

PR #44021 完整报告

vllm-project/vllm

[Cohere] fix RoutingMethodType

合并时间: 2026-06-06 07:25

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/44021>

执行摘要

- 一句话: 修复 Cohere MoE 路由未考虑无 renormalize 场景
- 推荐动作: 建议阅读 custom_routing_router.py 中路由类型判断逻辑, 理解 RoutingMethodType 的设计; 同时关注后续是否有其他 expert 实现补齐 Sigmoid 支持。

功能与动机

Cohere 2 MoE 模型的 router 中 renormalize 为可选配置, 原有代码未考虑该参数为 False 的情况, 导致路由类型始终被标记为 SigmoidRenorm, 可能引发 kernel 选择错误。

实现拆解

1. 在 CustomRoutingRouter.routing_method_type 中增加分支判断: 若 self.renormalize 为 True 则返回 SigmoidRenorm, 否则返回 Sigmoid。
2. 在 trtllm_bf16_moe.py 的 _supports_routing_method 白名单中添加 RoutingMethodType.Sigmoid, 保证该路由类型能被 BF16 kernel 支持。
3. 在 trtllm_nvfp4_moe.py 的 _supports_routing_method 白名单中添加 RoutingMethodType.Sigmoid, 并去除重复的 SigmoidRenorm 条目 (清理先前无意的重复注册)。

关键文件:

- vllm/model_executor/layers/fused_moe/router/custom_routing_router.py (模块 路由模块; 类别 source; 类型 core-logic; 符号 routing_method_type) : 核心逻辑修复: 路由类型判断由固定 SigmoidRenorm 改为根据 renormalize 动态选择 Sigmoid 或 SigmoidRenorm。
- vllm/model_executor/layers/fused_moe/experts/trtllm_bf16_moe.py (模块 MoE 专家; 类别 source; 类型 data-contract; 符号 _supports_routing_method) : 在 _supports_routing_method 白名单中添加 Sigmoid 和 SigmoidRenorm, 确保 BF16 kernel 可处理新的路由类型。
- vllm/model_executor/layers/fused_moe/experts/trtllm_nvfp4_moe.py (模块 MoE 专家; 类别 source; 类型 data-contract; 符号 _supports_routing_method) : 同步添加 Sigmoid 支持, 同时清理重复的 SigmoidRenorm 条目。

关键符号: routing_method_type, _supports_routing_method

关键源码片段

vllm/model_executor/layers/fused_moe/router/custom_routing_router.py

核心逻辑修复：路由类型判断由固定 SigmoidRenorm 改为根据 renormalize 动态选择 Sigmoid 或 SigmoidRenorm。

```
@property
def routing_method_type(self) -> RoutingMethodType:
    from vllm.model_executor.models.cohere2_moe import token_choice_with_bias
    from vllm.model_executor.models.llama4 import Llama4MoE

    # NOTE: FLASHINFER_TRTLLM 支持 Llama4 路由。
    if self.custom_routing_function == Llama4MoE.custom_routing_function:
        return RoutingMethodType.Llama4
    # Cohere MoE 使用 sigmoid -> top-k, 可选择性地后接 renormalize。
    if self.custom_routing_function == token_choice_with_bias:
        # 根据 renormalize 标志选择对应的路由类型
        if self.renormalize:
            return RoutingMethodType.SigmoidRenorm
        return RoutingMethodType.Sigmoid
    return RoutingMethodType.Custom
```

vllm/model_executor/layers/fused_moe/experts/trtllm_bf16_moe.py

在 _supports_routing_method 白名单中添加 Sigmoid 和 SigmoidRenorm，确保 BF16 kernel 可处理新的路由类型。

```
@staticmethod
def _supports_routing_method(
    routing_method: RoutingMethodType,
    weight_key: QuantKey | None,
    activation_key: QuantKey | None,
) -> bool:
    return routing_method in [
        RoutingMethodType.DeepSeekV3,
        RoutingMethodType.Llama4,
        RoutingMethodType.Renormalize,
        RoutingMethodType.RenormalizeNaive,
        RoutingMethodType.SigmoidRenorm, # 新增：支持有 renormalize 的路由
        RoutingMethodType.Sigmoid, # 新增：支持无 renormalize 的路由
    ]
```

评论区精华

无 reviewer 实质性讨论，仅 mergify 提示合并冲突，最终通过 rebase 解决。

- 暂无高价值评论线程

风险与影响

- 风险：变更主要集中在路由类型判断与 kernel 白名单注册，影响范围限于使用了 token_choice_with_bias 路由函数的 Cohere MoE 模型。若其他模型也复用了 CustomRoutingRouter 且 renormalize 默认值为 True，则行为未变；若之前错误地忽略了 renormalize=False 的场景，则本次修复可避免错误使用 SigmoidRenorm kernel。目前无测试配套，需依赖已有集成测试覆盖。
- 影响：对用户：Cohere 模型推理时若设置 renormalize=False，可获得正确的路由 kernel，可能改善性能或避免潜在错误。对系统：新增 RoutingMethodType.Sigmoid 枚举值，kernel 注册表扩展，无破坏性影响。对团队：需要确保其他 MoE 实现（如模块化 expert）也同步支持 Sigmoid，但目前仅 monolithic kernel 支持。
- 风险标记：路由类型变更，无测试配套，核心 MoE 路径

关联脉络

- 暂无明显关联 PR