# PR #44410 完整报告

vllm-project/vllm

[Bugfix] Fix VLLMNotFoundError when using LoRA adapter name in poolin···

合并时间：2026-06-04 10:22

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/44410

## 执行摘要

- 一句话：修复 Pooling 端点 LoRA 名称匹配后 404 错误
- 推荐动作：该 PR 属于高信号的小型 bugfix，值得精读其修复模式和测试设计，特别是如何构造 PoolingServingBase 的测试替身。

## 功能与动机

当 pooling/embed 服务通过 --lora-modules 加载 LoRA 适配器，且请求的 model 参数与适配器名称完全匹配时，本应正常处理，但实际上返回 404 错误。这是因为 _maybe_get_adapters 在设置 ctx.lora_request 后没有提前返回，继续执行 _is_model_supported 检查（该检查期望 model 名称为基础模型名），导致失败。修复思路与 OpenAIServing 中 _maybe_get_adapters 的行为保持一致（由 PR #36110 引入）。

## 实现拆解

1. 修复核心逻辑（vllm/entrypoints/pooling/base/serving.py）：在 _maybe_get_adapters 方法中，当 request.model 存在于 self.models.lora_requests 时，设置 ctx.lora_request 后立即 return None，避免后续的 _is_model_supported 检查失败引发 VLLMNotFoundError。
2. 新增单元测试（tests/entrypoints/serve/lora/test_serving_models.py）：
   - 创建 _ConcretePoolingServing 辅助类，继承 PoolingServingBase，仅实现必需的抽象方法。
   - 实现 _make_pooling_serving 和 _make_pooling_ctx 工厂函数，便于构造测试场景。
   - test_pooling_maybe_get_adapters_lora_name_sets_lora_request：验证当 model 名称匹配 LoRA 适配器时，_maybe_get_adapters 正确设置 lora_request 且不抛出异常。
   - test_pooling_maybe_get_adapters_unknown_model_raises：验证当 model 名称既不是基础模型也不是 LoRA 适配器时，仍抛出 VLLMNotFoundError，确保修复未过度放宽校验。

关键文件：

- vllm/entrypoints/pooling/base/serving.py（模块 端点服务；类别 source；类型 core-logic；符号 _maybe_get_adapters）：修复的核心文件，_maybe_get_adapters 方法增加 return None 防止误抛 VLLMNotFoundError

- tests/entrypoints/serve/lora/test_serving_models.py（模块 测试；类别 test；类型 test-coverage；符号 _ConcretePoolingServing, _make_pooling_serving, _make_pooling_ctx, test_pooling_maybe_get_adapters_lora_name_sets_lora_request）：新增完整的单元测试，覆盖正常匹配和未知 model 两种场景

关键符号：_maybe_get_adapters

## 关键源码片段

### vllm/entrypoints/pooling/base/serving.py

修复的核心文件，_maybe_get_adapters 方法增加 return None 防止误抛 VLLMNotFoundError

```
def _maybe_get_adapters(
    self,
    ctx: PoolingServeContext,
    supports_default_mm_loras: bool = False,
):
    request = ctx.request
    if request.model in self.models.lora_requests:
        ctx.lora_request = self.models.lora_requests[request.model]
        return None # 匹配到 LoRA 适配器名称后立即返回，避免后续 _is_model_supported 校验失败

    # 默认多模态 LoRA 支持
    if supports_default_mm_loras:
        default_mm_lora = self._get_active_default_mm_loras(request)
        if default_mm_lora is not None:
            ctx.lora_request = default_mm_lora

    if self._is_model_supported(request.model):
        return None

    # 未匹配任何已知 model 或 adapter 时抛出异常
    raise VLLMNotFoundError(f"The model `{request.model}` does not exist.")
```

## 评论区精华

审核者 noooop 在第一个 commit 的 diff 上评论要求添加测试来守护该修复。提交者在第二个 commit 中增加了完整的单元测试，随后审核者批准了 PR。

- 添加测试保护修复 (testing): 提交者随后在第二个 commit 中添加了完整的单元测试，审核者批准。

## 风险与影响

- 风险：风险很低。该修复仅改动一条语句（在已有条件分支内增加 return None），逻辑清晰且与 OpenAIServing 中的既有模式一致。测试覆盖了成功匹配和未知 model 两种场景，回归风险极小。

- 影响：
  - 用户：使用 --runner pooling 配合 --lora-modules 的用户将不再收到错误的 404 响应，适配器名称可正常用于路由。
  - 系统：不影响其他端点或非 LoRA 场景。
  - 团队：代码库行为一致性提升，减少未来混淆。
  - 风险标记：核心路径变更，缺少测试覆盖（已修复）

# 关联脉络

- PR #36110 Add _maybe_get_adapters in OpenAIServing: 该 PR 引入了 OpenAIServing 中的 _maybe_get_adapters 方法，本 PR 同步其行为到 PoolingServingBase 中。