PR #42913 完整报告

vllm-project/vllm

Revert "[torch.compile] Add patch for fullgraph compilation" (#42686)

合并时间：2026-05-18 21:02

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/42913

执行摘要

- 一句话：回滚引发 CI 失败的 torch.compile 补丁
- 推荐动作：可直接合并以快速恢复 CI。建议后续维护者关注 PyTorch 2.12 及以上版本是否确实修复该问题，并考虑是否有更安全的方式为 2.11 提供补丁。

功能与动机

PR #42686 引入的补丁导致 nightly build 中两项 CI 正确性测试失败：AsyncTP Correctness Tests 和 Sequence Parallel Correctness Tests 均出现编译输出与基线不一致。PR 描述明确提到“This PR is linked to 2 new CI failures”，因此需要回退以恢复 CI 稳定性。

实现拆解

1. 整体回退：将 vllm/env_override.py 中新增的 _patch_should_realize_on_reuse 函数定义及其末尾的调用 _patch_should_realize_on_reuse() 完全删除，共 62 行代码。
2. 函数内容回顾：被移除的函数包含一个针对 PyTorch 2.11.0 的 monke-ypatch，通过修改 StorageBox.should_realize_on_reuse 来改进中间结果物化决策，避免大数据图编译时重复计算。该补丁在 PyTorch 2.12 中已有官方修复。
3. 无其他文件变更：仅此一个文件修改，无测试或配置配套改动。

关键文件：

- vllm/env_override.py（模块 环境配置；类别 source；类型 core-logic；符号 _patch_should_realize_on_reuse, should_realize_on_reuse_patched）：唯一修改的文件，删除 _patch_should_realize_on_reuse 函数及其调用，共 62 行。该函数是 PR #42686 新增的 torch.compile 补丁。

关键符号：_patch_should_realize_on_reuse, should_realize_on_reuse_patched

关键源码片段

vllm/env_override.py

唯一修改的文件，删除 _patch_should_realize_on_reuse 函数及其调用，共 62 行。该函数是 PR #42686 新增的 torch.compile 补丁。

```
# vllm/env_override.py（删除的部分）
```

```
# 以下整个函数被删除（共 62 行）
def _patch_should_realize_on_reuse():
    """
    Patches the materialization heuristic in Inductor compilation.
    Patched in torch 2.12: https://github.com/pytorch/pytorch/pull/176994
    vLLM issue: https://github.com/vllm-project/vllm/issues/27828

    Without this patch, Inductor inlined a chain of computations when compiling
    the whole model graph at once. For example, the residual in fused_add_rms_norm
    would get recomputed every time, and this would cascade through the model.
    """
    # Only apply for 2.11, fix included in 2.12
    if not is_torch_equal("2.11.0"):
        return

    def should_realize_on_reuse_patched(self, users: int) -> bool:
        """
        A heuristic to decide if we should realize a tensor
        that is used multiple times.
        """
        from torch._inductor import config
        from torch._inductor.ir import Pointwise, Reduction, is_cpu
        from torch._inductor.virtualized import V

        if users > 1 and isinstance(self.data, (Pointwise, Reduction)):
            if is_cpu(self.data):
                opcount = self.data.inner_fn_opcount()
                heavy_ops = ["exp", "sigmoid"]
                if any(x in opcount.used_ops for x in heavy_ops):
                    return True
            if self.has_large_inner_fn():
                return True
            # Size-aware cost model comparing total memory traffic:
            # Inline: total_read_bytes * users
            # Materialize: total_read_bytes + output_bytes * (1 + users)
            total_read_bytes = sum(
                V.graph.get_dep_size_hint(dep) for dep in self.get_reads()
            )
            output_bytes = (
                V.graph.sizevars.optimization_hint(self.data.get_numel(), fallback=0)
                * self.data.dtype.itemsize
            )
            if total_read_bytes > 0 and output_bytes > 0:
                return total_read_bytes * (users - 1) >= output_bytes * (1 + users)
            return self.num_reads() > config.realize_reads_threshold
        return False

    from torch._inductor.ir import StorageBox
    StorageBox.should_realize_on_reuse = should_realize_on_reuse_patched
```

```
# 原文件末尾调用也被删除：
# _patch_should_realize_on_reuse()
```

评论区精华

无实质讨论。仅有一条来自 gemini-code-assist[bot] 的自动代码审查总结，确认移除内容。ProExpertProg 直接批准，未留下其他评论。

- 暂无高价值评论线程

风险与影响

- 风险：低风险：回退操作只删除代码，无新增逻辑。风险主要在于：若后续仍需此补丁（例如用户仍使用 PyTorch 2.11），会回退性能优化，但该补丁本身已针对 PyTorch 2.11 固定版本，且官方已在 2.12 修复，因此影响有限。
- 影响：
  - 对用户：使用 PyTorch 2.11 的用户可能恢复遇到大数据图编译时的中间结果重复计算问题（原始 issue #27828），但实际使用中影响较小。
  - 对系统：CI 正确性测试预计恢复通过。
  - 对团队：需在后续 PR 中重新提交更稳健的补丁方案，或等待用户升级 PyTorch。
  - 风险标记：暂无

关联脉络

- PR #42686 [torch.compile] Add patch for fullgraph compilation: 被回退的原 PR，引入了导致 CI 失败的 _patch_should_realize_on_reuse 函数。