

# PR #2878 完整报告

radixark/miles

[PPO] Support the critic under --megatron-to-hf-mode bridge

合并时间: 2026-09-03 04:43

原文链接: <http://prhub.com.cn/radixark/miles/pull/2878>

## 执行摘要

- 一句话: 为 PPO 训练的 bridge 模式增加 critic 模型支持, 修复权重加载冲突并更新文档。
- 推荐动作: 此 PR 值得精读。它是一个针对性强的功能增强, 解决了 PPO 训练路径中的一个具体限制。其设计亮点在于: 1) 通过“值头隐藏机制”(`_hide_critic_value_head_from_hf_load`) 优雅地解决了权重加载冲突, 该模式可复用于未来类似场景; 2) 将 critic 构建逻辑对齐到已有路径 (`raw/custom`), 保持了代码一致性。关注点在于该机制的健壮性以及 PPO+Bridge LoRA 组合未经测试的边界条件。对于负责 PPO、模型后端或检查点加载的工程师, 理解此次变更及其设计决策尤为有用。

## 功能与动机

根据 PR body 描述, PPO 训练之前会拒绝 `--megatron-to-hf-mode bridge`, 并显示 “Critic models are not supported”。根本原因在于 bridge 模型提供者忽略了 `role` 参数, 导致构建的 critic 模型输出层是一个与词表大小相同的语言模型头 (`vocab-sized`), 且权重可能与嵌入层绑定 (`tie_word_embeddings`)。这使得 `get_values` 函数本应得到 `[1, T, 1]` 的输出, 却得到了 `[1, T, vocab]`, 维度不匹配。因此需要修复 bridge 模式, 使其能正确构建 critic。

## 实现拆解

1. 修改模型提供者 (`miles/backends/megatron_utils/model_provider.py`): 在 bridge 模式的 provider 构建逻辑中, 增加了对 `role == "critic"` 的判断。在调用 `provider.finalize()` 之前, 设置 `provider.share_embeddings_and_output_weights = False`, 以解耦嵌入层和输出层。在 `wrapped_bridge_provider` 返回模型后, 如果 `post_process` 且是 critic 角色, 则将其 `output_layer` 替换为 `LinearForLastLayer` (维度为 `[hidden, 1]`), 构建 value head。这使 bridge 路径与 `raw` 和 `custom` 路径的 critic 构建逻辑对齐。
2. 实现值头隐藏机制 (`miles/backends/megatron_utils/checkpoint.py`): 新增 `_hide_critic_value_head_from_hf_load` 上下文管理器。在 `ddp_model` 中查找所有类型为 `LinearForLastLayer` 的子模块 (即 critic 的 value head), 在进入上下文时临时将其从模型中移除 (`delattr`), 在退出时恢复 (`setattr`)。这样在调用 `bridge.load_hf_weights(ddp_model)` 进行 HF 到 Megatron 的权重映射时, 这些 head 不会被尝试映射到 `lm_head.weight`, 从而避免了因形状不匹配 (例如 `tie_word_embeddings=false` 的模型在 `TP=1` 时) 或误匹配导致的错误。
3. 移除参数验证限制与更新测试 (`miles/utils/arguments.py`, `tests/fast/utils/test_arguments.py`): 在 `miles_validate_args` 中, 移除了断言

args.megatron\_to\_hf\_mode != "bridge" 的代码，允许 PPO 训练使用 bridge 模式。相应地，将单元测试 test\_bridge\_mode\_rejects\_critic 重命名为 test\_bridge\_mode\_accepts\_critic，并修改为验证 bridge 模式可以成功通过参数校验，且 args.use\_critic 为 True。

- 更新文档 (examples/ppo/README.md, docs/examples/ppo.md, docs/advanced/lora.md): 在 PPO 示例文档中，更新了关于 Megatron 后端的说明，明确指出 --megatron-to-hf-mode bridge 现已支持 critic。在 LoRA 文档中，将之前关于 PPO+Bridge LoRA 不支持的陈述，更新为“未经测试” (untested)，并解释了原因是 critic 不使用适配器。

关键文件：

- miles/backends/megatron\_utils/checkpoint.py (模块 检查点管理；类别 source；类型 core-logic；符号 \_hide\_critic\_value\_head\_from\_hf\_load)：引入了关键的值头隐藏机制 \_hide\_critic\_value\_head\_from\_hf\_load，用于在 HF 权重加载时临时移除 critic 的 value head，避免映射冲突。
- miles/backends/megatron\_utils/model\_provider.py (模块 模型构建；类别 source；类型 data-contract)：修改了 bridge 模式的模型提供者，使其能够根据 role 参数正确构建 critic 模型，包括解耦嵌入 / 输出层并替换输出层为 value head。
- miles/utils/arguments.py (模块 参数验证；类别 source；类型 core-logic)：移除了 miles\_validate\_args 中阻止 PPO 使用 bridge 模式的断言，是功能开启的入口点。
- tests/fast/utils/test\_arguments.py (模块 参数验证；类别 test；类型 test-coverage；符号 test\_bridge\_mode\_rejects\_critic, test\_bridge\_mode\_accepts\_critic)：更新了相关单元测试，以验证 bridge 模式现在可以接受 critic，并确认了功能行为的变更。
- docs/advanced/lora.md (模块 LoRA 文档；类别 docs；类型 documentation)：更新了关于 PPO+Bridge LoRA 支持状态的文档，从“不支持”更正为“未经测试”，避免用户误解。

