PR #26573 完整报告

sgl-project/sglang

[NPU] fix model llava-onevision-qwen2-7b-ov torch compiles error in npu case

合并时间：2026-05-30 17:10

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26573

执行摘要

- 一句话：修复 NPU 环境下 torch.compile 导致图执行错误
- 推荐动作：该 PR 是低风险的必要修复，可快速合并。值得关注的是后续是否需要对 NPU 平台上的 torch.compile 进行更精细的配置（例如仅禁用某些后端），而非完全禁用。

功能与动机

当在 NPU 上部署模型 lmms-lab/llava-onevision-qwen2-7b-ov 时，apply_scaling_penalties 函数中的 @torch.compile 导致图执行错误（附截图）。需要禁用该函数的编译以恢复正常执行。

实现拆解

1. 导入扩展：在 python/sglang/srt/sampling/penaltylib/repetition_penalty.py 中从 sglang.srt.utils 额外导入 is_npu 函数。
2. 添加平台检测变量：在模块级别添加 _is_npu = is_npu()，一次性缓存检测结果。
3. 条件禁用编译：将 @torch.compile 装饰器的 disable 参数设为 _is_npu，使得在 NPU 平台上完全跳过编译，直接使用 *eager* 模式执行。

关键文件：

- python/sglang/srt/sampling/penaltylib/repetition_penalty.py（模块 惩罚采样；类别 source；类型 dependency-wiring）：该文件实现了重复惩罚采样的核心函数 apply_scaling_penalties，本 PR 通过增加 NPU 检测来条件禁用 torch.compile，修复了 NPU 平台上的图执行错误。

关键符号：apply_scaling_penalties

评论区精华

代码风格讨论：机器人 reviewer gemini-code-assist[bot] 建议将 _is_npu 全局变量替换为直接传递 is_npu() 给 disable 参数，以减少模块命名空间污染。但另一位 reviewer Hexq0210 则要求保持一致的写法，并给出了与当前实现一致的示例（即仍使用 _is_npu）。PR 最终未采纳机器人的建议，保持了 _is_npu 全局变量的写法。

- 是否应内联 is_npu() 而非使用全局变量 (style): PR 保持了 _is_npu 全局变量的写法，未采纳内联建议。

风险与影响

- 风险：风险极低：变更仅涉及一行装饰器参数和一行导入，不会影响非 NPU 平台的行为（is_npu() 返回 False 时 torch.compile 正常启用）。仅在 NPU 平台下回退到 *eager* 模式，可能带来轻微性能损失，但保证了正确性。
- 影响：影响范围小：仅影响 NPU 平台上使用 *apply_scaling_penalties* 的场景（即需要重复惩罚采样的模型推理）。修复后 *llava-onevision-qwen2-7b-ov* 等模型可在 NPU 上正常执行而不出现编译错误。
- 风险标记：缺少测试覆盖

关联脉络

- PR #26696 [bugfix]: size CuteDSL MoE allgather buffers for the worst-case forward: 同为 NPU 平台相关的 bugfix，但影响模块不同（MoE vs 惩罚采样）。
- PR #26705 [Bugfix] Fix Ascend NPU CP attention for batch size > 1: 同为 NPU 平台上的 bugfix，关注注意力模块。
- PR #23122 [NPU] DFlash Speculative Decoding Support NPU: 之前为 NPU 添加新功能，本 PR 修复了该过程中的一个编译问题。