

PR #7199 完整报告

verl-project/verl

[megatron] fix: preserve generation config during merge

合并时间: 2026-08-03 14:04

原文链接: <http://prhub.com.cn/verl-project/verl/pull/7199>

执行摘要

- 一句话: 修复 Megatron 合并丢失生成配置与多卡重复上传
- 推荐动作: 值得精读。核心看点有三个: 一是“缺失即回退、损坏即失败”的 fail-closed 错误处理策略, 以及把存在性检查与解析解耦的实现方式; 二是分布式合并器中 rank 0 专属写入区与集合通信的分工, rank 0 守卫如何同时避免并发写文件和并发上传; 三是把“顺带修到的独立 bug”拆成单独 Issue (#7226) 以便二分定位的做法。评审讨论 (ErenAta16) 对这些问题有清晰交锋, 适合作为模型合并与分布式落盘代码 review 的参考案例。

功能与动机

来自 Issue #7198: MegatronModelMerger 的 `save_hf_model_and_tokenizer` 只保存 `model_config`、`processor` 和 `tokenizer`, 从不加载 / 保存 `GenerationConfig`; 对 `generation` 设置与 `config.json` 不一致的模型 (如 Qwen 多 EOS token ID), 合并输出会静默丢失这些设置——这是 #1246/#1277 已为 FSDP/legacy 路径修复的同类故障在 Megatron 路径上的遗留。PR body 强调这不是 #1277 的重复修复: 该 PR 只覆盖 FSDP/legacy 合并路径, 分布式 Megatron 合并器是后来新增的, 并未调用 `generation-config` 保存逻辑。Issue #7226 则是在验证 #7198 时发现的分布式控制流问题: `merge_and_save` 在每个 rank 上都会执行, `merge` 分支里的 `upload_to_huggingface` 没有 rank 守卫, 导致所有 rank 并发上传同一份合并产物。

实现拆解

实现拆解 (按 3 个步骤):

1. 在 `BaseModelMerger` 抽取通用配置读写 (`verl/model_merger/base_model_merger.py`)
 - `patch_model_generation_config` 原内联的 `try/except OSError` 改为调用新抽出的 `load_generation_config()`: 先用 `os.path.isfile` 判定文件是否存在, 缺失时返回 `None` (可选告警), 存在时才调用 `GenerationConfig.from_pretrained` 解析。
 - 关键设计: 把“文件缺失”与“文件损坏”解耦。缺失是受支持的回退路径; 损坏则让 JSON 解析错误向上传播 (fail-closed), 不再被宽泛的 `except OSError` 当作缺失静默吞掉, 避免合并产物在无可见报错的情况下改变生成行为。
 - 新增 `save_generation_config(target_dir)`: 以 `warn_if_missing=False` 加载 (save 路径缺失属于正常情况, 不打印告警); 有配置则 `save_pretrained` 写入目标目录, 无配置则删除目标目录中残留的陈旧 `generation_config.json`, 保证产物与源一致, 且复用同一公共方法, FSDP 路径行为兼容。

2. 接入分布式 Megatron 合并路径 (verl/model_merger/megatron_model_merger.py) - save_hf_model_and_tokenizer: 在既有 if self.rank == 0: 块内、model_config.save_pretrained 之后紧接 self.save_generation_config(self.config.target_dir), 与 index、processor、tokenizer、output validation 同区写入。其他 rank 只写各自分配的 safetensors 分片并参与 all_reduce / all_gather_object 集合通信, 不会并发写该文件。- merge_and_save: hf_upload 分支改为 if self.config.hf_upload and self.rank == 0, 修复 #7226; 分片写入与同步逻辑仍由所有 rank 参与, 只有上传这一可选动作限定 rank 0。
3. CPU 单元测试 (tests/model_merger/test_output_validation_on_cpu.py) - 新增 4 个用例: 保留多 EOS 配置 (eos_token_id=[151643, 151645] + max_new_tokens=4096); 源缺失时清理陈旧目标文件且无告警输出; load 缺失时告警并返回 None; 源文件为非法 JSON 时抛 OSError 且不动目标陈旧文件。- 验证: pytest -q tests/model_merger/test_output_validation_on_cpu.py 14 passed, ruff check / ruff format --check / py_compile 均通过; 作者明确未在本本地运行过多卡 Megatron 合并实测。
4. 配套改动: 无 CLI/API 变更、无文档变更。合并产物的实际差异仅在于: 源 checkpoint 元数据含独立 generation config 时, 输出目录会新增 generation_config.json 文件。