关键符号：\_hide\_critic\_value\_head\_from\_hf\_load

## 关键源码片段

### miles/backends/megatron\_utils/checkpoint.py

引入了关键的值头隐藏机制 \_hide\_critic\_value\_head\_from\_hf\_load，用于在 HF 权重加载时临时移除 critic 的 value head，避免映射冲突。

```
@contextmanager
def _hide_critic_value_head_from_hf_load(ddp_model):
    """
    在从 HuggingFace 加载权重时，临时隐藏模型中所有 `LinearForLastLayer` 类型的子模块（即 critic 的 value head）。
    这样做是为了防止 HF 的权重映射逻辑（如 `ColumnParallelMapping.hf_to_megatron`）尝试将这些 head
    映射到并不存在的 `lm_head.weight` 上，从而避免形状不匹配或错误的权重覆盖。
    实现方式是在进入上下文时通过 `delattr` 移除，退出时通过 `setattr` 恢复。
    """
    # 遍历模型的所有 chunk，找到类型为 LinearForLastLayer 的子模块（即 value head）
```

```
value_heads = [
    (chunk, name, head)
    for chunk in unwrap_model(ddp_model) # unwrap_model 处理 DDP 封装
    for name, head in chunk.named_children()
    if isinstance(head, LinearForLastLayer)
]
# 临时移除这些 head
for chunk, name, _ in value_heads:
    delattr(chunk, name)
try:
    yield
finally:
    # 恢复所有被移除的 head
    for chunk, name, head in value_heads:
        setattr(chunk, name, head)
```

## 评论区精华

从提供的审核意见来看，此 PR 的讨论较少。主要收到一位 bot 的自动评论，以及两位审核者（Zhichenzzz, yushengsu-thu）的批准（APPROVED）。未发现具体的技术讨论线程，表明该变更的方案和实现较为清晰直接，且经过了快速审核。

- 暂无高价值评论线程

## 风险与影响

- 风险：

1. 值头隐藏机制的正确性：\_hide\_critic\_value\_head\_from\_hf\_load 依赖于 delattr 和 setattr 动态修改模型结构。虽然在 try-finally 中保证了恢复，但需要确保在多进程（如 TP/PP）环境下此操作是安全的，且不会与其他权重加载或初始化过程冲突。当前实现是在主进程 ddp\_model 上操作，但 unwrap\_model 处理了分布式封装，风险可控。
2. 新旧模式行为一致性：修改后，bridge 模式和 raw 模式构建 critic 的方式在逻辑上对齐（均为解耦并使用 value head）。但需注意 bridge 模式下的 value head 权重是全新初始化的，而非从 HF checkpoint 加载（因为 HF 模型没有 value head），这与 raw 模式行为一致，但应在文档中明确说明。
3. PPO + Bridge LoRA 组合：根据 docs/advanced/lora.md 的更新，此组合是“未经测试”的。虽然当前实现使 critic 得以构建，但 critic 路径并未集成 LoRA 适配器。若用户误用，可能导致意料之外的行为。此风险通过文档更新进行了明确提示。
4. 文档一致性：PR 同时更新了三处文档，需确保新旧描述完全一致且准确，特别是从“不支持”到“支持”再到“未经测试”的转变逻辑清晰。

- 影响：

1. 用户影响：对于使用 PPO 训练的用户，现在可以选择 --megatron-to-hf-mode bridge，这提供了更大的灵活性，尤其是在从 HF 模型迁移或利用 bridge 提供的特定模型支持时。用户不再需要因为 critic 不支持而被迫使用 raw 模式。

2. 系统影响：扩展了 Miles 框架中 PPO 训练的功能路径。变更主要集中在 Megatron 后端的模型构建和检查点加载模块，影响范围相对独立，不涉及核心训练循环或调度逻辑。
3. 团队影响：为团队提供了在 PPO 训练中使用 bridge 模式的官方支持路径。更新的文档（特别是 LoRA 页面）有助于避免团队成员误判 PPO+Bridge LoRA 的支持状态。修改后的单元测试确保了此功能的持续集成覆盖。 - 风险标记：值头隐藏机制正确性，未经测试的 PPO+Bridge LoRA 组合，行为一致性（value head 初始化）

## 关联脉络

- PR #2764 perf(megatron): keep policy logits in model precision: 此 PR 也涉及 miles/backends/megatron\_utils/model.py 和 loss 计算路径的修改，虽然目标不同（性能优化 vs 功能支持），但同属于 Megatron 后端训练路径的改动，可能关注相似的模型构建与精度处理逻辑。