PR #23815 完整报告

sgl-project/sglang

[NPU] Fix DeepEP LL dispatch BF16 flag and skip triton kernel on NPU for Qwen3.5

合并时间：2026-04-29 10:19

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23815

执行摘要

- 一句话：修复 DeepEP 低延迟分发和 Qwen3.5 NPU triton kernel 崩溃
- 推荐动作：建议合并，修复明确且经 reviewer 确认。建议关注 #22822 对 dispatch 输出类型的自动化改造，以及后续统一环境变量后本 PR 的兼容性。

功能与动机

NPU 推理 Qwen3.5 MoE 模型时，DeepEP 低延迟分发路径未检查 SGLANG_DEEPEP_BF16_DISPATCH，始终对 hidden states 做 INT8/FP8 量化，导致 NPU 上 npu_grouped_matmul 无法处理 INT8 输入与 BF16 权重的类型不匹配；另外 fused_qkvzba_split_reshape_cat_contiguous triton kernel 使用 2D 网格，在 token 数超过 65536 时崩溃，该 kernel 对 NPU 非必需。

实现拆解

1. DeepEP 低延迟分发的 BF16 强制模式（deepep.py）：在 _dispatch_core 方法的 use_fp8 条件中增加 not _is_npu or not SGLANG_DEEPEP_BF16_DISPATCH 约束，使得 NPU 且设置了 BF16 分发环境变量时，不走 FP8 量化，直接使用 BF16 原始数据类型进行 all-to-all 通信。
2. Qwen3.5 GDN forward 跳过 triton kernel（qwen3_5.py）：在条件分支中增加 not _is_npu，使 NPU 自动 fallback 到 PyTorch-native 的 fix_query_key_value_ordering 路径，避免 triton kernel 的 2D 网格越界崩溃；并在 fallback 路径中对 b、a 张量手动调用 .contiguous()，确保内存布局一致性。
3. 无新增测试文件，但作者提供了完整的精度测试（ceval 准确率 0.893）和性能声明（与 triton 路径开销相当）。

关键文件：

- python/sglang/srt/layers/moe/token_dispatcher/deepep.py（模块 MoE 调度；类别 source；类型 core-logic；符号 _dispatch_core）：核心修复点：修改 DeepEP 低延迟分发的 use_fp8 条件，增加 NPU 且 BF16 环境变量设置时的跳过逻辑。
- python/sglang/srt/models/qwen3_5.py（模块 模型层；类别 source；类型 data-contract；符号 Qwen3_5GDN.forward）：次要修复点：在 Qwen3.5 GDN forward 中为 triton kernel 添加 NPU 隔离，并保证 fallback 路径的 contiguous。

关键符号：_dispatch_core, Qwen3_5GDN.forward

关键源码片段

python/sglang/srt/layers/moe/token_dispatcher/deepep.py

核心修复点：修改 DeepEP 低延迟分发的 use_fp8 条件，增加 NPU 且 BF16 环境变量设置时的跳过逻辑。

```
# python/sglang/srt/layers/moe/token_dispatcher/deepep.py

class DeepEPLowLatencyDispatcher:
    def _dispatch_core(self, hidden_states: torch.Tensor, topk_ids: torch.Tensor):
        use_nvfp4 = use_fp8 = False
        input_global_scale = self.quant_config.get("input_global_scale", None)
        if input_global_scale is not None:
            use_nvfp4 = True
        elif not get_moe_runner_backend().is_flashinfer_cutedsl() and (
            not _is_npu or not envs.SGLANG_DEEPEP_BF16_DISPATCH.get()
        ):
            # 对于 NPU 且设置了 BF16 dispatch 环境变量的情况，跳过 FP8 量化
            # 因为 NPU 的 grouped_matmul 不支持 INT8 输入 + BF16 权重
            use_fp8 = True

        # ... 后续逻辑保持不变 ...
        buffer.low_latency_dispatch(
            hidden_states,
            topk_ids,
            ...,
            use_fp8=use_fp8,
            ...
        )
```

python/sglang/srt/models/qwen3_5.py

次要修复点：在 Qwen3.5 GDN forward 中为 triton kernel 添加 NPU 隔离，并保证 fallback 路径的 contiguous。

```
# python/sglang/srt/models/qwen3_5.py

class Qwen3_5GDN(nn.Module):
    def forward(self, hidden_states, forward_batch):
        # ...
        if (
            self.num_v_heads // self.num_k_heads in [1, 2, 4]
            and not _is_cpu
            and not _is_npu # [ 修复 ] NPU 跳过 triton kernel，因 2D grid 在 token>65536 时崩溃
        ):
            mixed_qkv, z, b, a = fused_qkvzba_split_reshape_cat_contiguous(
                projected_states_qkvz, projected_states_ba, ...
            )
        elif _is_cpu and _is_amx_available:
            # CPU 路径
```

```
        ...
    else:
        query, key, value, z, b, a = self.fix_query_key_value_ordering(
            projected_states_qkvz, projected_states_ba
        )
        b = b.contiguous() # [ 修复 ] 确保内存连续，避免后续 cat 或 attention 异常
        a = a.contiguous()
        # ...
```

评论区精华

Reviewer @OrangeRedeng 确认了 DeepEP 输出类型问题的有效性，并提及正在开发自动选择分发输出类型的 PR（#22822），计划移除当前所有三个环境变量。未发现其他争议或未解决疑虑。

- DeepEP 输出类型确认与后续自动化计划 (design): 当前修复作为临时方案，后续将由 #22822 统一解决。

风险与影响

- 风险：
  1. 回归风险低：两处修改均为在现有条件下增加 and not _is_npu 或 not envs.SGLANG_DEEPEP_BF16_DISPATCH 约束，不影响 GPU/CUDA 路径。
  2. NPU 专用路径的性能：fallback 到 fix_query_key_value_ordering 可能略微增加 token 数较多时的显存开销（b.contiguous()、a.contiguous()），但作者声明显着差异。
  3. 依赖环境变量：修复依赖 SGLANG_DEEPEP_BF16_DISPATCH 的正确设置，且 reviewer 计划未来移除该环境变量，届时需要适配。 - 影响：直接影响 NPU 上 Qwen3.5 MoE 模型（尤其是 DeepEP 后端）的正常推理，解除两个 blocker；对 GPU 用户无行为变化。团队协作上，与 reviewer 的 #22822 存在上下游依赖。 - 风险标记：NPU 专用路径，依赖环境变量，未来可能被重构

关联脉络

- PR #22822 [WIP] Auto-select dispatch output type (draft): 与本 PR 主题直接相关，计划自动化 dispatch 输出类型选择并移除环境变量，将影响本 PR 修复的长期兼容性。