PR #7234 完整报告

verl-project/verl

[rollout, sglang] fix: keep SGLang LoRA-free when model.lora.merge=True

合并时间：2026-08-04 13:51

原文链接：http://prhub.com.cn/verl-project/verl/pull/7234

执行摘要

- 一句话：修复 SGLang 在 LoRA merge 模式下仍启用 adapter 的 bug
- 推荐动作：值得精读。这是一个典型的 bugfix 与设计收敛案例：通过引入单一谓词消除两个后端的行为漂移，并强调两个配置块的不同步问题，对理解 verl 中 LoRA 在 rollout 侧的处理很有帮助。

功能与动机

PR body 说明：当 model.lora.merge=True 时，训练引擎将 LoRA 权重合并进基础权重，导出完整 HF 键参数（peft_config=None），engine_workers.update_weights 走普通权重同步路径，不会在 SGLang 注册 adapter。但 SGLangHttpServer 仍以 model_config.lora_rank > 0 作为代理条件，在 launch_server 传入 enable_lora=True、在 generate 附加 lora_path，导致 merge 模式请求一个从未加载的 adapter。同时 vLLM 后端已有 `if model.lora.merge: lora_rank = 0` 逻辑，两个后端语义不一致。

实现拆解

1. 新增统一谓词：在 verl/workers/rollout/sglang_rollout/utils.py 新增 lora_served_as_adapter(model_config) 函数，同时检查 model_config.lora_rank 和 model_config.lora.get("rank") 两套配置块，并排除 merge=True 的情况，返回是否应以 adapter 模式启动 SGLang。这是整个变更的核心抽象，使 LoRA 决策在 rollout 侧只回答一个问题：“引擎是否必须持有 adapter”。
2. 改造 SGLang 服务器：在 verl/workers/rollout/sglang_rollout/async_sglang_server.py 中，将 launch_server 中设置 enable_lora 等参数的分支条件从 self.model_config.lora_rank > 0 改为 self.lora_as_adapter；将 generate 中附加 lora_path 的条件同样改为 self.lora_as_adapter。lora_as_adapter 属性从内联表达式改为委托给新函数，消除逻辑漂移。
3. 补充单元测试：新增 tests/workers/rollout/rollout_sglang/test_lora_merge_mode_on_cpu.py，通过 _StubModelConfig 模拟 HFModelConfig 的两种配置形态，覆盖 no_lora、megatron_merge、megatron_adapter、fsdp_merge、fsdp_adapter、merge_absent_defaults_to_adapter、merge_without_lora 等场景，并验证 sleep 的释放标签逻辑。测试文件被 cpu_unit_tests.yml 自动收集。

关键文件：

- verl/workers/rollout/sglang_rollout/utils.py（模块 推理服务；类别 source；类型 core-logic；符号 lora_served_as_adapter）：新增核心谓词 lora_served_as_adapter，统一 LoRA adapter 判断逻辑，是本次修复的核心抽象。
- verl/workers/rollout/sglang_rollout/async_sglang_server.py（模块 推理服务；类别 source；类型 dependency-wiring；符号 lora_as_adapter, launch_server, generate）：修改 SGLang 服务端的 LoRA 分支条件，将 launch_server 和 generate 的行为与新的谓词对齐。
- tests/workers/rollout/rollout_sglang/test_lora_merge_mode_on_cpu.py（模块 推理服务；类别 test；类型 test-coverage；符号 _StubModelConfig, TestLoraServedAsAdapter, test_no_lora, test_megatron_merge）：新增 CPU 单元测试，覆盖两种配置块下 merge/adapter 检测逻辑，以及 sleep 释放标签的行为。

