PR #24835 完整报告

sgl-project/sglang

[NPU] fix npu profiler

合并时间：2026-06-01 21:44

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24835

执行摘要

- 一句话：修复 NPU profiler 算子形状缺失
- 推荐动作：该 PR 改动简单直接，对于 NPU 用户来说是一个重要的 profiling 修复。值得精读以了解 NPU profiling 配置方式。

功能与动机

根据 PR body，动机是修复 NPU 上 profiling 时算子形状信息缺失的问题，即使设置了 SGLANG_PROFILE_RECORD_SHAPES 环境变量。该 PR 旨在修复此问题并同时收集算子统计信息。

实现拆解

1. 在 python/sglang/srt/managers/scheduler_components/profiler_manager.py 的 _start_profile 方法中，当 _is_npu 为 True 时，为 torch.profiler.profile 调用添加了 experimental_config 参数。
2. experimental_config 使用 torch_npu.profiler._ExperimentalConfig 配置了导出类型为 Text、profiler 等级为 Level1、禁用 msprof_tx、设置 AI 核心度量指标为 PipeUtilization 等参数。
3. 此变更仅在 NPU 路径生效，不影响其他设备。

关键文件：

- python/sglang/srt/managers/scheduler_components/profiler_manager.py（模块 性能分析；类别 source；类型 core-logic）：唯一变更文件，添加 NPU 特定的 profiling 配置

关键符号：ProfilerManager._start_profile

关键源码片段

python/sglang/srt/managers/scheduler_components/profiler_manager.py

唯一变更文件，添加 NPU 特定的 profiling 配置

```
# 在 _start_profile 方法中，当 torchprof_activities 存在时，创建 profiler 实例
self.torch_profiler = torch.profiler.profile(
    activities=torchprof_activities,
    with_stack=with_stack if with_stack is not None else True,
```

```
        record_shapes=record_shapes if record_shapes is not None else False,
        on_trace_ready=(
            None
            if not _is_npu
            else torch_npu.profiler.tensorboard_trace_handler(
                str(self.torch_profiler_output_dir)
            )
        ),
        experimental_config=(
            None
            if not _is_npu
            else torch_npu.profiler._ExperimentalConfig(
                export_type=torch_npu.profiler.ExportType.Text, # 导出类型为文本，也可选 Db
                profiler_level=torch_npu.profiler.ProfilerLevel.Level1, # 性能分析级别
                msprof_tx=False, # 禁用 msprof 传输
                aic_metrics=torch_npu.profiler.AiCMetrics.PipeUtilization, # AI Core
                度量：流水线利用率
                l2_cache=False, # 禁用 L2 cache 分析
                op_attr=False, # 不记录算子属性
                data_simplification=False, # 不简化数据
                record_op_args=False, # 不记录算子参数
                gc_detect_threshold=None, # 不设置 GC 检测阈值
            )
        ),
    )
    self.torch_profiler.start()
```

评论区精华

Reviewer McZyWu 询问为何添加这些可选参数并给出默认值，以及建议添加 export_type 的注释（另一个选项为 Db）。作者 zhaozx-cn 回应请参考 Ascend 主页了解收集 profiling 信息的更多信息。该讨论已解决。

- 关于 experimental_config 参数的默认值选择 (question): Author zhaozx-cn 回应请参考 Ascend 主页获取更多信息。

风险与影响

- 风险：变更仅影响 NPU 平台下的 profiling 路径，增加 experimental_config 配置，不影响其他功能。风险较低，但缺乏针对该配置的单元测试，可能导致 NPU 特定场景下配置不正确。
- 影响：此 PR 对用户的影响：使用 NPU 进行 profiling 的用户现在可以获得正确的算子形状信息和更丰富的算子统计信息。对系统无影响。对团队的影响：维护者需确保 NPU 环境兼容此配置。
- 风险标记：平台特定代码，缺少测试覆盖

关联脉络

- PR #26714 fix test cases failed in nightly pipeline: 同为 NPU 平台的问题修复，体现 NPU 持续维护。