

PR #23422 完整报告

sgl-project/sglang

[diffusion] support custom output folder name in GT generation workflow

合并时间: 2026-04-22 11:18

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/23422>

执行摘要

- 一句话: 为扩散模型 GT 生成 CI 工作流添加自定义输出文件夹名称支持, 提升灵活性。
- 推荐动作: 该 PR 值得 CI/ 基础设施维护者精读, 以了解如何参数化 CI 工作流和配套脚本。
关注点包括: 环境变量与输入参数的联动设计、默认行为的保持策略、以及发布脚本的参数化重构模式。对于核心模型开发工程师, 可快速浏览以知悉 CI 能力扩展。

功能与动机

根据 PR 描述, 动机是“允许用户为生成的 GT 图像指定自定义输出文件夹名称 (影响本地文件夹、产物名称和远程发布目标)”。这解决了现有工作流输出路径固定、缺乏灵活性的问题, 便于用户在同一 CI 运行中区分不同测试集或进行并行测试。

实现拆解

- 工作流输入扩展: 在 `.github/workflows/diffusion-ci-gt-gen.yml` 中, 新增 `output_name` 输入参数, 并设置环境变量 `OUTPUT_NAME` (默认为 `diffusion-ci-outputs`)。
- 本地路径与产物命名调整: 将工作流中所有 `--out-dir` 参数、产物上传的 `name` 和 `path` 字段, 从硬编码的 `diffusion-ci-outputs` 替换为动态的 `${{ env.OUTPUT_NAME }}`。
- 发布脚本参数化: 修改 `scripts/ci/utls/diffusion/publish_diffusion_gt.py`, 将硬编码的远程目标目录 `TARGET_DIR` 改为可配置的 `DEFAULT_TARGET_DIR`, 并为 `collect_images` 和 `publish` 函数添加 `target_dir` 参数, 允许通过命令行 `--target-dir` 指定。
- 远程发布路径联动: 在工作流的发布步骤中, 当 `output_name` 非空时, 动态构建 `--target-dir diffusion-ci/{output_name}` 参数传递给发布脚本, 实现自定义远程路径; 否则使用默认路径 `diffusion-ci/consistency_gt`。
- 无测试或文档配套: 此变更仅涉及 CI 基础设施脚本, 未添加或修改测试用例或文档。

关键文件:

- `.github/workflows/diffusion-ci-gt-gen.yml` (模块 工作流配置; 类别 `infra`; 类型 `configuration`): 这是 CI 工作流的入口文件, 定义了 GT 生成任务的触发、输入参数和作业步骤, 变更直接影响用户交互和任务执行流程。
- `scripts/ci/utls/diffusion/publish_diffusion_gt.py` (模块 CI 脚本; 类别 `infra`; 类型 `core-logic`; 符号 `collect_images`, `publish`): 这是发布 GT 图像到远程仓库的核心脚本, 变更使其支持自定义目标目录, 确保与工作流参数联动。

关键符号: collect_images, publish

关键源码片段

scripts/ci/utils/diffusion/publish_diffusion_gt.py

这是发布 GT 图像到远程仓库的核心脚本，变更使其支持自定义目标目录，确保与 workflow 参数联动。

```
# 定义默认远程目标目录，支持通过参数覆盖
```

```
DEFAULT_TARGET_DIR = "diffusion-ci/consistency_gt"
```

```
def collect_images(source_dir, target_dir):
```

```
    """
```

```
    从source_dir收集图像文件，返回(repo_path, content)元组列表。
```

```
    target_dir: 远程仓库中的目标目录，用于构建repo_path。
```

```
    """
```

```
    files = []
```

```
    for entry in sorted(os.listdir(source_dir)):
```

```
        full_path = os.path.join(source_dir, entry)
```

```
        if not os.path.isfile(full_path):
```

```
            continue
```

```
        if not any(entry.lower().endswith(ext) for ext in IMAGE_EXTENSIONS):
```

```
            continue
```

```
        with open(full_path, "rb") as f:
```

```
            content = f.read()
```

```
        repo_path = f"{target_dir}/{entry}" # 使用传入的 target_dir 构建路径
```

```
        files.append((repo_path, content))
```

```
    return files
```

```
def publish(source_dir, target_dir=None):
```

```
    """
```

```
    发布图像到远程仓库。
```

```
    target_dir: 可选，指定远程目标目录；默认为DEFAULT_TARGET_DIR。
```

```
    """
```

```
    target_dir = target_dir or DEFAULT_TARGET_DIR # 处理默认值
```

```
    files_to_upload = collect_images(source_dir, target_dir)
```

```
    if not files_to_upload:
```

```
        print(f"No image files found in {source_dir}")
```

```
        return
```

```
    print(f"Found {len(files_to_upload)} image(s) to upload to {REPO_OWNER}/{REPO_NAME}/  
    {target_dir}")
```

```
    # 后续上传逻辑保持不变 ...
```

```
def main():
```

```
    parser = argparse.ArgumentParser()
```

```
    parser.add_argument("--source-dir", required=True, help="Directory containing GT images")
```

```
    parser.add_argument(  
        "--target-dir",  
        required=False,
```

```
default=None,  
help=f"Target directory in the remote repo (default: {DEFAULT_TARGET_DIR})", #  
    新增命令行参数  
)  
args = parser.parse_args()  
publish(args.source_dir, args.target_dir) # 传递 target_dir 参数
```

评论区精华

Review 讨论较少，仅有一次由 mickqian 的批准，无具体评论。这表明变更设计清晰，直接满足了需求，未引发技术争议或设计权衡讨论。

- 暂无高价值评论线程

风险与影响

- 风险：

1. 向后兼容性风险：默认行为保持不变，对现有用户无影响。但若用户之前依赖硬编码路径的脚本或文档，可能需要更新。
2. 参数传递错误风险：在 `.github/workflows/diffusion-ci-gt-gen.yml` 中，`output_name` 输入为空时，`--target-dir` 参数可能传递空字符串，但发布脚本的 `target_dir` or `DEFAULT_TARGET_DIR` 逻辑能正确处理，风险较低。
3. 路径安全风险：自定义的 `output_name` 可能包含特殊字符或路径遍历序列（如 `../`），但工作流在 GitHub Actions 环境中运行，且仅用于构建本地目录和远程路径，风险可控。
4. 发布脚本逻辑风险：`publish` 函数新增可选参数，但调用方（工作流）已适配，不会破坏现有调用。

- 影响：

1. 用户影响：为需要并行生成多组 GT 图像或组织测试集的用户提供了灵活性，不影响默认用户。
2. 系统影响：仅修改 CI 工作流和发布脚本，不影响核心推理引擎、模型或运行时性能。
3. 团队影响：简化了扩散模型测试的 CI 管理，支持更复杂的测试场景，如同时为不同模型版本生成 GT。 - 风险标记：路径参数化，向后兼容性

关联脉络

- PR #23411 [Docs] Improve SGLang Diffusion docs navigation and compatibility table: 同属 diffusion 模块，涉及扩散模型相关工具或文档的改进，可能共享上下文。
- PR #23413 Fix docker build error: 同属 infra 标签，涉及基础设施或 CI/CD 的修复或增强，体现团队对工具链的持续维护。
- PR #23404 [docs] update logo images for google, qwen, wan, and zimage: 同属 infra 标签，涉及 CI 或文档站点的资源管理，反映基础设施的细节优化趋势。