PR #29509 完整报告

sgl-project/sglang

[NPU]GLM-4.7-Flash optimize with fused kernels

合并时间：2026-06-30 19:22

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/29509

执行摘要

- 一句话：NPU GLM-4.7 MLA 与 MoE 融合算子和配置优化
- 推荐动作：不推荐当前状态合并。需先修复 reviewer 指出的两个 NameError 问题，并补充测试覆盖 Context Parallel 和 q_lora_rank 为 None 的场景。PR 的融合算子思路正确，但实现安全性和完整度需加强。

功能与动机

PR body 明确说明 'Introduce a fused Triton kernel to improve model performance'，并提供了精度对比（前 / 后一致）和速度测试截图（显示显著加速）。

实现拆解

1. MLA 预处理路径优化(deepseek_v2_attention_mla_npu.py): 在非 AG 非 Context Parallel 且序列长度小于 65536 的条件下，调用新引入的融合算子 fused_split_qk_norm，将原本分离的 split、q_a_layernorm、kv_a_layernorm 合并为一个核，减少内核启动开销和显存带宽占用。
2. MoE top-k norm_type 固定(topk.py): 将 norm_type 从动态判断 (0 if topk_config.scoring_func == 'softmax' else 1) 简化为直接固定为 1（sigmoid），简化控制流并利用固定配置的优化路径。
3. 重构 else 分支(deepseek_v2_attention_mla_npu.py): 重写 m.q_lora_rank is not None 分支，分离 AG 场景与非 AG 场景，在非 AG 条件下优先使用融合算子，否则 fallback 到原始 split+norm 路径。

关键文件：

- python/sglang/srt/hardware_backend/npu/modules/deepseek_v2_attention_mla_npu.py（模块 NPU MLADE；类别 source；类型 core-logic）：核心变更文件：引入 fused_split_qk_norm 融合算子，重构 MLA 预处理控制流。
- python/sglang/srt/hardware_backend/npu/moe/topk.py（模块 MoE Gating；类别 source；类型 core-logic）：MoE top-k 配置简化：固定 norm_type 为 1，移除动态判断。

关键符号：forward_mla_prepare_npu, fused_topk_npu, fused_split_qk_norm

关键源码片段

python/sglang/srt/hardware_backend/npu/modules/deepseek_v2_attention_mla_npu.py

核心变更文件：引入 fused_split_qk_norm 融合算子，重构 MLA 预处理控制流。

```
# python/sglang/srt/hardware_backend/npu/modules/deepseek_v2_attention_mla_npu.py
# forward_mla_prepare_npu 函数片段（head 版本，省略 AG 与 CP 场景分支）
if m.q_lora_rank is not None:
    qkv_latent = get_attn_tp_context().fetch_qkv_latent()
    if not dsa_use_prefill_cp(forward_batch) and qkv_latent.shape[0] < 65536:
        # 融合算子：一次 kernel 完成 split + q_a_layernorm + kv_a_layernorm
        # 减少内存读写和 kernel launch 开销
        q, k_nope, k_pe = fused_split_qk_norm(
            qkv_latent,
            m.q_a_layernorm,
            m.kv_a_layernorm,
            m.q_lora_rank,
            m.kv_lora_rank,
            m.qk_rope_head_dim,
            eps=m.q_a_layernorm.variance_epsilon,
        )
    else:
        # 原 fallback：split + 两次 layernorm
        q, latent_cache = qkv_latent.split([m.q_lora_rank, ...], dim=-1)
        q = m.q_a_layernorm(q)
        k_nope = m.kv_a_layernorm(latent_cache[..., :m.kv_lora_rank]).unsqueeze(1)
        k_pe = latent_cache[..., m.kv_lora_rank:].unsqueeze(1)
    q_lora = q if m.use_dsa else None
    q = m.q_b_proj(q)[0].view(-1, m.num_local_heads, m.qk_head_dim)
else:
    # q_lora_rank 为 None 的分支：同样需要保证 k_pe 被定义
    q = m.q_proj(hidden_states)[0].view(-1, m.num_local_heads, m.qk_head_dim)
    latent_cache = m.kv_a_proj_with_mqa(hidden_states)[0]
    k_nope = m.kv_a_layernorm(latent_cache[..., :m.kv_lora_rank]).unsqueeze(1)
    k_pe = latent_cache[..., m.kv_lora_rank:].unsqueeze(1) # 新增，修复 NameError
```

python/sglang/srt/hardware_backend/npu/moe/topk.py

MoE top-k 配置简化：固定 norm_type 为 1，移除动态判断。

```
# python/sglang/srt/hardware_backend/npu/moe/topk.py
# fused_topk_npu 函数中的 npu_moe_gating_top_k 调用片段（head 版本）
topk_weights, topk_ids, _ = torch.ops.npu.npu_moe_gating_top_k(
    router_logits.to(torch.float32),
    k=topk_config.top_k,
    bias=(correction_bias.to(torch.float32) if correction_bias is not None else None),
    k_group=topk_config.topk_group if use_grouped_topk else 1,
    group_count=topk_config.num_expert_group if use_grouped_topk else 1,
    group_select_mode=(1 if use_grouped_topk else 0),
    renorm=0,
    # 1 for sigmoid, 0 for softmax
```

```
    norm_type=1, # 固定 1，移除动态判断（之前为 (0 if scoring_func=="softmax" else 1)）
    routed_scaling_factor=(1 if renormalize else topk_config.routed_scaling_factor),
    eps=1e-20,
)
```

评论区精华

Reviewer（gemini-code-assist[bot]）指出两个 高优先级问题：

1. 当 Context Parallel 启用时（dsa_use_prefill_cp 为 True），融合分支不会定义 latent_cache，后续 m.rebuild_cp_kv_cache 使用时会触发 NameError。当前状态：未解决。
2. 当 m.q_lora_rank is None 时，外部作用域的 k_pe 定义被移除，导致 m.rotary_emb 调用 k_pe 时 NameError。当前状态：未解决。 PR 未包含 reviewer 建议的修复，也未合并相关修改。

- Context Parallel 时 fused_split_qk_norm 导致 latent_cache 未定义 (correctness): 未解决；PR 未采纳任何修改。
- q_lora_rank 为 None 时 k_pe 未定义 (correctness): 未解决；PR 未包含修复。

风险与影响

- 风险：
  1. 回归风险（高）：review 指出的两个 NameError 均属运行时崩溃，若合并将导致 Context Parallel 场景及 q_lora_rank 为 None 的模型无法推理。
  2. 性能风险（低 - 中）：融合算子仅在序列长度 < 65536 且非 CP 时启用，其他场景保持原逻辑，无直接性能退化。
  3. 测试覆盖不足：PR 未提供针对新融合路径的单元测试或集成测试，无法确保 edge case 的可靠性。 - 影响：影响范围：仅 NPU 后端 DeepSeek V2/V3/GLM4.7 系列模型。如果修复了 review 问题，预计可带来 MLA 前处理阶段的速度提升（PR 速度截图显示显著加速）。影响程度：中高——若当前形式合并，会导致特定配置下的崩溃。 - 风险标记：缺少测试覆盖，未修复 Review 指出的崩溃问题，核心路径变更

关联脉络

- PR #29420 [AMD][DSV4] Remove per-batch D2H syncs in MTP to avoid bubbles between 2 batches: 同属 NPU/AMD 后端 DeepSeek V4 模型性能优化系列，关注 MLA 与 MTP 推理效率。
- PR #28980 [NPU] Support DeepSeek V4 Flash MTP on Ascend: 同为 NPU 后端 DeepSeek 系列模型推理优化，涉及 attention 和 allocator 组件。