PR #36124 完整报告

sgl-project/sglang

[AMD] Quark shared-experts gate: recognise a trailing MTP layer

合并时间：2026-08-24 15:53

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/36124

执行摘要

- 一句话：修复 Quark 门禁漏判尾部 MTP 层，恢复 GLM-5.2-MXFP4 共享专家融合
- 推荐动作：值得精读：改动虽小，但触及量化门禁的核心判别逻辑，且性能收益显著、无测试配套是明确的改进点。关注两个设计决策：一是把 draft 层识别收敛为 _is_draft_layer() 单一入口，与既有 MTP 层区间遍历逻辑保持一致；二是用 int(getattr(..., 0) or 0) 在赋值处统一归一 None，避免在多个使用点重复防御。建议后续补 _is_draft_layer 的单元测试。

功能与动机

PR body 说明：can_fuse_shared_expert() 只按 mtp. 前缀识别 MTP draft stack，而 GLM-5.2-MXFP4 将 draft 层拼写为 model.layers.78（num_hidden_layers=78），于是 3 个被排除的 draft 投影否决了全部 78 层目标模型的共享专家融合；同时 from_config() 的 prequantized 分支从未转发 hf_config，导致层数信息不可用。失融合的代价不止 fused shared expert 本身，还会让 aiter fused_moe 的 key 从 257 experts/topk 9 退化为 256/8，并使 tuned FlyDSL config 从命中变为 miss（启发式 fallback）。回归由 #35200 引入（在 load-time-override 移除后恢复了 gate）。

实现拆解

1. 提取 draft-stack 计数：在 QuarkConfig.__init__中新增 self.num_hidden_layers 与 self.num_nextn_predict_layers，从 hf_config 读取；为容忍 None 值，用 int(getattr(..., 0) or 0) 强制归一，避免 range(None) 抛 TypeError。
2. 修复 from_config() 的 prequantized 分支：此前构造 QuarkConfig 时未传 hf_config，层数永远不可用；现在改为从 config.get("hf_config") 透传，与 requantization 分支保持一致。
3. 新增 _is_draft_layer() 判别方法：同时识别两种 checkpoint 拼写——mtp. 前缀（Qwen3.5 风格）与主 decoder 末尾追加的 model.layers.[num_hidden_layers, num_hidden_layers + num_nextn_predict_layers) 区间（GLM-5.2 风格），并注明该区间与 MTP 模型中 get_spec_layer_idx_from_weight_name() 遍历的范围一致。
4. 替换门禁逻辑：can_fuse_shared_expert() 中的 not layer.startswith("mtp.") 改为 not self._is_draft_layer(layer)，使尾部 MTP 层不再否决目标模型的共享专家融合。 测试与部署配套：本次未新增测试文件；验证依赖 PR body 中的 MI355X TP4 性能对比（Docker0820 vs 本 PR），并以融合开关为唯一变量。

关键文件：

- python/sglang/srt/layers/quantization/quark/quark.py（模块 量化配置；类别 source；类型 core-logic；符号 _is_draft_layer）：唯一变更文件，承载门禁修复全部逻辑：新增 _is_draft_layer()、__init__提取层数、from_config() 透传 hf_config、can_fuse_shared_expert() 切换判别入口，直接决定 AMD 上 GLM-5.2-MXFP4 共享专家融合是否生效。

关键符号：_is_draft_layer, can_fuse_shared_expert, QuarkConfig.from_config, QuarkConfig.init

