PR #22955 完整报告

sgl-project/sglang

[Diffusion] Fix ModelOpt B200 CI artifact coverage

合并时间：2026-04-17 23:33

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22955

执行摘要

- 一句话：修复 ModelOpt B200 diffusion CI 覆盖，优化权重文件选择和 artifact 保存。
- 推荐动作：建议工程团队精读 transformer_load_utils.py 中的 _prefer_mixed_safetensors_files 函数，理解其设计权衡：在遇到混合和非混合文件共存时，优先选择混合版本以避免重复张量名问题。同时，关注测试 artifact 保存机制，确保在 CI 中正确配置环境变量以利用此功能。

功能与动机

PR body 明确指出："Follow-up to #22772 for the B200 ModelOpt diffusion CI coverage." 目标是修复 B200 套件中 ModelOpt diffusion 测试的覆盖问题，具体包括：移除 BF16 质量比较测试和相关配置；保存生成 artifacts 以供验证；修复 FLUX.2 NVFP4 CI 启动路径，因为原始 NVFP4 仓库是原始 safetensors 导出，缺少 Diffusers config.json，导致加载失败。

实现拆解

1. 核心加载逻辑调整：在 python/sglang/multimodal_gen/runtime/loader/transformer_load_utils.py 中新增正则表达式 _MIXED_SAFETENSORS_RE 和函数 _prefer_mixed_safetensors_files，用于优先选择混合精度 safetensors 文件（如 foo-mixed.safetensors），避免加载重复张量名导致的验证错误。该函数在 resolve_transformer_safetensors_to_load 中被调用，置于现有过滤逻辑之前。
2. 测试 artifact 保存机制：在 python/sglang/multimodal_gen/test/server/test_server_common.py 中新增 _save_diffusion_artifact 方法，当环境变量 SGLANG_DIFFUSION_ARTIFACT_DIR 设置且测试用例 ID 包含 "modelopt" 时，保存生成的图像或视频文件（视频文件添加 _5s.mp4 后缀），并在 test_diffusion_generation 方法中调用。
3. 测试配置更新：在 python/sglang/multimodal_gen/test/server/testcase_configs.py 中，为 MODELOPT_T2I_CI_sampling_params 和 MODELOPT_T2V_CI_sampling_params 添加固定种子（"seed": 0）和视频时长参数（seconds=5），确保测试确定性；将常量 MODELOPT_FLUX2_NVFP4_MODEL 重命名为 MODELOPT_FLUX2_NVFP4_WEIGHTS 以澄清用途。在 python/sglang/multimodal_gen/test/server/gpu_cases.py 中，更新 flux2_modelopt_nvfp4_t2i 测试用例，使用基础模型路径并添加 --transformer-weights-path 参数指向量化权重。

4. 单元测试覆盖：在 python/sglang/multimodal_gen/test/unit/test_transformer_quant.py 中新增 test_resolve_transformer_safetensors_to_load_prefers_mixed_export 测试，验证混合文件优先逻辑。
5. CI 工作流增强：在 .github/workflows/pr-test-multimodal-gen.yml 中添加环境变量 SGLANG_DIFFUSION_ARTIFACT_DIR 和 Upload diffusion artifacts 步骤，将保存的 artifacts 上传供后续分析。

关键文件：

- python/sglang/multimodal_gen/runtime/loader/transformer_load_utils.py（模块 加载器；类别 source；类型 core-logic；符号 _prefer_mixed_safetensors_files）：核心加载逻辑文件，新增混合精度 safetensors 文件优先逻辑，解决 ModelOpt NVFP4 仓库中重复张量名导致的加载失败问题。
- python/sglang/multimodal_gen/test/server/test_server_common.py（模块 测试框架；类别 test；类型 test-coverage；符号 _save_diffusion_artifact）：主要测试文件，新增 artifact 保存功能，确保 CI 能保存 ModelOpt 测试生成的媒体输出供上传和分析。
- python/sglang/multimodal_gen/test/unit/test_transformer_quant.py（模块 单元测试；类别 test；类型 test-coverage；符号 test_resolve_transformer_safetensors_to_load_prefers_mixed_export）：单元测试文件，新增测试用例验证混合文件优先逻辑，确保加载器行为符合预期。
- python/sglang/multimodal_gen/test/server/testcase_configs.py（模块 测试配置；类别 test；类型 configuration）：测试配置文件，更新采样参数以添加固定种子和视频时长，确保测试确定性和一致性。
- python/sglang/multimodal_gen/test/server/gpu_cases.py（模块 测试用例；类别 test；类型 test-coverage）：GPU 测试用例文件，更新 FLUX.2 NVFP4 测试用例以使用正确的模型路径和量化权重参数，修复加载失败问题。
- .github/workflows/pr-test-multimodal-gen.yml（模块 CI 管道；类别 infra；类型 infrastructure）：CI 工作流文件，新增 artifact 保存和环境变量设置，支持上传生成的 diffusion artifacts 供后续分析。

