

PR #31107 完整报告

sgl-project/sglang

[NPU] Determine the topk norm_type through scoring_func

合并时间: 2026-07-15 15:26

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/31107>

执行摘要

- 一句话: NPU MoE topk norm_type 根据 scoring_func 动态选择, 修复精度
- 推荐动作: 该 PR 为 pinpoint bugfix, 改动虽小但修复了重要精度问题。建议相关开发者关注 NPU MoE topk 路径对 scoring_func 的依赖, 并在新增模型时显式指定 scoring_func。值得阅读以了解 NPU topk 的配置方式。

功能与动机

PR #29509 重构了 topk 逻辑, 但将 NPU 端的 norm_type 固定为 1 (sigmoid), 这影响了使用 softmax 的模型 (如 ds coder v2 lite instruct) 的精度。需要根据每个模型的 scoring_func 正确设置 norm_type。

实现拆解

1. 修改 NPU topk kernel 的 norm_type 逻辑: 在 python/sglang/srt/hardware_backend/npu/moe/topk.py 的 fused_topk_npu 函数中, 将第 100 行的 norm_type=1 改为 norm_type=(0 if topk_config.scoring_func == "softmax" else 1)。这样, 当 scoring_func 为 softmax 时 norm_type 为 0, 否则为 1 (sigmoid), 与模型配置对齐。
2. 在 GLM4 MoE Lite 模型中明确 scoring_func: 在 python/sglang/srt/models/glm4_moe_lite.py 的 __init__ 方法中, 构造专家模块时传入 scoring_func="sigmoid" (原为硬编码缺失), 确保该模型的 topk 行为与之前一致。
3. 无测试 / 配置 / 部署配套变更: 本次改动仅涉及两行源码, 未新增测试, 依赖已有 CI 验证精度。

关键文件:

- python/sglang/srt/hardware_backend/npu/moe/topk.py (模块 NPU 内核; 类别 source; 类型 core-logic): 核心修复文件: 修改 fused_topk_npu 函数中 norm_type 的赋值逻辑, 由固定 1 改为根据 scoring_func 动态决定。
- python/sglang/srt/models/glm4_moe_lite.py (模块 模型定义; 类别 source; 类型 data-contract): 模型配置修复: 在构造专家模块时显式传入 scoring_func='sigmoid', 确保该模型 (默认使用 sigmoid) 行为不变。

关键符号: fused_topk_npu, init

关键源码片段

python/sglang/srt/models/glm4_moe_lite.py

模型配置修复：在构造专家模块时显式传入 `scoring_func='sigmoid'`，确保该模型（默认使用 `sigmoid`）行为不变。

```
# python/sglang/srt/models/glm4_moe_lite.py (行 228)
self.experts = get_moe_impl_class(quant_config)(
    ...
    routed_scaling_factor=self.routed_scaling_factor,
    prefix=add_prefix("experts", prefix),
    **({"scoring_func": "sigmoid"}), # 显式指定 scoring_func 为 sigmoid
)
```

评论区精华

PR 的 review 讨论较少，主要来自 bot 和合并者。关键点：

- `sglang-npu-bot` 请求 `zRzRzRzRzRzRzR` 确认模型的修改符合预期，指出目前已知的 GLM5.1、GLM5.2、Flash 等模型 `scoring_func` 均为 `sigmoid`。
- CI 不稳定导致 glm 测试失败，后来通过 PR #31289 调整了测试阈值，最终合并。
- GLM 模型 `scoring_func` 确认 (question): 确认这些模型 `scoring_func` 均为 `sigmoid`，因此只需在 GLM4 MoE Lite 中显式传入 `sigmoid` 即可。
- CI 测试稳定性 (testing): CI 已通过并合并。

风险与影响

- 风险：
 1. 回归风险：修改仅影响 NPU 后端 MoE topk 路径，且只改变 `norm_type` 的赋值方式，逻辑清晰，回归风险低。但若未来新增其他 `scoring_func` 类型（如 `relu`），需要同步扩展此判断逻辑。
 2. 模型精度风险：对于未在 PR 中明确指定 `scoring_func` 的模型，其默认行为取决于 `TopKConfig` 的初始化值。当前修改保证了 `scoring_func` 为 `softmax` 时正确使用 0，否则使用 1，与原有硬编码 1 (`sigmoid`) 的兼容性需额外验证。已在 GLM4 MoE Lite 中显式传入 `sigmoid`，但其他 `sigmoid` 模型（如 GLM5.x）未显式传入，依赖默认配置，建议追踪确认。
 3. 性能风险：无，仅改变一个参数值。
- 影响：
 - 用户影响：修复了 NPU 上 `ds coder v2 lite instruct` 的精度问题（准确率提升至 81%）。所有使用 NPU MoE 且 `scoring_func` 为 `softmax` 的模型都将受益。
 - 系统影响：无。
 - 团队影响：维护者需确保未来添加新模型时正确设置 `scoring_func`，避免回归。
 - 风险标记：缺少测试覆盖，隐式依赖默认 `scoring_func`

关联脉络

- PR #29509 refactor topk part for npu: 本 PR 直接修复了 #29509 引入的 norm_type 硬编码问题。
- PR #31289 [CI] Lower GLM-5.2 NVFP4 MTP speed threshold: 与本 PR 相关的 CI 稳定性修复，确保测试通过。