PR #27774 完整报告

sgl-project/sglang

[NPU] Fix dead patch_model monkey-patch breaking NPU torch.compile capture

合并时间：2026-06-10 16:15

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27774

执行摘要

- 一句话：修复 NPU torch.compile 捕获的 monkey-patch 失效 bug
- 推荐动作：该 PR 值得合并，它是一个针对特定平台（NPU）的精确 bug 修复，修改简洁且安全。关注点：代码评审者应注意到 Python 导入语义的陷阱——from module import name 会创建本地引用副本，而 import module 允许动态属性访问。这种模式在需要支持 monkey-patch 的场景下尤为重要。

功能与动机

CUDA Graph Runner/Backend 重构 (#23906) 破坏了 NPU 的 --enable-torch-compile 路径，导致 NPU CI 测试 test_npu_compile_graph_tp1_bf16.py 在 CUDA 图捕获时失败。根本原因是重构后 patch_model 的导入方式使得 NPU 的 monkey-patch 变成了静默的 no-op。修复方案是改变导入方式，在调用点通过模块名访问 patch_model，从而让 monkey-patch 生效。

实现拆解

1. 修改导入语句：在 python/sglang/srt/model_executor/runner/decode_cuda_graph_runner.py 中，将 from sglang.srt.compilation.torch_compile_decoration import patch_model 替换为 from sglang.srt.compilation import torch_compile_decoration。同时保留另一个导入 set_torch_compile_config，因为它没有被 monkey-patch，无需修改。
2. 修改调用点：在 _capture_one_stream 方法中，将 with patch_model(...) 改为 with torch_compile_decoration.patch_model(...)。这样，每次调用时都会从模块对象中动态解析 patch_model，从而能让 NPU 运行前设置的 monkey-patch 生效。
3. 测试验证：通过重新运行 NPU --enable-torch-compile 相关测试和 GPU 编译测试来验证修复。

关键文件：

- python/sglang/srt/model_executor/runner/decode_cuda_graph_runner.py（模块 解码器；类别 source；类型 data-contract）：修复的核心文件：修改导入方式，将 from ... import patch_model 替换为 import torch_compile_decoration，并在调用点使用模块属性访问 patch_model，从而恢复 NPU monkey-patch 的有效性。

关键符号：_capture_one_stream

关键源码片段

python/sglang/srt/model_executor/runner/decode_cuda_graph_runner.py

修复的核心文件：修改导入方式，将 from ... import patch_model 替换为 import torch_compile_decoration，并在调用点使用模块属性访问 patch_model，从而恢复 NPU monkey-patch 的有效性。

```
# decode_cuda_graph_runner.py (head version)

# 修改前：from ...torch_compile_decoration import patch_model # 按值导入，monkey-patch 失效
# 修改后：导入模块本身，运行时动态解析 patch_model
from sglang.srt.compilation import torch_compile_decoration
# set_torch_compile_config 没有被 monkey-patch，仍然可以按值导入
from sglang.srt.compilation.torch_compile_decoration import set_torch_compile_config

class DecodeCudaGraphRunner:
    # ... 其他代码 ...

    def _capture_one_stream(self, stream_idx: Optional[int] = None) -> None:
        # ... 前面的代码 ...
        for bs in capture_range:
            for variant_label, _variant_has_lora in lora_variants:
                _set_capture_lora_variant(variant_label)
                # 修改前：with patch_model(...) # 使用本地副本，monkey-patch 无效
                # 修改后：通过模块属性访问，每次调用都解析最新的 patch_model
                with torch_compile_decoration.patch_model(
                    self.model_runner.model,
                    bs in self.compile_bs,
                    num_tokens=bs * self.num_tokens_per_bs,
                    tp_group=self.model_runner.tp_group,
                ) as forward:
                    self.capture_one_shape(bs, forward, stream_idx, variant_label)
```

评论区精华

该 PR 仅有作者的自我操控评论和自动 CI 评论，无人工 review 讨论。自动代码审查机器人 gemini-code-assist[bot] 仅声明无反馈。

- 暂无高价值评论线程

风险与影响

- 风险：本 PR 修改集中在导入和调用方式上，风险极低。修改后的代码在 GPU 上行为不变（模块属性解析到相同的默认 patch_model），在 NPU 上恢复正确的 monkey-patch 行为。唯一的潜在风险是如果未来有其他代码同样通过 from ... import patch_model 方式导入并期望被 monkey-patch，则可能再次出现类似问题。
- 影响：受影响用户：NPU 平台上使用 --enable-torch-compile 的用户。修复后，他们能正常使用 CUDA 图捕获，获得性能提升。GPU 用户无影响。影响范围：仅修改 decode CUDA graph runner 的导入，不影响其他后端或功能。

- 风险标记：平台特定代码，依赖 Python 导入语义

关联脉络

- PR #23906 [Refactor] Cuda Graph Runner/Backend Refactor: 本 PR 是修复因 PR #23906 重构引入的回归 bug。重构将 patch_model 移动到新模块并改变了导入方式，导致 NPU monkey-patch 失效。