PR #41845 完整报告

vllm-project/vllm

[Bugfix] Fix OOM in tensorizer LoRA deserialization

合并时间：2026-05-07 17:17

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/41845

执行摘要

- 一句话：修复 tensorizer 反序列化 LoRA 时未指定设备导致 OOM
- 推荐动作：值得合并。修复清晰，一行变更解决明确的 OOM 问题。

功能与动机

修复 CI 中 test_tp2_serialize_and_deserialize_lora 在 22 GiB GPU 上因 tensorizer 反序列化 LoRA 时直接占用 GPU 显存导致 OOM 的问题（见 Buildkite 构建 #64715 失败记录）。调用方已将 device='cpu' 传入 from_local_checkpoint，但 tensorizer 路径未透传此参数。

实现拆解

1. 在 vllm/lora/lora_model.py 的 TensorDeserializer 构造函数调用中，添加 device=device 参数，使 LoRA 权重反序列化时使用调用方指定的设备（CPU）。
2. 该参数由 worker_manager 在调用 from_local_checkpoint 时传入，值为 'cpu'。
3. 其他反序列化路径（如 safetensors、torch.load）已正确使用 device 参数，本次仅对齐 tensorizer 路径的行为。

关键文件：

- vllm/lora/lora_model.py（模块 LoRA；类别 source；类型 data-contract）：TensorDeserializer 缺少 device 参数导致 OOM 的根源文件，修复仅在此进行。

关键符号：未识别

关键源码片段

vllm/lora/lora_model.py

TensorDeserializer 缺少 device 参数导致 OOM 的根源文件，修复仅在此进行。

```
# vllm/lora/lora_model.py 第 203-207 行
# 修复前：缺少 device 参数，TensorDeserializer 默认将张量加载到 GPU，
# 导致 KV cache 已占用显存后再度分配显存触发 OOM
# 修复后：传入 device 参数（通常为 "cpu"），由调用方 worker_manager 控制
# LoRA 权重先落 CPU，再由 LoRA manager 按内存预算移动至 GPU
tensorizer_args = tensorizer_config._construct_tensorizer_args()
tensors = TensorDeserializer(
    lora_tensor_path,
```

```
        dtype=tensorizer_config.dtype,
        device=device, # 关键修复：显式传入 device 参数
        **tensorizer_args.deserialization_kwargs,
    )
```

评论区精华

Code review 机器人（gemini-code-assist）确认变更正确；审核者 jeejeelee 批准，认为可修复 LoRA 失败。无实质性讨论。

- 暂无高价值评论线程

风险与影响

- 风险：极低风险：仅添加一个参数传递，不改变反序列化逻辑，且与其他路径行为一致。唯一的依赖是 tensorizer 库的 TensorDeserializer 支持 device 参数（已确认支持）。
- 影响：影响范围小：仅影响使用 tensorizer 加载 LoRA 权重的场景（其他反序列化路径未受影响）。修复后，在显存受限的 GPU（如 22 GiB）上可正常通过 CI 测试。
- 风险标记：低风险

关联脉络

- 暂无明显关联 PR