

PR #1797 完整报告

THUDM/slime

pass critic role through to create RayTrainGroup

合并时间: 2026-04-03 09:11

原文链接: <http://prhub.com.cn/THUDM/slime/pull/1797>

执行摘要

此 PR 修复了创建 critic 训练组时未正确传递 role 参数的问题, 通过修改 allocate_train_group 函数增加 role 参数并显式传递 "critic" 值, 确保角色分配符合代码注释预期。变更范围小、风险低, 属于常规维护性修复。

功能与动机

根据 PR body 引用的代码注释 (slime/ray/actor_group.py 第 75-77 行), 创建 critic 的 RayTrainGroup 时使用 actor 角色不符合预期。作者 znculee 发现此不一致问题并提交修复, 确保 critic 训练组正确使用 "critic" 角色。

实现拆解

修改 slime/ray/placement_group.py 文件:

1. allocate_train_group 函数: 增加 role 参数, 默认值为 "actor", 并将参数传递给 RayTrainGroup 构造函数。

```
python def allocate_train_group(args, num_nodes, num_gpus_per_node, pg, role="actor"): return RayTrainGroup( args=args, num_nodes=num_nodes, num_gpus_per_node=num_gpus_per_node, pg=pg, num_gpus_per_actor=0.4, role=role, # 新增参数传递 )
```
2. create_training_models 函数: 在创建 critic 训练组时显式传递 role="critic"。

```
python critic_model = allocate_train_group( args=args, num_nodes=args.critic_num_nodes, num_gpus_per_node=args.critic_num_gpus_per_node, pg=pgs["critic"], role="critic", # 显式指定角色 )
```

评论区精华

无 review 评论, PR 直接由 zhuzilin 合并。提交历史显示作者先提交修复 commit, 然后合并 main 分支更新, 表明这是基于最新代码的简单修复。

风险与影响

风险分析:

- 变更范围小, 仅修改一个函数签名和两处调用。
- 向后兼容, 新增参数有默认值, 不影响现有调用。
- 逻辑简单, 只是参数传递, 不涉及复杂业务逻辑。

影响分析:

- 对用户无直接影响, 这是内部训练逻辑修复。
- 对系统影响小, 确保 critic 训练组正确使用 "critic" 角色, 修复了原有错误。
- 对团队影响小, 代码变更简单易懂。

关联脉络

与近期历史 PR 对比:

- PR #1775: 修复 Megatron LR scheduler 重复恢复问题, 同为训练相关逻辑修复。
- PR #1741: 修复 encoder_only 属性缺失导致的运行时错误, 同为 bugfix 类型。

此 PR 延续了仓库近期对训练逻辑和配置的持续优化趋势, 体现了对代码细节的关注。