关键符号：_prefer_mixed_safetensors_files, _save_diffusion_artifact, test_resolve_transformer_safetensors_to_load_prefers_mixed_export

关键源码片段

python/sglang/multimodal_gen/runtime/loader/transformer_load_utils.py

核心加载逻辑文件，新增混合精度 safetensors 文件优先逻辑，解决 ModelOpt NVFP4 仓库中重复张量名导致的加载失败问题。

```
import re
import os

# 新增正则表达式，匹配以 "-mixed" 结尾的 safetensors 文件名（可能包含分片后缀）
_MIXED_SAFETENSORS_RE = re.compile(r".*-mixed(?:-\d+-of-\d+)?\.safetensors$")

def _prefer_mixed_safetensors_files(safetensors_list: list[str]) -> list[str]:
```

```
    """
    优先选择混合精度 transformer 导出文件，避免加载重复张量名。
    一些原始 ModelOpt NVFP4 仓库同时包含 `foo-mixed.safetensors` 和 `foo.safetensors`，
    它们是完整的替代导出文件而非分片，同时加载会触发重复张量名验证错误。
    """
    # 过滤出所有混合文件
    mixed_files = [
        path
        for path in safetensors_list
        if _MIXED_SAFETENSORS_RE.match(os.path.basename(path))
    ]
    # 如果没有混合文件或所有文件都是混合文件，则返回原列表
    if not mixed_files or len(mixed_files) == len(safetensors_list):
        return safetensors_list

    # 记录选择混合文件并忽略非混合兄弟文件
    logger.info(
        "Using %d mixed transformer safetensors file(s) and ignoring %d sibling "
        "non-mixed file(s): %s",
        len(mixed_files),
        len(safetensors_list) - len(mixed_files),
        mixed_files,
    )
    return mixed_files

# 在 resolve_transformer_safetensors_to_load 函数中调用此函数
# 注意：此调用被添加在 _filter_duplicate_precision_variant_safetensors 之前
```

评论区精华

review 评论中，gemini-code-assist[bot] 指出两个关键问题：

- 种子缺失：MODELOPT_T2I_CI_sampling_params 和 MODELOPT_T2V_CI_sampling_params 缺少固定种子，可能导致质量检查因非确定性输出而失败。PR 通过添加 "seed": 0 解决了此问题。
- 配置路径矛盾：在 flux_2_nvfp4.py 中，逻辑强制使用基础模型路径进行配置，与文档中关于从覆盖仓库读取配置的说明可能冲突。但此问题未在本 PR 中直接解决，评论指出需考虑添加注释或更新文档。最终，PR 专注于修复 CI 启动路径，通过 --transformer-weights-path 参数正确加载量化权重。
  - 测试采样参数缺少固定种子 (correctness): PR 通过在这些参数中添加 "seed": 0 解决了此问题，确保测试确定性。
  - 配置路径与文档矛盾 (design): 此问题未在本 PR 中直接解决，但 PR 通过使用 --transformer-weights-path 参数正确加载量化权重，避免了配置加载失败。

风险与影响

- 风险：技术风险包括：

- 回归风险：_prefer_mixed_safetensors_files 函数可能错误过滤掉必要的非混合文件，影响其他模型的加载，特别是在包含多种 safetensors 文件的仓库中。
- 性能影响：artifact 保存机制会增加磁盘 I/O 和存储开销，尤其在频繁运行 CI 时可能积累大量文件。
- 兼容性问题：测试配置中固定种子和参数变更（如视频时长）可能与其他测试用例或环境变量设置冲突，导致不一致行为。
- 安全风险：无显著安全风险，但 artifact 保存可能暴露生成内容，需确保敏感数据不被意外上传。
- 影响：影响范围和程度：
- 用户影响：对最终用户无直接影响，主要面向开发者和 CI 维护者。
- 系统影响：CI 流程更可靠，ModelOpt diffusion 测试能正确执行并保存输出 artifacts，便于调试和验证。加载逻辑变更仅影响使用混合精度 safetensors 文件的量化模型，对其他模型无影响。
- 团队影响：简化了测试维护，通过移除 BF16 质量检查减少 CI 复杂度；artifact 保存提供了可视化验证手段，提升问题诊断效率。
- 风险标记：核心路径变更，测试依赖环境变量，配置兼容性风险

关联脉络

- PR #22772 [未知，PR body 提及]: PR body 明确说明本 PR 是 #22772 的后续，专注于 B200 ModelOpt diffusion CI coverage，表明两者在功能演进上连续。
- PR #23052 [diffusion] doc: update doc: 历史 PR 中涉及 diffusion 文档更新，与本 PR 中的文档变更（如更新量化文档）相关，共同完善 diffusion 模块的文档覆盖。
- PR #23028 [codex] Update diffusion skills: 另一个 diffusion 相关 PR，更新 benchmark/profile 技能，与本 PR 的 CI 和测试改进有协同作用，反映 diffusion 模块的持续优化。