

PR #6384 完整报告

verl-project/verl

[trainer] feat: deprecate main_ppo.py warning

合并时间: 2026-05-18 12:31

原文链接: <http://prhub.com.cn/verl-project/verl/pull/6384>

执行摘要

- 一句话: 标记 `main_ppo.py` 和 `RayPPOTrainer` 为弃用
- 推荐动作: 建议合并前修正 review 中提出的问题: 修正拼写错误, 并将消息改为仅传递替代模块名称 (如 `"verl.trainer.main_ppo_sync"`), 而不是完整句子, 以利用 `deprecated` 装饰器的自动格式化功能。

功能与动机

根据 PR body 描述, `main_ppo.py` 已被弃用, 将在 v0.8.0 中被 `main_ppo_sync.py` 替代, 需要向用户发出警告以引导迁移。

实现拆解

1. 新增 `deprecated` 导入: 在 `verl/trainer/main_ppo.py` 和 `verl/trainer/ppo/ray_trainer.py` 中, 从 `verl.utils.import_utils` 导入 `deprecated` 函数。
2. 装饰 `main` 函数: 在 `main_ppo.py` 的 `main` 函数上添加 `@deprecated("main_ppo.py is deprecated, and will be replaced by main_ppo_sync.py in v0.8.0, please use main_ppo_sync.py instead.")`。
3. 装饰 `RayPPOTrainer` 类: 在 `ray_trainer.py` 的 `RayPPOTrainer` 类定义前添加相同的 `@deprecated` 装饰器。

关键文件:

- `verl/trainer/main_ppo.py` (模块 主入口; 类别 `source`; 类型 `dependency-wiring`; 符号 `main`): 主入口文件, 添加了 `deprecated` 导入和装饰器, 是弃用警告的核心修改点之一。
- `verl/trainer/ppo/ray_trainer.py` (模块 Ray 训练器; 类别 `source`; 类型 `dependency-wiring`; 符号 `RayPPOTrainer`): 作为 `RayPPOTrainer` 类的定义文件, 添加了相同的弃用装饰器, 确保通过该类实例化的方式也能收到警告。

关键符号: `main`

关键源码片段

`verl/trainer/main_ppo.py`

主入口文件, 添加了 `deprecated` 导入和装饰器, 是弃用警告的核心修改点之一。

```
# verl/trainer/main_ppo.py
```

```

from verl.utils.import_utils import deprecated

@deprecated(
    # TODO: 修正为仅传递替代模块名，而非完整句子
    "main_ppo.py is deprecated, and will be replaced by main_ppo_sync.py in v0.8.0, please use main_ppo_sync.py instead."
)
@hydra.main(config_path="config", config_name="ppo_trainer", version_base=None)
def main(config):
    """Main entry point for PPO training with Hydra configuration management."""
    auto_set_device(config)
    config = migrate_legacy_reward_impl(config)
    run_ppo(config)

```

verl/trainer/ppo/ray_trainer.py

作为 `RayPPOTrainer` 类的定义文件，添加了相同的弃用装饰器，确保通过该类实例化的方式也能收到警告。

```

# verl/trainer/ppo/ray_trainer.py
from verl.utils.import_utils import deprecated, load_class_from_fqn

@deprecated(
    # TODO: 修正为仅传递替代类名，而非完整句子
    "main_ppo.py is deprecated, and will be replaced by main_ppo_sync.py in v0.8.0, please use main_ppo_sync.py instead."
)
class RayPPOTrainer:
    """Distributed PPO trainer using Ray for scalable reinforcement learning."""
    # 类定义保持不变 ...

```

评论区精华

review 评论指出 `deprecated` 装饰器会自动构造警告消息，传入完整句子会导致冗余和语法错误（如重复“Please use”），且存在拼写错误（“wil”）。建议只提供替代模块名或类名，而不是完整句子。

- `deprecated` 装饰器使用方式 (design): 未解决。PR 未被修改即合并。
- 弃用消息应引用替代类而非脚本文件 (correctness): 未解决。PR 未被修改即合并。

风险与影响

- 风险：低风险。仅添加装饰器，不影响运行时逻辑。但当前消息格式错误可能导致用户困惑，且 `pytype` 检查会因传递字符串而非类型而报错。
- 影响：对用户：运行 `main_ppo.py` 或使用 `RayPPOTrainer` 时会在控制台或日志中显示弃用警告，提示迁移到 `main_ppo_sync.py`。对系统：无功能影响。
- 风险标记：暂无

关联脉络

- PR #6340 [trainer] feat: support ReMax in synchronous TransferQueue trainer: 涉及同步 Trainer (main_ppo_sync.py) 的增强, 与弃用 old PPO trainer 的动机相关。