PR #24120 完整报告

sgl-project/sglang

[diffusion] CI: change ground truth upload path and improve publish script

合并时间：2026-04-30 12:26

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24120

执行摘要

- 一句话：优化 diffusion CI 的 GT 上传路径并增加增量发布逻辑
- 推荐动作：建议精读 publish_diffusion_gt.py 中的 git_blob_sha + filter_changed_files 增量发布模式，该模式可迁移至其他 CI 数据上传场景。工作流参数化 publish_target_dir 的设计也值得在其他 CI 中复用。

功能与动机

PR body 未提供详细动机，但从变更内容推断：(1) 将 sglang 生成的 ground truth 与 official 生成的 GT 分离到独立子目录，便于管理；(2) 引入增量发布机制，减少不必要的 API 调用和上传数据量，提升 CI 稳定性（尤其在频繁并发推送时）。

实现拆解

1. 重写发布脚本增量逻辑(scripts/ci/utils/diffusion/publish_diffusion_gt.py):
   - 新增 git_blob_sha() 计算本地文件的 Git blob SHA 值。
   - 新增 get_remote_blob_shas() 通过 GitHub Contents API 获取远程目标目录中所有文件的 SHA。
   - 新增 filter_changed_files() 对比本地与远程 SHA，只返回需要更新的文件。
   - 在 publish() 主流程的 commit 重试循环中，先获取远程 blob SHA，过滤出变更文件，再调用 create_blobs()；若无变更则提前返回。
2. 调整默认目标路径: DEFAULT_TARGET_DIR 从 diffusion-ci/consistency_gt 改为 diffusion-ci/consistency_gt/sglang_generated。
3. 更新 CI 工作流(.github/workflows/diffusion-ci-gt-gen.yml):
   - 新增 publish_target_dir 输入参数，默认值为 diffusion-ci/consistency_gt/sglang_generated。
   - 将以往的 output_name 条件拼接改为固定使用 --target-dir "${{ env.PUBLISH_TARGET_DIR }}"，简化调用逻辑。
4. 同步测试文件:
   - python/sglang/multimodal_gen/test/test_utils.py: 将 SGL_TEST_FILES_CONSISTENCY_GT_BASE 常值指向 sglang_generated 子目录，并从查找列表中去掉了重复的通用 Base URL。

- python/sglang/multimodal_gen/test/server/test_server_common.py: 更新错误提示中的路径为 diffusion-ci/consistency_gt/sglang_generated/。

关键文件：

- scripts/ci/utils/diffusion/publish_diffusion_gt.py（模块 CI 脚本；类别 infra；类型 infrastructure；符号 git_blob_sha, get_remote_blob_shas, filter_changed_files）：核心变更文件：实现增量发布逻辑（git_blob_sha、get_remote_blob_shas、filter_changed_files）并修改默认目录。
- python/sglang/multimodal_gen/test/server/test_server_common.py（模块 测试服务器公共；类别 test；类型 test-coverage）：更新一致性校验失败时的错误提示路径，指向新子目录。
- python/sglang/multimodal_gen/test/test_utils.py（模块 测试工具；类别 test；类型 test-coverage）：更新常量和搜索路径：将 CONSISTENCY_GT_BASE 指向新子目录，并简化搜索顺序。
- .github/workflows/diffusion-ci-gt-gen.yml（模块 CI 工作流；类别 infra；类型 infrastructure）：CI 工作流增加 publish_target_dir 参数并统一使用 --target-dir 调用发布脚本。

关键符号：git_blob_sha, get_remote_blob_shas, filter_changed_files, publish

关键源码片段

scripts/ci/utils/diffusion/publish_diffusion_gt.py

核心变更文件：实现增量发布逻辑（git_blob_sha、get_remote_blob_shas、filter_changed_files）并修改默认目录。

```
import hashlib
from urllib.error import HTTPError

def git_blob_sha(content: bytes) -> str:
    """计算 Git blob 对象的 SHA1 值 (用于对比远程文件是否变化)."""
    header = f"blob {len(content)}\0".encode()
    return hashlib.sha1(header + content).hexdigest()

def get_remote_blob_shas(repo_owner, repo_name, target_dir, token):
    """通过 GitHub Contents API 获取目标目录下所有文件的远程 SHA."""
    url = (
        f"https://api.github.com/repos/{repo_owner}/{repo_name}/contents/"
        f"{target_dir}?ref={BRANCH}"
    )
    try:
        response = make_github_request(url, token)
    except HTTPError as e:
        if e.code == 404:
            return {}
        raise
    entries = json.loads(response)
```

```
    return {
        item["path"]: item["sha"]
        for item in entries
        if item.get("type") == "file" and "sha" in item
    }

def filter_changed_files(files, remote_blob_shas):
    """过滤出与远程 SHA 不匹配的文件 (即需要更新的)."""
    return [
        (path, content)
        for path, content in files
        if remote_blob_shas.get(path) != git_blob_sha(content)
    ]

def publish(source_dir, target_dir=None):
    target_dir = target_dir or DEFAULT_TARGET_DIR
    # ... 前置验证 ...
    max_retries = 5
    for attempt in range(max_retries):
        try:
            branch_sha = get_branch_sha(REPO_OWNER, REPO_NAME, BRANCH, token)
            tree_sha = get_tree_sha(REPO_OWNER, REPO_NAME, branch_sha, token)
            # --- 新增：增量检测 ---
            remote_blob_shas = get_remote_blob_shas(REPO_OWNER, REPO_NAME, target_dir,
            token)
            changed_files = filter_changed_files(files_to_upload, remote_blob_shas)
            if not changed_files:
                print("No image changes to publish.")
                return
            # 只对 changed_files 创建 blob
            tree_items = create_blobs(REPO_OWNER, REPO_NAME, changed_files, token)
            # ... 后续 tree/commit 逻辑 ...
        except Exception as e:
            # 异常处理 (rate limit / permission / 并发重试 )
            ...
```

评论区精华

无人工审阅评论。仅有一条 gemini-code-assist[bot] 的 quota 提醒，不涉及技术讨论。

- 暂无高价值评论线程

风险与影响

- 风险：回归风险(低)：测试常量 SGL_TEST_FILES_CONSISTENCY_GT_BASE 改为仅指向 sglang_generated 子目录，若存在仍依赖旧根目录的测试可能找不到文件；但所有相关测试已同步更新，风险可控。 兼容风险(低)：CI 工作流新增 publish_target_dir 参数，旧版调用未设置时使用默认值，与现有行为兼容。增量发布逻辑完全替换了原有的 ' 直接创建所有 blob' 方式，若 get_remote_blob_shas 因网络或权限失败会触发 HTTPError 并优雅降

级（跳过 / 返回空字典），不会中断 CI。 性能风险(无)：增量发布反而减少 API 调用，对 CI 整体有利。

- 影响：用户：无直接影响，变更完全限定于 CI 基础设施和测试代码。 系统：发布脚本不再向 remote 重复上传未变化的文件，减轻 GitHub API 压力，降低 rate limit 触发概率；多 job 并发时冲突可能性减小。 团队：需要确保后续添加 diffusion 测试时，GT 文件路径使用新常量。
- 风险标记：CI 路径变更，缺少审阅讨论

关联脉络

- 暂无明显关联 PR