PR #24555 完整报告

sgl-project/sglang

[LoRA] Use deterministic lora_id for --lora-paths so multi-node ranks agree

合并时间：2026-05-07 13:20

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24555

执行摘要

- 一句话：修复多节点 LoRA 的 lora_id 不一致问题
- 推荐动作：可安全合并。建议为 deterministic_id 添加单元测试，并考虑扩展至多节点 LoRA 的集成测试。

功能与动机

多节点分布式 LoRA 推理时，每个节点独立解析 --lora-paths 参数，由于 uuid4() 每次调用生成不同的 ID，导致同一适配器在不同节点上的 lora_id 不同。主节点广播请求后，其他节点无法匹配本地LoRA引用，引发断言错误或KeyError，使得多节点--lora-paths适配器无法使用。

实现拆解

1. 新增确定性 ID 生成方法：在 lora_registry.py 的 LoRARef 类中添加 deterministic_id(lora_name, lora_path) 静态方法，使用 uuid5(NAMESPACE_URL, f"{name}\0{path}").hex 生成 32 位十六进制 ID，其中 NUL 分隔符避免名称和路径的边界模糊。
2. 修改 CLI 参数解析：在 server_args.py 的 check_lora_server_args() 中，将四种 --lora-paths 输入格式（name=path 字符串、裸路径字符串、字典列表、顶层字典）的 LoRARef 构造都替换为传入 lora_id=LoRARef.deterministic_id(...)，确保解析时生成一致的 ID。
3. 保持动态加载路径不变：LoRARef 构造函数的默认 lora_id 仍为 uuid4().hex，仅 CLI 解析路径使用确定性 ID，动态 HTTP 加载的适配器仍使用随机 ID，因为其广播机制本身就是跨节点一致的。
4. 验证与测试：在双节点 B200 集群上对四种输入格式进行字节级对比验证，diff 输出为空，确认 ID 一致。

关键文件：

- python/sglang/srt/lora/lora_registry.py（模块 LoRA；类别 source；类型 core-logic；符号 deterministic_id）：新增 deterministic_id 静态方法，提供确定性 ID 生成逻辑；修改导入添加 NAMESPACE_URL 和 uuid5。
- python/sglang/srt/server_args.py（模块 参数解析；类别 source；类型 core-logic）：修改 check_lora_server_args() 中所有 --lora-paths 解析分支，传入 lora_id 参数，使多节点 ID 一致。

关键符号：deterministic_id

评论区精华

无 review 评论。

- 暂无高价值评论线程

风险与影响

- 风险：低风险。变更仅在 CLI 解析路径修改 lora_id 的生成方式，不影响运行时逻辑。确定性 ID 可能导致哈希碰撞（理论上概率极低），但使用 uuid5 和完整名称路径空间可忽略。单节点功能不受影响，多节点 CI 缺少覆盖，需注意后续回归。
- 影响：影响仅限于分布式 LoRA 场景。用户使用 --lora-paths 时多节点 ID 一致，功能恢复正常。动态加载的适配器 ID 不变。无性能影响。
- 风险标记：缺少多节点 CI 测试

关联脉络

- 暂无明显关联 PR