PR #23624 完整报告

sgl-project/sglang

[diffusion] fix: unify LTX-2.3 HQ codepath gates for all LTX-2.3 variants

合并时间：2026-04-24 17:44

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23624

执行摘要

- 一句话：统一 LTX-2.3 高质量代码路径，修复语义差异
- 推荐动作：建议仔细审视非 HQ LTX-2.3 输出变化是否符合预期，并优先合并后续的一致性 GT 更新 PR。开发者可关注 is_ltx23_native_variant 辅助函数的统一使用。

功能与动机

PR #23366 为 LTX-2.3 HQ 管道添加了专用代码路径，但审计发现 HQ 分支（确定性重噪生成器、fp32 混合、sigma 扩展）对于所有 LTX-2.3 变体都是语义上正确的形式，而非 HQ 分支仅保留了回归前的基线行为。本 PR 统一了这些路径，确保所有 LTX-2.3 变体使用官方对齐的实现。

实现拆解

实现拆解分为三个部分：

1. ltx_2_pipeline.py: LTX2SigmaPreparationStage.forward 中，对 LTX-2.3 变体始终使用分辨率感知的 token 数（latent_frames × H × W）计算 sigma 偏移，而非仅限于 HQ 管道；
2. denoising_av.py: LTX2RefinementStage.forward 中，将 HQ 专用条件 is_hq_pipeline 替换为 is_ltx23，使所有 LTX-2.3 变体使用确定性重噪生成器和 fp32 混合；
3. 扩展 sigma 调度最后一步为 [..., 0.0011, 0.0] 的条件也从 HQ 扩展到所有 LTX-2.3 变体，匹配官方 res2s 循环。同时更新了性能基线文件以反映新的执行时间。

关键文件：

- python/sglang/multimodal_gen/runtime/pipelines_core/stages/denoising_av.py（模块 扩散流水线；类别 source；类型 core-logic；符号 LTX2RefinementStage.forward）：核心逻辑变更：将 HQ 条件从 pipeline_class_name 改为 is_ltx23_native_variant，统一所有 LTX-2.3 变体的噪声生成和 fp32 混合行为。
- python/sglang/multimodal_gen/runtime/pipelines/ltx_2_pipeline.py（模块 扩散流水线；类别 source；类型 core-logic；符号 LTX2SigmaPreparationStage.forward）：Sigma 偏移计算条件统一：对所有 LTX-2.3 变体使用分辨率感知的 token 数，而非仅限 HQ 管道。
- python/sglang/multimodal_gen/test/server/perf_baselines.json（模块 性能基线；类别 test；类型 test-coverage）：更新 HQ 管道的性能基线数据，反映统一后路径的执行时间变

化。

关键符号：LTX2RefinementStage.forward, LTX2SigmaPreparationStage.forward, _build_stage2_renoise_generator, is_ltx23_native_variant

关键源码片段

python/sglang/multimodal_gen/runtime/pipelines_core/stages/denoising_av.py

核心逻辑变更：将 HQ 条件从 pipeline_class_name 改为 is_ltx23_native_variant，统一所有 LTX-2.3 变体的噪声生成和 fp32 混合行为。

```
# denoising_av.py - LTX2RefinementStage.forward 核心变更
# 原条件：is_hq_pipeline = server_args.pipeline_class_name == "LTX2TwoStageHQPipeline"
# 新条件：使用 is_ltx23_native_variant 判断，适用于所有 LTX-2.3 变体
is_ltx23 = is_ltx23_native_variant(server_args.pipeline_config.vae_config.arch_config)
if is_ltx23:
    # 确定性重噪生成器：匹配官方 LTX-2.3 行为
    renoise_generator = self._build_stage2_renoise_generator(batch, video_reference_for_gen)
else:
    renoise_generator = None
# 后续所有 is_hq_pipeline 条件均替换为 is_ltx23
```

python/sglang/multimodal_gen/runtime/pipelines/ltx_2_pipeline.py

Sigma 偏移计算条件统一：对所有 LTX-2.3 变体使用分辨率感知的 token 数，而非仅限 HQ 管道。

```
# ltx_2_pipeline.py - LTX2SigmaPreparationStage.forward 关键变更
if is_ltx23_native_variant(server_args.pipeline_config.vae_config.arch_config):
    # 所有 LTX-2.3 变体均使用分辨率感知的 sigma 偏移
    # 但保留 pipeline_class_name 判断以区分 HQ（半分辨率）和非 HQ（默认 token 数）
    if server_args.pipeline_class_name == "LTX2TwoStageHQPipeline":
        # 计算半分辨率 latent 的 tokens
        latent_num_frames = (int(batch.num_frames) - 1) // int(server_args.pipeline_config.vae_
        temporal_compression) + 1
        latent_height = int(batch.height) // int(server_args.pipeline_config.vae_scale_factor)
        latent_width = int(batch.width) // int(server_args.pipeline_config.vae_scale_factor)
        batch.sigmas = build_official_ltx2_sigmas(
            int(batch.num_inference_steps),
            number_of_tokens=latent_num_frames * latent_height * latent_width,
        )
    else:
        # 非 HQ 使用默认 token 数（4096）
        batch.sigmas = build_official_ltx2_sigmas(int(batch.num_inference_steps))
else:
    # 非 LTX-2.3 使用线性 sigma 调度
    batch.sigmas = np.linspace(1.0, 1.0 / int(batch.num_inference_steps), int(batch.num_
    inference_steps)).tolist()
```

评论区精华

由于 PR 由作者自行合并，且 review 评论数为 0，因此无实质性设计讨论。提交历史显示作者进行了多轮修正（G1、G3），说明实现细节经过仔细交叉验证。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险在于：1）非 HQ LTX-2.3 变体的输出语义会改变，可能导致已部署管道的生成结果不一致；2）性能基线文件中的预期时间变化较大（如 AVDenoisingStage 从 20227ms 变为 25873ms），可能引发 CI 回归告警；3）需要后续更新一致性 GT 基准，否则一致性测试可能失败。风险范围局限于扩散模块的 LTX-2.3 系列。
- 影响：对用户：使用 LTX-2.3 非 HQ 管道的用户将获得更准确的生成结果，但输出会变化；对系统：CI 中的性能基线已更新，但一致性测试需等待后续 PR 更新 GT；对团队：工程清理了历史遗留条件分支，降低了维护成本。
- 风险标记：核心路径统一可能改变非 HQ 输出，一致性 GT 基准需后续更新，CI 性能基线大幅变化

关联脉络

- PR #23366 [diffusion] model: support LTX2.3 high quality pipeline: 本 PR 是对 #23366 引入的 HQ 代码路径的统一审计和修复