

PR #6842 完整报告

verl-project/verl

[ckpt] fix: write HF config before bridge save to fix shard inference

合并时间: 2026-06-25 10:23

原文链接: <http://prhub.com.cn/verl-project/verl/pull/6842>

执行摘要

- 一句话: 调整 HF config 写入顺序修复 Megatron bridge 断点分片推断
- 推荐动作: 值得精读的 bugfix 示例: 展示了如何通过调整顺序解决隐式依赖问题, 并暴露了分布式环境下容易被忽视的竞态条件。虽然改动量小, 但 root cause 分析和修复方式值得参考。建议在类似多阶段写入场景中注意资源读取的时序依赖。

功能与动机

修复 Issue #6821: 在 Megatron bridge 保存 checkpoints 时, 由于 HF config 在 bridge 保存之后才写入, bridge 在推断分片时使用了过时或缺失的 config, 导致 MoE 模型的部分 expert 参数 (如 `model.language_model.layers.{38..39}.mlp.experts.gate_up_proj`) 被遗漏, 保存的 checkpoint 不完整。

实现拆解

1. 调整 `_save_checkpoint` 方法中的步骤顺序: 在 `verl/utils/checkpoint/megatron_checkpoint_manager.py` 的 `save_checkpoint` 方法中, 将原先步骤 2 (保存 HF 格式模型权重 via bridge) 和步骤 3 (写入 HF config/tokenizer) 互换。
2. 添加分布式同步: 在写入 config 之后插入 `torch.distributed.barrier()`, 确保 rank 0 完成 config 写入后所有 rank 再进入 bridge 保存, 防止竞态条件。
3. 改动量小, 无 API 变更: 仅涉及单文件内约 10 行代码的顺序调整, 不改变任何接口签名或行为逻辑。
4. 手动验证: 在 8xH20 上使用 Qwen3.5-35B-A3B MoE GRPO 验证, 修复前缺失的 expert 参数在修复后正确保存。没有添加新的测试 (需要多 GPU Megatron bridge 环境)。

关键文件:

- `verl/utils/checkpoint/megatron_checkpoint_manager.py` (模块 检查点; 类别 source; 类型 core-logic): 唯一变更文件, 包含了所有修复逻辑: 调整 `_save_hf_config_and_tokenizer` 与 `_save_model_as_hf_via_bridge` 的执行顺序, 并添加 `barrier` 同步。

关键符号: `save_checkpoint`, `_save_hf_config_and_tokenizer`, `_save_model_as_hf_via_bridge`

关键源码片段

verl/utils/checkpoint/megatron_checkpoint_manager.py

唯一变更文件，包含了所有修复逻辑：调整 `_save_hf_config_and_tokenizer` 与 `_save_model_as_hf_via_bridge` 的执行顺序，并添加 `barrier` 同步。

```
# verl/utils/checkpoint/megatron_checkpoint_manager.py
# 在 save_checkpoint 方法中，调整后的保存顺序片段：

# —— 2. HF config / tokenizer (rank 0)


---


if self.should_save_hf_model:
    self._save_hf_config_and_tokenizer(local_path) # 先写 config
    torch.distributed.barrier() # 等待所有 rank 确认 config 已写入

# —— 3. Save model weights in HF format via bridge


---


if self.should_save_hf_model:
    hf_ckpt_path = get_hf_model_checkpoint_path(local_path)
    log_with_rank(f"Saving HF model checkpoint to {hf_ckpt_path} with bridge",
                  rank=self.rank, logger=logger)
    self._save_model_as_hf_via_bridge(hf_ckpt_path) # 后写权重，此时 config 已就绪
    log_with_rank(f"Saved bridge checkpoint to {hf_ckpt_path}",
                  rank=self.rank, logger=logger)
```

评论区精华

竞态条件风险：gemini-code-assist[bot] 指出 `_save_hf_config_and_tokenizer` 仅在 rank 0 上执行，其他 rank 会立即进入 `_save_model_as_hf_via_bridge` 并尝试读取 config，可能读到不完整或缺失的文件，产生竞态条件。建议添加 `torch.distributed.barrier()`。

决策结论：作者采纳建议，在第二个 commit 中添加了 `barrier` 调用，修复了竞态条件。wuxibin89 随后批准了该 PR。无未解决疑虑。

- HF config 写入顺序导致分片推断错误 (correctness): 采纳建议，在第 2 个 commit 中添加了 `torch.distributed.barrier()`。

风险与影响

- 风险：低风险：变更仅涉及同一文件内的步骤顺序调整和 `barrier` 添加，不改变任何计算逻辑、配置接口或 checkpoint 格式。风险主要在于新添加的 `barrier` 可能在进度已有依赖的情况下引入不必要的同步延迟，但 checkpoint 保存本身已是同步密集操作，影响可忽略。另外，如果 `_save_hf_config_and_tokenizer` 内部在 rank 0 外有副作用，`barrier` 可能掩盖错误，但根据当前实现风险极低。
- 影响：影响范围：仅影响 Megatron bridge 路径下保存 HF 格式 checkpoint 的流程，修复了 MoE 等具有分片推断需求的模型在保存时丢失部分权重的 bug。对用户透明，没有 API 或配置文件改动。影响程度：对使用 Megatron bridge 保存 checkpoint 的用户是重要修复，确保检查点完整性；其他路径不受影响。
- 风险标记：竞态条件修复，核心保存路径变更

关联脉络

- PR #6821 [Bug] Checkpoint-time shard inference diverges from runtime model state due to config write ordering in Megatron bridge path: 本 PR 直接修复该 Issue 报告的 bug。