PR #6435 完整报告

verl-project/verl

[docker] chore: upgrade sglang to 0.5.12

合并时间：2026-05-27 19:52

原文链接：http://prhub.com.cn/verl-project/verl/pull/6435

执行摘要

- 一句话：升级 sglang 到 0.5.12 并适配代码
- 推荐动作：该 PR 核心目标（升级 sglang）合理，但 Dockerfile 中的版本参数存在明显错误，建议在合并前修复 CUDA 和 TransformerEngine 版本。agent_loop 的 prompt 长度控制和统一 padding 方法设计良好，值得独立审查。attention 后端默认值的变更谨慎且有明确版本判断，可减少新版本兼容问题。整体建议在修复 Dockerfile 版本后批准。

功能与动机

SGLang 0.5.12 带来了新功能和修复，为了保持兼容并利用改进，版本需要升级。PR 描述为“Upgrade sglang to 0.5.12, test image”。同时解决了新版 sglang 中 FA3 CUDA graph 捕获 bug 导致的兼容问题（#22800）。

实现拆解

1. 更新 Docker 基础镜像及依赖元数据：docker/Dockerfile.stable.sglang 将 FROM 改为 lmsysorg/sglang:v0.5.12，新增 CUDA_VERSION、TRANSFORMER_ENGINE_VERSION 等构建参数，并重写 apt/pip 安装步骤以支持更灵活的版本控制。
2. 修复 sglang 内核包兼容性：verl/workers/rollout/sglang_rollout/sglang_rollout.py 中 _set_envs_and_config 将原来直接 assert sgl-kernel 改为先尝试 sglang_kernel，再回退到 sgl_kernel，以兼容新版 sglang 的内核包重命名。
3. 调整 attention 后端默认策略：verl/workers/rollout/sglang_rollout/async_sglang_server.py 中 launch_server 将硬编码的 attention_backend='fa3' 改为根据 sglang 版本选择：>=0.5.12 时默认 flashinfer，否则用 fa3（因新版本 FA3 CUDA-graph 有 bug）。同时将 mm_attention_backend 从硬编码 'fa3' 改为可由 engine_kwargs 覆盖，默认 None 让 sglang 自动选择。
4. 强化 agent_loop 的 prompt 长度预检：verl/experimental/agent_loop/agent_loop.py 在 apply_chat_template 返回前增加 prompt_ids 长度检查，超过 rollout.prompt_length 时对纯文本模式左截断，对多模态模式直接报错（防止占位符错位）。
5. 提取统一 token padding 工具方法：将 _agent_loop_postprocess 中原有的内联 padding 逻辑抽取为 _pad_token_ids 方法，统一处理左右填充和维度扩展，提高可维护性。

6. 配套测试与 CI 更新：更新多个测试文件（test_special_server_adapter.py、test_multi_modal.py、test_basic_agent_loop.py 等）以使用 normalize_token_ids 处理 tokenizer 输出，增加多模态测试的 prompt_length 配置；更新 .github/workflows/sgl.yml 等 CI 文件中的镜像标签。

关键文件：

- verl/experimental/agent_loop/agent_loop.py（模块 Agent 循环；类别 source；类型 core-logic；符号 _pad_token_ids）：核心重构：新增 prompt 长度截断逻辑和统一 _pad_token_ids 方法，影响 agent_loop 所有调用者。
- verl/workers/rollout/sglang_rollout/sglang_rollout.py（模块 SGLang rollout；类别 source；类型 core-logic；符号 _set_envs_and_config）：修复 sglang 内核包兼容性，确保在 sglang 0.5.12 中能正确加载内核。
- verl/workers/rollout/sglang_rollout/async_sglang_server.py（模块 SGLang 服务器；类别 source；类型 core-logic；符号 launch_server）：调整 attention 后端默认策略，并解耦 mm_attention_backend，提升新版本稳定性。
- docker/Dockerfile.stable.sglang（模块 Docker 部署；类别 infra；类型 infrastructure）：构建新版本 sglang 镜像的基础，包含版本参数化和所有依赖变更。
- tests/checkpoint_engine/test_special_server_adapter.py（模块 测试；类别 test；类型 test-coverage）：测试适配：使用 normalize_token_ids 处理 tokenizer 输出，确保与 agent_loop 改动的兼容性。

