

PR #26980 完整报告

sgl-project/sglang

[fix] Skip routed expert capture for draft model under spec v2

合并时间: 2026-06-25 07:56

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/26980>

执行摘要

- 一句话: 修复 draft 模型污染 target 路由专家捕获
- 推荐动作: 对于涉及 speculative decoding 或 MoE draft 模型的开发者, 建议仔细阅读此 PR 的设计: 如何通过配置标志实现跨后端的统一捕获门控, 以及在初始化早期阻断副作用的时机选择。对于使用者, 应升级以修复相关 bug。建议未来补充自动化测试以确保回归覆盖。

功能与动机

症状: 启用 `--enable-return-routed-experts` 且 MoE draft 模型时, 返回的 `routed_experts` 与仅 target 的基准不一致。根因: `RoutedExpertsCapturer` 是进程全局单例, draft 与 target 共享同一缓冲区, draft 的 MoE TopK 会写入 target 的捕获缓冲区。PR 描述中明确提出需要保护 target 的捕获不受 draft 影响。

实现拆解

1. 添加配置标志: 在 `TopKConfig` 中新增 `allow_routed_experts_capture` 字段 (默认 True), 用于标识该 TopK 是否允许触发路由专家捕获。
2. 提取统一捕获入口: 在 `layers/moe/topk.py` 中新增 `capture_routed_experts_if_allowed` 函数, 将所有后端 (CUDA `_post_process_topk_ids`、NPU `fused_topk_npu`) 的直接捕获调用替换为该函数, 确保捕获行为受标志控制。
3. 新建 draft 阻断函数: 在 `state_capturer/routed_experts.py` 中新增 `disable_routed_experts_capture_for_draft`, 遍历模型模块将每个 TopK 实例的 `allow_routed_experts_capture` 设为 False。
4. 集成到 ModelRunner: 在 `ModelRunner.initialize` 中, 对 draft worker 在 `backend/graph` 初始化之前调用 `disable_routed_experts_capture_for_draft`, 同时修改 `init_routed_experts_capturer` 和 `forward` 中的判断, 确保 draft worker 跳过捕获器的初始化和 `on_forward_end` 调用。
5. 适配 BYPASS 路径: 在 `fused_moe_triton/layer.py` 的 BYPASS 路径中, 将 `allow_routed_experts_capture` 标志通过函数参数传递, 确保重建的 `TopKConfig` 保留正确的捕获状态。

关键文件:

- python/sglang/srt/layers/moe/topk.py (模块 MoE 路由; 类别 source; 类型 core-logic; 符号 TopKConfig, capture_routed_experts_if_allowed) : 核心修改: 新增 TopKConfig.allow_routed_experts_capture 标志和统一的捕获入口函数 capture_routed_experts_if_allowed, 所有后端通过此函数进行捕获, 实现标志门控。
- python/sglang/srt/state_capturer/routed_experts.py (模块 专家捕获; 类别 source; 类型 entrypoint; 符号 disable_routed_experts_capture_for_draft) : 新增 disable_routed_experts_capture_for_draft 函数, 遍历模型模块将 TopK 实例的 allow_routed_experts_capture 设为 False, 是 draft 侧关闭捕获的关键入口。
- python/sglang/srt/model_executor/model_runner.py (模块 模型执行; 类别 source; 类型 data-contract) : 在 ModelRunner.initialize 中对 draft worker 调用 disable_routed_experts_capture_for_draft, 在 backend 初始化前完成阻断; 同时修改 init_routed_experts_capturer 和 forward 中的 draft 判断。
- python/sglang/srt/hardware_backend/npu/moe/topk.py (模块 NPU 路由; 类别 source; 类型 dependency-wiring) : 将 NPU 后端的直接捕获调用替换为 capture_routed_experts_if_allowed, 确保 NPU 路径也受标志控制。
- python/sglang/srt/layers/moe/fused_moe_triton/layer.py (模块 融合 MoE; 类别 source; 类型 core-logic) : 在 BYPASS 路径中传递 allow_routed_experts_capture 标志, 确保重建的 TopKConfig 携带正确的捕获状态。

关键符号: capture_routed_experts_if_allowed, disable_routed_experts_capture_for_draft

关键源码片段

python/sglang/srt/layers/moe/topk.py

核心修改: 新增 TopKConfig.allow_routed_experts_capture 标志和统一的捕获入口函数 capture_routed_experts_if_allowed, 所有后端通过此函数进行捕获, 实现标志门控。

```
@dataclass
class TopKConfig:
    top_k: int
    # ... 其他已有字段 ...
    # Draft-side MoE blocks set this False so they never write the target's
    # process-global routed-experts capture buffer.
    allow_routed_experts_capture: bool = True
```

```
def capture_routed_experts_if_allowed(
    topk_config: TopKConfig,
    layer_id: Optional[int],
    topk_ids: torch.Tensor,
) -> None:
    """所有后端的单一捕获入口, 受到每个配置的退出选项控制。"""
    if not topk_config.allow_routed_experts_capture:
        return
    if (cap := get_global_experts_capturer()) is not None:
        cap.capture(layer_id=layer_id, topk_indices=topk_ids)
```

评论区精华

Review 评论为空，PR 作者在描述中提出了三个审查焦点：确认 `capture_routed_experts_if_allowed` 是唯一捕获站点、检查 BYPASS 路径中标志的正确传递、确保 `disable_routed_experts_capture_for_draft` 在 graph/backend 初始化前运行。未产生实质性讨论。

- 确认捕获入口的唯一性 (design): 代码中已将所有后端的捕获调用替换为该函数，保证唯一入口。

风险与影响

- 风险：主要风险在于新增的 `allow_routed_experts_capture` 标志可能被某些后端或自定义 TopK 实现绕过，例如若存在直接调用全局捕获器 `cap.capture()` 的路径未迁移到统一入口，则标志无效。BYPASS 路径需确保 `allow_routed_experts_capture` 正确传递，否则可能导致行为不一致。另外，测试文件在最终 commit 中被移除 (commit 4bec9f7)，缺少自动化回归覆盖，潜在回归需依赖手动测试或后续补加测试。
- 影响：影响用户：修复了使用 MoE draft 模型（如 DeepSeek-V3 MTP、Qwen3-MoE EAGLE3）且启用 `--enable-return-routed-experts` 时路由专家返回不准确的 bug。影响系统：添加了轻量布尔判断和一次模型遍历，性能影响可忽略。影响团队：为未来类似跨模型副作用阻断提供了可复用的模式（配置标志 + 统一入口 + 初始化早期阻断）。
- 风险标记：测试已移除，BYPASS 路径易出错，跨后端一致性

关联脉络

- 暂无明显关联 PR