关键文件:

- verl/model_merger/base_model_merger.py (模块 合并基类; 类别 source; 类型 data-contract; 符号 load_generation_config, save_generation_config, patch_model_generation_config): 核心改动所在: 将 generation config 的加载 / 保存抽成 BaseModelMerger 公共方法, 实现缺失回退与损坏报错 (fail-closed) 的语义区分, 并为 FSDP/Megatron 两条路径提供统一入口。
- verl/model_merger/megatron_model_merger.py (模块 合并流程; 类别 source; 类型 data-contract; 符号 save_hf_model_and_tokenizer, merge_and_save): 把 generation config 保存接入分布式 Megatron 合并的 rank 0 写入区, 并给 hf_upload 加上 rank 0 守卫, 分别修复 #7198 与 #7226。
- tests/model_merger/test_output_validation_on_cpu.py (模块 测试覆盖; 类别 test; 类型 test-coverage; 符号 test_save_generation_config_preserves_pretrained_values, test_save_generation_config_allows_missing_file, test_load_generation_config_warns_when_file_is_missing, test_save_generation_config_rejects_invalid_source): 4 个新增 CPU 用例覆盖多 EOS 保留、缺失清理、缺失告警与损坏拒绝四条关键路径, 是该行为契约的回归保障。

关键符号: load_generation_config, save_generation_config, patch_model_generation_config, save_hf_model_and_tokenizer, merge_and_save

关键源码片段

verl/model_merger/megatron_model_merger.py

把 generation config 保存接入分布式 Megatron 合并的 rank 0 写入区, 并给 hf_upload 加上 rank 0 守卫, 分别修复 #7198 与 #7226。

```
# all_reduce 汇总 total_size、all_gather_object 聚合 weight_map 之后,
```

```

# 非权重产物只在 rank 0 落盘，其余 rank 只写各自分片并参与集合通信。
if self.rank == 0:
    with open(target_dir / "model.safetensors.index.json", "w") as f:
        json.dump(
            {
                "metadata": {"total_size": numel},
                "weight_map": saves_indexes,
            },
            f,
            indent=4,
        )
    print(f"model saved to {target_dir} with {numel=}")

    self.model_config.save_pretrained(self.config.target_dir)
    # 新增：紧邻 model_config 等 rank 0 产物写入，保存独立预训练
    # generation_config，修复 #7198 —— 多 EOS 等仅存在于该文件的设置
    # 之前在 Megatron 合并产物中被静默丢弃
    self.save_generation_config(self.config.target_dir)

    processor = hf_processor(self.hf_model_config_path, trust_remote_code=self.config.
trust_remote_code)
    tokenizer = hf_tokenizer(self.hf_model_config_path, trust_remote_code=self.config.
trust_remote_code)
    if processor is not None:
        print(f"Saving processor to {self.config.target_dir}")
        processor.save_pretrained(self.config.target_dir)
    if tokenizer is not None:
        print(f"Saving tokenizer to {self.config.target_dir}")
        tokenizer.save_pretrained(self.config.target_dir)
    validate_hf_model_output(self.config.target_dir)

def merge_and_save(self):
    # ... (加载分布式 checkpoint 并对状态字典做合并、释放临时对象) ...
    if self.config.operation == "test":
        if not self.config.test_hf_dir:
            raise ValueError("test_hf_dir must be provided for test operation")
        self._validate_state_dict(merged_state_dict)
    elif self.config.operation == "merge":
        self.save_hf_model_and_tokenizer(merged_state_dict)
        # 修复 #7226：此前所有 rank 都会执行 upload_to_huggingface，
        # 同一份合并产物被多卡并发上传；现在上传限定 rank 0
        if self.config.hf_upload and self.rank == 0:
            self.upload_to_huggingface()
    else:
        raise ValueError(f"Unknown operation: {self.config.operation}")

