

PR #36573 完整报告

sgl-project/sglang

Fix `_is_compiling` dynamo tracing: import torch instead of `sys.modules` lookup

合并时间: 2026-08-27 07:17

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/36573>

执行摘要

- 一句话: 修复 `_is_compiling` 导致 dynamo 追踪崩溃
- 推荐动作: 该 PR 值得精读, 因为其根因分析展示了 import 顺序如何影响 dynamo 追踪, 以及如何通过简单的 import 改写避免。对于维护涉及 `torch.compile` 与动态 import 的开发者有参考价值。

功能与动机

PR 正文明确指出 `test_runtime_context_config_bags.py::TestRoleNamespaceEnforcement::test_record_mode_bag_read_traces_under_torch_compile` 在 torch 2.13 下失败, 报 `InternalTorchDynamoError: RuntimeError: dictionary changed size during iteration`。根因是 `_is_compiling()` 通过 `sys.modules.get('torch')` 探测 torch, 导致 dynamo 对整个 `sys.modules` 字典构建变量追踪器, 而 dynamo 自身 lazy import 会修改该字典。此问题属于导入顺序偶然性, CI 在部分环境通过是因为这些模块已提前加载。

实现拆解

实现变更集中在 `python/sglang/srt/runtime_context.py` 的 `_is_compiling()` 函数:

1. 移除 `sys.modules.get('torch')` 查找方式, 改为函数内 `import torch`;
2. 保留使用 `torch.compiler.is_compiling()` 作为标准编译探测;
3. 更新注释说明原因。这种改动使得 dynamo 将 import 作为常量绑定处理, 不会物化 `sys.modules` 字典, 从而避免迭代期间被修改的异常, 同时保持模块导入轻量。配套测试包括 `test_runtime_context_config_bags.py` 等, 但本次 PR 未直接修改测试文件。

关键文件:

- `python/sglang/srt/runtime_context.py` (模块 运行时上下文; 类别 source; 类型 dependency-wiring; 符号 `_is_compiling`): 核心修复文件, 修改了 `_is_compiling()` 的实现方式, 从依赖 `sys.modules` 改为函数内 `import`。

关键符号: `_is_compiling`

关键源码片段

`python/sglang/srt/runtime_context.py`

核心修复文件, 修改了 `_is_compiling()` 的实现方式, 从依赖 `sys.modules` 改为函数内 `import`。

```
# python/sglang/srt/runtime_context.py
```

```
def _is_compiling() -> bool:
```

```
    # Recording 有 Python 副作用 (set mutation、文件 I/O、atexit) ,  
    # 绝不能在图追踪时执行; torch.compiler.is_compiling() 是 dynamo 的官方探针。  
    # 函数级 import 保持模块轻量; sys.modules 查找会破坏 fullgraph 追踪  
    # (dynamo 会枚举该字典, 而其他 import 会在追踪过程中修改它) 。  
    import torch
```

```
    return torch.compiler.is_compiling()
```

评论区精华

本次 PR 没有 review 讨论记录, 核心讨论集中在 PR 正文中, 作者详细分析了根因和验证过程, 包括二分定位到 #34496 引入的 `sgl_kernel lazy import` 是触发条件, 并确认预导入 `sgl_kernel` 后测试可通过。

- 暂无高价值评论线程

风险与影响

- 风险: 风险较低:
 1. 改动仅针对 `_is_compiling()` 内部实现, 行为等价, 不影响运行时逻辑;
 2. 函数级 `import torch` 在首次调用时可能增加一次导入开销, 但功能函数调用频率低, 影响可忽略;
 3. 可能仍然存在其他依赖 `sys.modules` 的代码, 但本次改动仅针对该函数。 - 影响: 影响范围限于 `sglang` 运行时配置上下文中对 `torch.compile` 的适配, 特别是正确性测试在 `torch 2.13` 下能够通过。对用户无感知, 对团队而言消除了导入顺序偶然性导致的 CI 不稳定。 - 风险标记: 依赖导入顺序, `torch.compile` 相关

关联脉络

- PR #34496 Make `sgl_kernel` import lazy in `srt/utils/common.py`: PR 正文指出该 PR 引入了 `sgl_kernel` 的 lazy import, 导致导入顺序变化, 从而暴露了此 bug。