PR #44347 完整报告

vllm-project/vllm

[Bugfix] Update TrtLLM MoE routing methods

合并时间：2026-06-03 17:56

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/44347

执行摘要

- 一句话：修复 TrtLLM MoE 路由方法分类及 dtype 检查
- 推荐动作：建议尽快合入，以修复 CI 失败和模型兼容性问题。该 PR 展现了精细的路由方法分类调整，可精读 get_routing_method_type 的决策树逻辑，了解不同模型的路由模式。

功能与动机

修复 PR#43859 引入的回归，该变更导致 nvidia/NVIDIA-Nemotron-3-Nano-30B-A3B-FP8 模型在使用 FLASHINFER_TRTLLM 后端时失败，报错："FP8 MoE backend FLASHINFER_TRTLLM does not support the deployment configuration since kernel does not support quantization scheme..."。同时需要更新路由方法与 flashinfer 对应版本保持一致，并修正 Step-3.7-Flash 模型的路由分类。

实现拆解

1. RoutingMethodType 枚举更新：在 vllm/model_executor/layers/fused_moe/config.py 中新增 Sigmoid = (8,) 枚举值，代表 Sigmoid -> TopK（无归一化）；将 Unspecified 编号从 8 改为 9；更新各枚举注释以更精确地与 flashinfer 对应。
2. get_routing_method_type逻辑重构：该函数新增routed_scaling_factor参数（默认1.0）。has_e_score_bias 分支改为先检查 scoring_func == "sigmoid"，再检查 renormalize、num_expert_group 和 routed_scaling_factor 以区分 DeepSeekV3、MiniMax2 和 Unspecified；scoring_func == "sigmoid" 分支移除 top_k == 1 的特殊处理，统一返回 SigmoidRenorm（若 renormalize）或 Sigmoid；这些变更防止了 Step-3.7-Flash 被误判。
3. _supports_router_logits_dtype 修复：在 trtllm_fp8_moe.py 和 trtllm_nvfp4_moe.py 中，将该方法从“拒绝 float32”改为“允许 bfloat16 和 float32”，以支持 Nemotron-3-Nano 等模型的 float32 router_logits。
4. 移除 e_score_correction_bias 转换：在 _apply_per_tensor 和 apply 方法中删除将 e_score_correction_bias 转换为 bfloat16 的代码，因为 flashinfer 已修复相关 issue（#2909），不再需要此 workaround。
5. routed_scaling_factor 参数传递：在 fused_topk_bias_router.py、grouped_topk_router.py、zero_expert_router.py 中，将 routing_method_type 属性调用 get_routing_method_type 时传入 routed_scaling_factor，使路由分类决策能够感知缩放因子。

6. _supports_routing_method 扩展：在两个 expert 文件中，将新增的 RoutingMethodType.Sigmoid 加入支持的路由方法列表。

关键文件：

- vllm/model_executor/layers/fused_moe/config.py（模块 MoE 配置；类别 source；类型 core-logic；符号 RoutingMethodType, get_routing_method_type）：核心配置文件，新增 Sigmoid 枚举值，重写 get_routing_method_type 决策逻辑并新增 routed_scaling_factor 参数，直接影响所有模型的路由分类。
- vllm/model_executor/layers/fused_moe/experts/trtllm_fp8_moe.py（模块 MoE 专家；类别 source；类型 core-logic；符号 _supports_router_logits_dtype, _supports_routing_method, _apply_per_tensor）：修复 _supports_router_logits_dtype，移除 e_score_correction_bias 转换，将 Sigmoid 加入支持路由列表。
- vllm/model_executor/layers/fused_moe/experts/trtllm_nvfp4_moe.py（模块 MoE 专家；类别 source；类型 core-logic；符号 _supports_router_logits_dtype, apply）：与 trtllm_fp8_moe.py 相同的修复：_supports_router_logits_dtype 和移除 e_score_correction_bias 转换。
- vllm/model_executor/layers/fused_moe/router/fused_topk_bias_router.py（模块 MoE 路由；类别 source；类型 data-contract；符号 routing_method_type）：在 routing_method_type 属性中传递 routed_scaling_factor 参数。
- vllm/model_executor/layers/fused_moe/router/grouped_topk_router.py（模块 MoE 路由；类别 source；类型 data-contract；符号 routing_method_type）：与 fused_topk_bias_router.py 相同的参数传递变更。
- vllm/model_executor/layers/fused_moe/router/zero_expert_router.py（模块 MoE 路由；类别 source；类型 data-contract；符号 routing_method_type）：与以上两个 router 相同的参数传递变更。

