `vllm/model_executor/models/exaone4_5.py` (模块 模型; 类别 source; 类型 core-logic; 符号 `EXAONE4_5_VisionAttention.forward`, `Exaone4_5_VisionBlock.forward`, `Exaone4_5_VisionBlock.dynamic_arg_dims`): 单个修改文件, 所有变更均在此文件中, 透传 `sequence_lengths` 参数并注册 `dynamic_arg_dims`。

关键符号: `EXAONE4_5_VisionAttention.forward`, `Exaone4_5_VisionBlock.forward`

关键源码片段

vllm/model_executor/models/exaone4_5.py

单个修改文件，所有变更均在此文件中，透传 sequence_lengths 参数并注册 dynamic_arg_dims。

```
def forward(
    self,
    x: torch.Tensor,
    cu_seqlens: torch.Tensor,
    rotary_pos_emb_cos: torch.Tensor,
    rotary_pos_emb_sin: torch.Tensor,
    max_seqlen: int | None = None,
    # 修复：添加 sequence_lengths 参数，与 PR #42787 的 Qwen2.5-VL 签名对齐
    sequence_lengths: torch.Tensor | None = None, # Only used for FlashInfer CuDNN backend
) -> torch.Tensor:
    # ... 原有逻辑 ...
    context_layer = self.attn(
        query=q, key=k, value=v,
        cu_seqlens=cu_seqlens, max_seqlen=max_seqlen,
        sequence_lengths=sequence_lengths, # 透传给底层 attention 后端
    )
    # ...
```

```
@support_torch_compile(
    dynamic_arg_dims={
        "x": 0,
        "cu_seqlens": 0,
        "sequence_lengths": 0, # 新增：确保 torch.compile 追踪该参数
        "rotary_pos_emb_cos": 0,
        "rotary_pos_emb_sin": 0,
    },
    ...
)
```

```
class Exaone4_5_VisionBlock(nn.Module):
    def forward(
        self,
        x: torch.Tensor,
        cu_seqlens: torch.Tensor,
        rotary_pos_emb_cos: torch.Tensor,
        rotary_pos_emb_sin: torch.Tensor,
        max_seqlen: int | None = None,
        seqlens: list[int] | None = None,
        sequence_lengths: torch.Tensor | None = None, # 新增：从父类 forward 链接收
    ) -> torch.Tensor:
        x_attn = self.attn(
            self.norm1(x),
            cu_seqlens=cu_seqlens,
            rotary_pos_emb_cos=rotary_pos_emb_cos,
            rotary_pos_emb_sin=rotary_pos_emb_sin,
```

```
max_seqlen=max_seqlen,  
sequence_lengths=sequence_lengths, # 透传至注意力层  
)  
# ... 其余逻辑不变
```

评论区精华

无 Review 评论；由 Isotr0py 批准合并，未触发额外讨论。

- 暂无高价值评论线程

风险与影响

- 风险：
 1. 回归风险低：变更仅添加可选参数 `sequence_lengths: torch.Tensor | None = None`，默认值为 `None`，不影响现有调用路径。
 2. `torch.compile` 兼容性：正确注册 `dynamic_arg_dims`，但若未来修改参数名需同步更新。
 3. 无测试覆盖：未添加单元测试或集成测试，但手动验证了多模态请求成功。
- 影响：
 1. 用户影响：修复 EXAONE-4.5 多模态模型在 FlashInfer CuDNN 后端下的崩溃问题，使模型可用。
 2. 系统影响：无性能或功能降级，仅极小参数传递开销。
 3. 团队影响：降低维护成本，避免继承自 Qwen2.5-VL 的代码因签名不一致而断裂。 - 风险标记：缺少测试覆盖

关联脉络

- PR #42787 Add `sequence_lengths` to Qwen2.5-VL vision backbone: 本 PR 修复的回归由 #42787 引入，需要同步更新 EXAONE-4.5 的重载类以保持签名一致。
- PR #35741 [Bugfix] Fix missing `sequence_lengths` in `qwen3_omni_moe_thinker`: 相同模式的修复，本 PR 明确引用为解决类似问题的方法。