```

评论区精华

核心交锋围绕三点：

1) Copilot 行内评论指出 `load_generation_config` 在保存路径每次缺失都打印告警，对不带独立配置的模型是行为变更（与现状 Megatron 路径不符），通过新增 `warn_if_missing` 参数解决：patch 路径保留原告警，save 路径传入 `False` 抑制告警。2) ErenAta16 发现 diff 中混入与标题无关的 rank 0 上传守卫，评估为真实且更严重的 bug——多卡下每个 rank 都会并发上传同一份合并 checkpoint，建议拆出并显式记录；作者随后拆分到独立 Issue #7226，并在 PR 描述中写明。3) ErenAta16 认可错误处理收窄：先查文件存在再解析，使损坏配置从被 `except OSError` 吞掉变为传播解析错误，是更优行为；并强调“缺失是回退、损坏即抛错”这一策略需要显式文档化，作者已在描述中补充。此外，作者确认 `save_generation_config` 位于既有 `if self.rank == 0` 块内，与 `model_config/processor/tokenizer` 同批写入，不存在多 rank 并发写同一文件的竞态；wuxibin89 最终 approve。

- 保存路径缺失时的告警噪音 (design): 增加 `warn_if_missing` 参数：patch 路径保留原告警，save 路径以 `warn_if_missing=False` 调用，将缺失视为预期情况静默处理。
- diff 中混入与标题无关的 rank 0 上传守卫 (design): 拆分为独立 Issue #7226 并在 PR 描述中显式记录；作者确认该守卫同时避免并发写 `generation_config.json` 和并发上传。
- 损坏配置从静默忽略改为显式报错 (correctness): 双方认可 fail-closed 是更优行为；作者在 PR 描述中明确“缺失是受支持的回退，损坏则传播解析错误”。
- `save_generation_config` 是否与 rank 0 产物写入同区 (question): 作者确认该调用位于既有 `if self.rank == 0` 块内、紧邻 `model_config.save_pretrained`；审阅者接受确认，无进一步问题。

风险与影响

- 风险：1) fail-closed 行为收紧：损坏的源 `generation_config.json` 从被 `except OSError` 静默当作缺失，变为合并过程直接抛错 (`OSError: not a valid JSON file`)。这是刻意设计，但若历史 checkpoint 中恰好有坏文件，合并流程会失败，影响面集中在 `model_merger` 路径。2) 分布式控制流：hf_upload 依赖 `self.rank` 守卫，单卡时 `rank == 0` 恒真、无回归；但作者明确未跑过多卡 Megatron 合并实测，rank 守卫的真实分布式行为尚未在本地验证，依赖 CI 或后续回归。3) 目标目录清理：源配置缺失时 `save_generation_config` 会删除输出目录中残留的 `generation_config.json`，若用户预先放置自定义配置会被移除——这与“产物与源一致”的契约一致，但值得在使用文档中说明。4) FSDP 路径窄幅变更：`patch_model_generation_config` 改用 `load_generation_config` 后，缺失文件的行为一致；但损坏文件从“告警并回退”变为“抛错”，是同一公共路径上的行为收紧。
- 影响：用户侧：启用 hf_upload 的多卡 Megatron 合并不再出现每 rank 并发上传同一份 checkpoint、Hugging Face 仓库被反复覆盖 / 限流的问题；Qwen 等多 EOS 模型合并产物保留独立 `generation config`，生成行为与源模型一致。系统侧：merge 路径在源含配置时新增一个产物文件 `generation_config.json`，无 API/CLI 变更；BaseModelMerger 成为 `generation config` 读写的单一入口，未来新增合并器后端可直接复用。团队侧：与 #7193 的输出完整性校验共同完善 `model_merger` 合并链路的数据契约；本 PR 附带 4 个 CPU 单元测试覆盖核心语义，但多卡行为仍缺实测，是当前覆盖上的主要缺口。
- 风险标记：fail-closed 行为变更，分布式 rank 控制流，数据契约变更，缺多卡实测，目标目录陈旧文件清理

关联脉络

- PR #7193 [ckpt, model] fix: validate model merger outputs: 与本 PR 修改同一批文件（`base_model_merger.py`、`megatron_model_merger.py`、`test_output_validation_on_cpu.py`），同属 `model_merger` 链路的事实性加固，共同完善合并产物输出完整性与数据契约。