关键符号：_pad_token_ids, apply_chat_template, _set_envs_and_config, launch_server

关键源码片段

verl/experimental/agent_loop/agent_loop.py

核心重构：新增 prompt 长度截断逻辑和统一 _pad_token_ids 方法，影响 agent_loop 所有调用者。

```
# verl/experimental/agent_loop/agent_loop.py
# 在 apply_chat_template 返回之前，新增 prompt 长度检查与截断
prompt_length = self.rollout_config.prompt_length
if len(prompt_ids) > prompt_length:
    if images or videos or audios:
        raise ValueError(
            f"Multimodal prompt produced {len(prompt_ids)} tokens, exceeding "
            f"rollout.prompt_length={prompt_length}. Truncating multimodal token "
            f"sequences corrupts vision/audio feature alignment. Reduce the "
            f"multimodal input size or increase rollout.prompt_length."
        )
    logger.warning(
        "Prompt of %d tokens exceeds rollout.prompt_length=%d; left-truncating.",
        len(prompt_ids), prompt_length,
    )
    prompt_ids = prompt_ids[-prompt_length:]

# 新提取的填充方法，替代内联 tokenizer.pad 调用
```

```
def _pad_token_ids(
    self,
    tokens: list[int],
    *,
    max_length: int,
    padding_side: str,
    return_attention_mask: bool,
) -> dict[str, torch.Tensor]:
    """Right/left pad a flat list of token ids to a (1, max_length) tensor."""
    self.tokenizer.padding_side = padding_side
    padded = self.tokenizer.pad(
        {"input_ids": tokens},
        padding="max_length",
        max_length=max_length,
        return_tensors="pt",
        return_attention_mask=return_attention_mask,
    )
    if padded["input_ids"].dim() == 1:
        padded["input_ids"] = padded["input_ids"].unsqueeze(0)
    if return_attention_mask:
        padded["attention_mask"] = padded["attention_mask"].unsqueeze(0)
    return padded
```

评论区精华

1. Dockerfile 版本验证问题：gemini-code-assist[bot] 指出 CUDA_VERSION=13.0.2、TRANSFORMER_ENGINE_VERSION=v2.15、torchcodec --index-url=cu130 等版本不存在，且使用了个人 fork 依赖，将导致构建失败。作者未公开回应，但 PR 最终由 wuxibin89 批准，可能已知风险或在后续提交中修复。| 2. agent_loop 空白 response_ids 处理：wuxibin89 在 review 中要求 _pad_token_ids 不应处理空 response_ids，因为不应出现空的情况。作者 ETOgaosion 承诺修复文档字符串。

- Dockerfile 版本与依赖正确性 (correctness): 作者未公开回复，但 PR 最终被批准，可能已在实际构建中验证或后续修复。建议在合并前修正版本参数。
- agent_loop 空白 response_ids 处理 (design): 作者承诺修正文档，逻辑上未实际改动，但约定不再处理空 response_ids。

风险与影响

- 风险：
  1. Dockerfile 构建失败风险：CUDA_VERSION=13.0.2 和 TRANSFORMER_ENGINE_VERSION=v2.15 在官方源中不存在，直接使用会导致构建失败；torchcodec 的 cu130 索引也不存在；个人 fork 依赖 (ETOgaosion/qwen-vl-utils) 不稳定。| 2. 默认 attention backend 变更影响：从 FA3 切换到 flashinfer 可能改变推理速度和 token 分布，需要用户验证效果。| 3. sglang_kernel 回退可能不完整：若两个包都缺失，原行为是直接失败，现在回退后仍会失败，但错误信息更清晰；部分环境可能仍需要手动安装。| 4. agent_loop prompt 截断风险：新增的硬报错可能打破现有使用多

模态的正常流程；左截断可能丢弃系统提示等重要内容。| 5. 缺乏对新版本全面集成测试：CI 仅更新了镜像标签，未覆盖所有 sglang 0.5.12 新功能。 - 影响：用户：升级后必须使用新的 Docker 镜像，并注意 attention_backend 默认值变更；agent_loop 用户需检查 prompt 长度配置，多模态场景需要更大的 prompt_length 或减小输入。系统：构建脚本需要正确的依赖版本，目前版本设置错误会导致失败。团队：需要修复Dockerfile中的版本参数才能用于生产，agent_loop 的改动提高了代码质量但需关注行为变化。 - 风险标记：Dockerfile 版本不存在，核心逻辑变更（agent_loop），默认attention backend 切换，个人 fork 依赖不稳定，prompt 截断行为变化

关联脉络

- 暂无明显关联 PR