

PR #2743 完整报告

radixark/miles

Skip non-numeric rewards in the episode average

合并时间: 2026-08-26 08:43

原文链接: <http://prhub.com.cn/radixark/miles/pull/2743>

执行摘要

- 一句话: 跳过非数值奖励, 修复 OPD 训练崩溃
- 推荐动作: 值得精读, 作为处理自定义奖励函数返回非数值类型的防御性编程范例。建议阅读 `miles/ray/rollout/metrics.py` 中的 `_compute_training_sample_metrics` 函数, 并考虑补充测试覆盖非数值奖励场景。

功能与动机

PR body 指出: `_compute_training_sample_metrics` (#2710) 对每个 rollout 的原始奖励取平均。当 `--reward-key` 未设置时, `Sample.get_reward_value` 返回自定义奖励函数产生的任意结果——on-policy distillation 返回教师模型的评分响应, 导致 `sum(rewards)` 抛出 `TypeError: unsupported operand type(s) for +: 'int' and 'dict'`, 使整个 rollout 崩溃。该问题已在 CI run 32803020540 中出现, 必须修复以确保训练流程稳定。

实现拆解

1. 在 `miles/ray/rollout/metrics.py` 中导入 `Number` 类型 (来自 `numbers` 模块), 用于判断奖励是否为数值。
2. 修改 `_compute_training_sample_metrics` 中的奖励收集逻辑: 使用 `isinstance(raw_reward, Number)` 检查, 仅当奖励为数值时才将其加入对应 rollout 的奖励列表。
3. 非数值奖励不会影响 `num_training_samples` 的计数, 只会被排除在平均奖励计算之外, 同时保留 `episode_raw_reward` 的逻辑, 若所有奖励均为非数值则返回 0.0。此修改不影响样本计数逻辑, 且与后续 `post_process_rewards` 的 OPD 处理兼容。

关键文件:

- `miles/ray/rollout/metrics.py` (模块 指标计算; 类别 source; 类型 bugfix; 符号 `_compute_training_sample_metrics`): 核心修复文件: 在奖励平均计算中增加类型检查, 跳过非数值奖励。

关键符号: `_compute_training_sample_metrics`

关键源码片段

`miles/ray/rollout/metrics.py`

核心修复文件：在奖励平均计算中增加类型检查，跳过非数值奖励。

```
from numbers import Number

def _compute_training_sample_metrics(args, samples):
    """计算训练样本指标：跳过非数值奖励。"""
    rewards_by_rollout = {}
    use_metadata_reward = bool(samples and samples[0].metadata and "raw_reward" in
samples[0].metadata)
    for position, sample in enumerate(samples):
        # 确定 rollout 键
        if sample.rollout_id is not None:
            rollout_key = ("rollout", sample.group_index, sample.rollout_id)
        elif sample.index is not None:
            rollout_key = ("sample", sample.group_index, sample.index)
        else:
            rollout_key = ("position", sample.group_index, position)
        # 获取原始奖励
        raw_reward = sample.metadata["raw_reward"] if use_metadata_reward else sample.get_
reward_value(args)
        # 仅保留数值奖励，非数值（如 dict）跳过避免 sum() 崩溃
        if isinstance(raw_reward, Number):
            rewards_by_rollout.setdefault(rollout_key, []).append(raw_reward)
    # 计算每个 rollout 的平均奖励
    rollout_rewards = [sum(rewards) / len(rewards) for rewards in rewards_by_rollout.values()]
    return {
        "num_training_samples": len(samples),
        "episode_raw_reward": sum(rollout_rewards) / len(rollout_rewards) if rollout_rewards else
0.0,
    }
```

评论区精华

Review 过程中，Claude 机器人仅提示了配置，Shi-Dong 批准了 PR，无实质讨论。PR body 中作者详细说明了设计权衡：虽然 OPD 的标量奖励定义为 0.0，但共享指标助手不应假设所有自定义奖励函数都返回标量，因此选择跳过非数值类型而非强制转换。

- 暂无高价值评论线程

风险与影响

- 风险：
 1. 该修改使用 `isinstance(raw_reward, Number)` 判断，对于 numpy 数值类型可能无法识别，因为 `np.float32` 等并不继承自 `numbers.Number`，可能导致有效数值奖励被错误跳过，影响指标准确性。
 2. 修改位于核心指标计算路径，若实际奖励为纯字符串等类型，该修改无法覆盖，仍需依赖具体实现。

3. 没有新增测试，容易回归，需补充针对非数值奖励的单元测试。 - 影响：影响范围限于训练指标计算，不影响训练本身。修复可防止 OPD 等自定义奖励函数导致整个训练任务崩溃，提高系统稳定性。对使用默认奖励配置（数值奖励）的用户无影响。 - 风险标记：缺少测试覆盖，数值类型判断可能遗漏 numpy 类型

关联脉络

- PR #2710 Log compaction-aware rollout metrics: 本 PR 修改的 `_compute_training_sample_metrics` 函数正是 #2710 引入的，且 PR body 明确引用 #2710 作为问题来源。