PR #32001 完整报告

sgl-project/sglang

[NPU][Fix Issue]: Send expert weights contiguous tensor across cards during EPLB rebalance

合并时间：2026-07-27 09:20

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/32001

执行摘要

- 一句话：修复 EPLB 重平衡时权重张量非连续导致 RuntimeError
- 推荐动作：对于 NPU 平台的开发和维护人员值得关注，特别是涉及跨卡通信的 EPLB 流程；普通用户可以忽略。

功能与动机

修复之前的 PR(#25663) 引入的 RuntimeError：在 EPLB 专家重平衡期间，跨卡发送专家权重时，由于 transpose 后的张量非连续，导致 HCCL P2P 通信报错 "Tensors must be contiguous"。

实现拆解

1. 定位问题：在 moe_methods.py 的 process_weights_after_loading 方法中，处理权重的代码 weight.data = npu_format_cast(weight.data.transpose(1, 2)) 在 transpose 后未保证内存连续，而后续 EPLB 重平衡时需通过 HCCL 发送该权重，非连续张量导致通信失败。
2. 修复方案：在 transpose 后链式调用 .contiguous()，即 weight.data = npu_format_cast(weight.data.transpose(1, 2).contiguous())，显式将张量转为连续内存布局。
3. 影响范围：仅一行变更，仅影响 Ascend NPU 平台上的 EPLB 专家重平衡流程。

关键文件：

- python/sglang/srt/hardware_backend/npu/quantization/moe_methods.py（模块 NPU 量化；类别 source；类型 core-logic）：核心修复文件，在权重处理流程中添加 contiguous 调用以解决跨卡通信问题。

关键符号：process_weights_after_loading

关键源码片段

python/sglang/srt/hardware_backend/npu/quantization/moe_methods.py

核心修复文件，在权重处理流程中添加 contiguous 调用以解决跨卡通信问题。

```
# file: python/sglang/srt/hardware_backend/npu/quantization/moe_methods.py
# process_weights_after_loading 方法中的权重处理部分
```

```
# 原代码：weight.data = npu_format_cast(weight.data.transpose(1, 2))
# 修复后：在 transpose 后添加 .contiguous() 保证内存连续，
# 因为后续 EPLB 重平衡需要通过 HCCL 跨卡发送该权重，非连续张量会触发 RuntimeError
weight.data = npu_format_cast(weight.data.transpose(1, 2).contiguous())
```

评论区精华

无 review 评论，PR 由 Hexq0210 直接批准。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。仅增加 .contiguous() 调用，不会改变张量数值，语义等价。但由于 NPU 厂商可能针对特定布局有优化，连续化可能引入微小的内存复制开销，但在 EPLB 重平衡这种低频操作中可忽略。
- 影响：影响范围小：仅影响 Ascend NPU 上使用 EPLB 专家负载均衡的模型（如 MoE 模型）在触发重平衡时的正确性。修复后避免了 RuntimeError，保证重平衡正常执行。
- 风险标记：暂无

关联脉络

- PR #25663 （未知，原 PR 标题）：本 PR 修复了该 PR 引入的 RuntimeError