

PR #7264 完整报告

verl-project/verl

[ckpt, megatron] fix: megatron save checkpoints with strict false when vanilla_bridge is false

合并时间: 2026-08-05 10:31

原文链接: <http://prhub.com.cn/verl-project/verl/pull/7264>

执行摘要

- 一句话: 修复 Megatron 非 vanilla 路径保存 ckpt 时忽略 strict 配置
- 推荐动作: 值得快速阅读即可: 单行改动澄清了 Megatron-Bridge 保存路径的 strict 透传设计。关注两点: 一是 checkpoint_config.strict 的默认值及其文档语义, 二是后续是否应为 strict=False 的导出场景补充测试。总体属于低风险修复。

功能与动机

PR body 仅有模板内容, 未提供额外背景; commit 信息为“fix megatron save checkpoints with strict false”。结合代码推断: 当用户配置 `checkpoint_config.strict=False` 时 (典型场景是 Megatron 训练的自定义模型结构与 HF 结构存在差异, 导出时允许缺失或多余的权重), 非 vanilla_bridge 路径的 `save_hf_weights` 调用未传 `strict`, bridge 默认按严格模式校验并抛出, 导致保存失败。该修复让 `strict` 配置对保存路径生效, 与 vanilla 路径保持一致。

实现拆解

实现拆解:

1. 问题定位: verl/utils/checkpoint/megatron_checkpoint_manager.py 的 `_save_model_as_hf_via_bridge` 方法, 负责通过 Megatron-Bridge 将模型权重导出为 HF 格式。该方法分为 vanilla_bridge、PEFT、普通导出三条路径。
2. 核心改动: 将普通导出分支的调用由 `self.bridge.save_hf_weights(self.model, hf_ckpt_path)` 改为 `self.bridge.save_hf_weights(self.model, hf_ckpt_path, strict=self.checkpoint_config.strict)`, 把 checkpoint 配置中的 `strict` 语义透传给 bridge, 使 `strict=False` 时允许权重不完全匹配。
3. 影响范围: 改动仅影响非 vanilla_bridge、非 PEFT 的保存路径; vanilla 路径与 PEFT 路径不受影响。该行为依赖 `checkpoint_config.strict` 的默认值, 若默认 True 则原有行为完全不变。
4. 配套改动: 本次无测试、配置或文档变更, 属于单行修复。

关键文件:

- verl/utils/checkpoint/megatron_checkpoint_manager.py (模块 检查点管理; 类别 source; 类型 core-logic; 符号 `_save_model_as_hf_via_bridge`): 唯一改动文件, 位于 Megatron checkpoint 保存主路径, 修复非 vanilla bridge 导出 HF 权重时 `strict` 配置失效的问题。

关键符号: `_save_model_as_hf_via_bridge`

关键源码片段

`verl/utils/checkpoint/megatron_checkpoint_manager.py`

唯一改动文件，位于 Megatron checkpoint 保存主路径，修复非 vanilla bridge 导出 HF 权重时 strict 配置失效的问题。

```
def _save_model_as_hf_via_bridge(self, hf_ckpt_path: str):
    """Save model weights through megatron-bridge."""
    if self.vanilla_bridge:
        # vanilla 路径：直接调用 bridge 保存完整权重（带扩展参数）
        self.bridge.save_weights(self.model, hf_ckpt_path, **self._get_bridge_extended_args())
    else:
        if self.peft_cls is not None:
            # PEFT 场景：只导出 adapter，与基础权重解耦
            hf_adapter_ckpt_path = os.path.join(hf_ckpt_path, "adapter")
            self.bridge.save_hf_adapter(self.model, hf_adapter_ckpt_path, self.peft_cls)
            log_with_rank(
                f"Saved HF PEFT adapter checkpoint to {hf_adapter_ckpt_path}",
                rank=self.rank,
                logger=self.logger,
                log_only_rank_0=True,
            )
        else:
            # 普通 HF 权重导出：透传 strict 配置，使 strict=False 时允许
            # 权重结构与 HF 不完全匹配（修复前未传 strict，强制严格校验，
            # 自定义 Megatron 结构模型导出会失败）
            self.bridge.save_hf_weights(self.model, hf_ckpt_path, strict=self.checkpoint_config.strict)
```

评论区精华

该 PR 没有任何 review 评论或 issue 讨论。唯一的 review 记录来自合并者 wuxibin89，状态为 DISMISSED（无正文），说明该改动未经实质性评讨论即被合并，可能作为低风险单行修复直接通过。

- 暂无高价值评论线程

风险与影响

- 风险：
 1. strict 语义反转风险：改动前 `save_hf_weights` 未传 `strict`，`bridge` 走默认严格行为；改动后行为取决于 `checkpoint_config.strict` 的默认值。若默认值非 `True`，则原本会因权重不匹配而失败的保存会静默成功，可能产出权重不完整的 checkpoint。
 2. 覆盖面有限：只影响非 `vanilla_bridge`、非 PEFT 的普通导出分支；`vanilla` 路径仍走 `save_weights`，两者 `strict` 语义仍需保持一致。

3. 缺少测试：本次没有新增针对 `strict=False` 导出行为的测试（如结构不匹配模型的导出用例），该路径主要依赖集成环境验证。
4. 影响面：改动在 `megatron_checkpoint_manager.py` 保存主路径上，但风险限定在该单行调用上。
 - 影响：对用户：修复了 Megatron 后端使用非 vanilla bridge 且配置 `strict=False` 时的 checkpoint 导出失败问题，影响使用自定义结构模型（如与 HF 结构不一致的模型）做导出的用户。对系统：改动极小，运行时行为仅在 `strict` 配置生效时改变，无性能影响。对团队：该修复补齐了 vanilla 与非 vanilla 路径在 `strict` 语义上的一致性，与近期 Megatron 导出链路的一系列 PR（PP/VPP delta export、delta_sharded、generation config 保留）同属 checkpoint 质量加固方向。
- 风险标记：strict 默认值依赖，缺少测试覆盖，可能静默导出不完整权重

关联脉络

- PR #7223 [megatron] feat: PP/VPP support for the steady delta export: 同属 Megatron checkpoint 导出链路，改动 `delta_export.py` 与 `transformer_impl.py`，均在加固导出正确性。
- PR #7181 [megatron] feat: delta_sharded on Megatron-Bridge param mappings (TP+EP, hybrid-Mamba): 同一 Megatron-Bridge 权重导出 / 同步功能线，涉及 `delta_export` 与 checkpoint engine。
- PR #7199 [megatron] fix: preserve generation config during merge: 同为 Megatron 保存 / 合并链路 bugfix，关注导出产物完整性。