关键源码片段

python/sglang/srt/layers/quantization/quark/quark.py

唯一变更文件，承载门禁修复全部逻辑：新增 _is_draft_layer()、__init__提取层数、from_config() 透传 hf_config、can_fuse_shared_expert() 切换判别入口，直接决定 AMD 上 GLM-5.2-MXFP4 共享专家融合是否生效。

```
# python/sglang/srt/layers/quantization/quark/quark.py
# QuarkConfig.__init__ 中提取 draft-stack 层数，供 _is_draft_layer()
# 区分“追加的 MTP/NextN 层”与“目标模型自身层”。
# “无 draft stack” 在 HF 配置里常显式写成 None，与字段缺失同样常见，
# 因此用 int(.. or 0) 强制归一为 0，避免后续构造 range(None) 抛 TypeError。
self.num_hidden_layers = getattr(hf_config, "num_hidden_layers", None)
self.num_nextn_predict_layers = int(
    getattr(hf_config, "num_nextn_predict_layers", 0) or 0
)

def _is_draft_layer(self, layer: str) -> bool:
    """判断被排除的层是否属于 MTP/NextN draft stack。

    draft 层在多数 checkpoint 中都被排除量化，它不反映目标模型
    自身如何存储 shared experts，因此不应否决融合决策。
    checkpoint 拼写有两种：带 mtp. 前缀（如 Qwen3.5），或追加在
    主 decoder 末尾、落在区间
    model.layers.[num_hidden_layers, num_hidden_layers + num_nextn_predict_layers),
    与 MTP 模型 get_spec_layer_idx_from_weight_name() 遍历的范围一致。
    """
    if layer.startswith("mtp."):
        return True
    if self.num_hidden_layers is None:
        return False
    base = self.num_hidden_layers
    return any(
        layer.startswith(f"model.layers.{base + i}.")
        for i in range(self.num_nextn_predict_layers)
    )

def can_fuse_shared_expert(self) -> bool:
```

```
    # shared_expert 主体被排除量化，门禁不能因此否决融合；
    # 但 draft 层的 "shared_expert" 排除项要先剔除。
    if any(
        "shared_expert" in layer
        and "shared_expert_gate" not in layer
        and not self._is_draft_layer(layer)
        for layer in self.exclude_layers
    ):
        return False
    # 后续逻辑：无 per-layer 配置则直接可融合；否则对比 layer 0
    # 的 routed experts 与 shared experts 量化规格，规格不一致或
    # 名字无法匹配（ValueError）时都不能融合。
    ...
```

评论区精华

PR 无 inline review 评论，HaiShaw 直接 APPROVED。核心设计权衡记录在 commit 历史中：作者补交 tolerate a None nextn count，指出 ModelConfig 将 num_nextn_predict_layers 默认为 None，checkpoint 显式携带 null 与字段缺失等价，getattr(..., 0) 只覆盖缺席场景，导致 range(None) 抛 TypeError；结论是在赋值处统一 int(.. or 0) 强制转换，而不是在使用处逐个加保护。

- num_nextn_predict_layers 为 None 时 range(None) 崩溃 (correctness): 在赋值处用 int(getattr(..., 0) or 0) 强制转换，而不是在各使用点逐个加保护。
- 无 review 评论，HaiShaw 直接 APPROVED (other): 设计决策通过 commit message 自述完成闭环。

风险与影响

- 风险： 1) 测试缺口：改动未配套单元测试，_is_draft_layer() 依赖命名约定（mtp. 前缀 + 数字区间），未来第三种拼写会静默误判。 2) 配置键依赖：prequantized 分支依赖 config 中是否存在 hf_config 键，键缺失时 num_hidden_layers 为 None，判别退化为仅前缀逻辑，GLM-5.2 类模型会静默失去优化而非报错。 3) 融合决策改变：fused_moe key 从 256/8 变为 257/9，TPOT -16% 的收益未附逐 token 数值一致性对比，存在精度回归风险。 4) 区间逻辑耦合：_is_draft_layer() 的层数区间与 get_spec_layer_idx_from_weight_name() 的遍历逻辑需保持同步维护。
  - 影响：直接影响 AMD MI355X 上 GLM-5.2-MXFP4 TP4 推理：conc4 下 Mean TTFT -6.0%、Median TPOT -16.0%，conc64 下 Mean TTFT -5.7%、Median TPOT -8.4%。对同样把 draft 层追加在主 decoder 末尾的 Quark checkpoint 同样生效；Qwen3.5 等 mtp. 前缀模型与无 draft 层模型行为不变。代码侧影响 sglang/srt 量化子系统，团队后续维护 Quark 门禁时需同步关注命名约定。
  - 风险标记：核心路径变更，缺少测试覆盖，静默回退路径，配置键依赖

关联脉络

- PR #35200 （PR body 提及的引入 PR，标题未在本次材料中提供）：PR body 明确说明：本次修复的 shared-expert gate 由 #35200 引入（在 load-time-override 移除后恢复），是本次回归的源头。