关键符号：get_routing_method_type, _supports_router_logits_dtype, _supports_routing_method, _apply_per_tensor, apply, routing_method_type

关键源码片段

vllm/model_executor/layers/fused_moe/config.py

核心配置文件，新增 Sigmoid 枚举值，重写 get_routing_method_type 决策逻辑并新增 routed_scaling_factor 参数，直接影响所有模型的路由分类。

```
# 注释已按盘古排版规则添加
class RoutingMethodType(IntEnum):
    # ... 其他枚举值 ...
    MiniMax2 = (7,) # Sigmoid + Bias -> TopK -> ScaledSumNormalize (routeScale=1.0, epsilon=1e-20)
    Sigmoid = (8,) # Sigmoid -> TopK (no renormalization) — 新增枚举
    Unspecified = (9,) # 编号从 8 改为 9
    # ... DeepseekV4, Custom, Simulated ...

def get_routing_method_type(
    scoring_func: str,
```

```
    top_k: int,
    renormalize: bool,
    num_expert_group: int | None,
    has_e_score_bias: bool,
    routed_scaling_factor: float | None = 1.0, # 新增参数，默认 1.0
) -> RoutingMethodType:
    # ... 处理 sqrtsoftplus ...

    if has_e_score_bias:
        # 原来直接用 num_expert_group 区分 DeepSeekV3 和 MiniMax2
        # 改为更严格的检查：先确认 renormalize，再根据 num_expert_group 和 scaling_factor 判定
        if scoring_func == "sigmoid":
            if not renormalize:
                return RoutingMethodType.Unspecified
            if (num_expert_group or 0) > 0:
                return RoutingMethodType.DeepSeekV3
            if routed_scaling_factor in (None, 1.0):
                return RoutingMethodType.MiniMax2
            # routed_scaling_factor 非 1.0 时返回 Unspecified，防止误匹配
            return RoutingMethodType.Unspecified
        else:
            return RoutingMethodType.Unspecified

    if scoring_func == "sigmoid":
        # 移除原来的 top_k == 1 分支（导致 Llama4 匹配丢失）
        # 统一根据 renormalize 判定：有 renormalize 则 SigmoidRenorm，否则 Sigmoid
        if renormalize:
            return RoutingMethodType.SigmoidRenorm
        # 没有 renormalize 的 sigmoid 路由现在返回 Sigmoid，而非 Unspecified
        return RoutingMethodType.Sigmoid

    # ... 处理 softmax ...
    return RoutingMethodType.Unspecified
```

评论区精华

本 PR 无 review 评论。唯一审核人 jeejeelee 直接批准（APPROVED），无讨论。

- 暂无高价值评论线程

风险与影响

- 风险：
  1. 回归风险：_supports_router_logits_dtype 从拒绝 float32 改为同时接受 bfloat16 和 float32，可能使得某些本应被拒绝的 float32 配置意外通过，但 PR body 的测试结果覆盖了多种模型和量化方案，降低了风险。
  2. 兼容性风险：新增 Sigmoid 枚举值并调整 Unspecified 编号（8->9）可能影响序列化或依赖枚举数值的模块；但由于枚举值在 flashinfer 侧也有对应定义，且测试覆盖了主要模

型，风险可控。

3. 逻辑变更：get_routing_method_type 的条件判断变化可能影响其他未测试的模型路由分类，需关注后续 issue。 - 影响：直接影响使用 TrtLLM MoE 后端的模型，特别是 Nemotron-3-Nano、DeepSeek R1、MiniMax M2、Step-3.7-Flash 等。修复了 FP8 和 NVFP4 模型的路由兼容性；增强了路由分类的准确性。对使用其他 MoE 后端（如 cuBLAS）的模型无影响。影响范围限定在 MoE 路由模块，涉及 6 个文件，变更量小（+24/-24）。 - 风险标记：回归修复，枚举值编号变更，逻辑分支重构

关联脉络

- PR #43859 [some PR that introduced regression]: 本 PR 明确 revert 了 PR#43859 中 _supports_router_logits_dtype 的变更，该变更导致了 Nemotron-3-Nano FP8 模型的 CI 失败。