关键符号：lora_served_as_adapter, lora_as_adapter, launch_server, generate

关键源码片段

verl/workers/rollout/sglang_rollout/utils.py

新增核心谓词 lora_served_as_adapter，统一 LoRA adapter 判断逻辑，是本次修复的核心抽象。

```
# verl/workers/rollout/sglang_rollout/utils.py

def lora_served_as_adapter(model_config) -> bool:
    """判断 SGLang 是否应将 LoRA 作为 hot-swappable adapter 提供服务。

    HFModelConfig 携带两套互不同步的 LoRA 配置块：megatron 运行设置
    `model.lora.rank`（dict），fsdp 运行设置扁平的 `model.lora_rank`，
    所以必须同时检查两者才能确定 LoRA 是否启用。

    当 `model.lora.merge=True` 时，训练器将 adapter 合并进基础权重，并
    推送完整 HF 键权重更新（`peft_config=None`），SGLang 中永远不会加载
    adapter：此时引擎不能以 `enable_lora` 启动，请求也不能携带
    `lora_path`。
    """
    # 两个配置块任一启用即视为 LoRA 开启
    lora_enabled = model_config.lora_rank > 0 or model_config.lora.get("rank", 0) > 0
    # merge 模式下 adapter 已合并，不再作为 adapter 服务
    return lora_enabled and not model_config.lora.get("merge", False)
```

verl/workers/rollout/sglang_rollout/async_sglang_server.py

修改 SGLang 服务端的 LoRA 分支条件，将 launch_server 和 generate 的行为与新的谓词对齐。

```
# verl/workers/rollout/sglang_rollout/async_sglang_server.py

    @property
    def lora_as_adapter(self) -> bool:
        """是否将 LoRA 作为 adapter 服务，委托给工具函数统一判断。"""
```

```
        return lora_served_as_adapter(self.model_config)

    # launch_server 中原来的 `if self.model_config.lora_rank > 0:` 改为：
    if self.lora_as_adapter:
        args.update(
            {
                "enable_lora": True,
                "max_lora_rank": self.model_config.lora_rank,
                "lora_target_modules": self.model_config.target_modules,
            }
        )

    # generate 中原来的 `if self.model_config.lora_rank > 0:` 改为：
    if self.lora_as_adapter:
        generate_request.lora_path = SGLANG_LORA_NAME
```

评论区精华

该 PR 没有 review 评论，仅有 issue 评论。主要讨论集中在代码格式：

- wuxibin89 要求按 CONTRIBUTING.md 格式化代码。jamesruio 回应 pre-commit 失败并非本 PR 引入，而是 main 分支上 verl/workers/engine/megatron/delta_export.py 已存在未格式化问题，并给出本地 ruff format --check 验证结果：该 PR 涉及的文件全部已格式化。
- ChangyiYang 表示确认没问题，要求 rebase。作者 rebase 后合入。
- 代码格式检查（pre-commit）失败 (style): 确认格式问题与 PR 无关，main 分支自身存在该问题。
- Rebase 请求 (other): 作者 rebase 后合入。

风险与影响

- 风险：主要风险是改变 SGLang 服务器启动和请求行为，可能影响现有 LoRA adapter 模式的用法。由于 lora_served_as_adapter 在 merge 键缺失时默认视为 adapter（与旧行为一致），旧配置不会受影响；而 merge 模式现在能正确保持 LoRA-free。但该函数同时读取两套配置块，未来若新增配置块可能遗漏。另外 launch/generate 的 e2e 行为依赖真实 SGLang 服务器，现有测试仅覆盖 CPU 单元层，缺少端到端验证。整体风险较低，因为与 vLLM 行为对齐，且有单元测试保护。
- 影响：影响使用 SGLang rollout 且配置了 LoRA 的用户：merge 模式现在能正常工作，不再请求不存在的 adapter；adapter 模式行为不变。对团队而言，改动集中在 sglang_rollout 模块，统一了 LoRA 判断逻辑，后续维护更简单。对系统整体影响小，不涉及配置、API 或部署变更。
- 风险标记：核心路径变更，配置兼容性，缺少 e2e 测试

关联脉络

- PR #7181 [megatron] feat: delta_sharded on Megatron-Bridge param mappings (TP+EP, hybrid-Mamba): 该 PR 涉及 SGLang 后端 delta 权重同步（delta_loader），与本 PR 的 LoRA 权重更新路径同属 rollout 权重同步机制，且都影响 SGLang 服务器行为。
- PR #7179 [vllm] refactor: clean up weight sync: 该 PR 重构了 vLLM 权重同步，并已包含 model.lora.merge 判断逻辑；本 PR 使 SGLang 与 vLLM 语义对齐。