

PR #2246 完整报告

THUDM/slime

fix: cast gpu_id to int in sort_key to prevent lexicographic ordering

合并时间: 2026-08-12 13:29

原文链接: <http://prhub.com.cn/THUDM/slime/pull/2246>

执行摘要

- 一句话: 修复 gpu_id 字符串排序导致 16 卡 GPU 顺序错乱
- 推荐动作: 该 PR 是小而关键的 bugfix, 值得精读, 尤其是 placement_group.py 中的 sort_key 逻辑。建议合并后补充一个针对多卡排序的单元测试, 以覆盖此场景, 防止回归。

功能与动机

关联 Issue #2245 指出, 由于 gpu_id 使用字符串排序而非 int 排序, 导致排序后的 gpu_id 为 0,1,10,11,12,13,14,15,2,3,4,5,6,7,8,9, 在 16 卡机器上, 8 卡训练 8 卡推理时, 划分给 sglang 的卡为 2,3,4,5,6,7,8,9, 但 sglang 通过传入 base_gpu_id 控制后 8 张卡, 实际索引超出范围导致报错。Issue 明确说明该错误并非由设置 CUDA_VISIBLE_DEVICES 引起, 而是排序规则错误地使用了字符串排序。

实现拆解

1. 定位问题: 在 slime/ray/placement_group.py 中的 sort_key 函数内, 排序元组返回 (node_ip_parts, gpu_id), 其中 gpu_id 可能是字符串类型, 导致按字典序排序。
2. 修复方案: 将 gpu_id 转换为 int 后返回, 即 return (node_ip_parts, int(gpu_id)), 确保按数值顺序排序。
3. 影响范围: 该函数用于 placement group 创建时的 GPU 分配排序, 修复后多卡 GPU 顺序将正确, 避免后续 device map 错误。
4. 测试配套: 本次改动未包含直接对应的测试文件变更, 但该逻辑简单, 且 Issue 中已提供复现场景。

关键文件:

- slime/ray/placement_group.py (模块 调度器; 类别 source; 类型 core-logic; 符号 sort_key): 核心修复文件, 修改 sort_key 函数中的 gpu_id 排序逻辑, 从字符串排序改为 int 排序, 修复多卡 GPU 分配顺序错误的问题。

关键符号: sort_key

关键源码片段

[slime/ray/placement_group.py](#)

核心修复文件，修改 sort_key 函数中的 gpu_id 排序逻辑，从字符串排序改为 int 排序，修复多卡 GPU 分配顺序错误的问题。

```
# slime/ray/placement_group.py
```

```
def sort_key(x):
    index, node_identifier, gpu_id = x
    # Sort by node IP number and then by GPU ID
    try:
        # 尝试将节点标识解析为 IP 地址
        ip_address = node_identifier
        node_ip_parts = list(map(int, ip_address.split(".")))
    except ValueError:
        # 如果解析失败，尝试将主机名解析为 IP 地址
        try:
            ip_address = socket.gethostbyname(node_identifier)
            node_ip_parts = list(map(int, ip_address.split(".")))
        except (socket.gaierror, TypeError):
            # 如果仍失败，则将标识符的每个字符转换为 ASCII 值，
            # 提供稳定且一致的数值表示用于排序
            node_ip_parts = [ord(c) for c in node_identifier]

    # 关键修复：将 gpu_id 从字符串转为 int，避免按字典序排序（例如 10 排在 2 之前），
    # 确保 GPU 按数值递增正确排序，防止后续 device map 错乱
    return (node_ip_parts, int(gpu_id))
```

评论区精华

该 PR 无 review 评论和讨论线程记录。

- 暂无高价值评论线程

风险与影响

- 风险：

1. 回归风险：改动仅涉及一行，将 gpu_id 强制转为 int，如果 gpu_id 原本存在非数值字符串（如 'A'），则可能抛出 ValueError，但根据 Ray 的 GPU ID 格式，通常为整数，风险较低。
2. 兼容性：该改动不影响较少数量的 GPU（如少于 10 卡），因为字符串排序与 int 排序结果一致。
3. 部署影响：修复后 GPU 排序更符合直觉，但可能改变现有环境下的 GPU 分配顺序，若用户依赖旧顺序可能需要调整配置，但旧顺序本身是错误的。- 影响：影响范围：该修复影响所有使用 placement group 的多卡场景，尤其是单机 16 卡及以上的训练和推理。对于 10 卡以下的场景，行为不变。由于排序结果更正确，用户无需额外配置即可获得正确的 GPU 分配，解决了 Issue #2245 中描述的 device map 错误。团队维护成本极低，修复符合预期。- 风险标记：缺少测试覆盖

关联脉络

- PR #2170 Fix placement group crash for external engines under debug_rollout_only:
同一文件 `slime/ray/placement_group.py` 的修改, 涉及 placement group 的 GPU 分配逻辑, 本 PR 进一步修复了 GPU 排序问题。