PR #25411 完整报告

sgl-project/sglang

[diffusion] Default Qwen Image VAE precision to bf16

合并时间：2026-05-16 21:55

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/25411

执行摘要

- 一句话：扩散模型 VAE 默认精度改为 bf16
- 推荐动作：该 PR 质量良好，数据充分，值得合并。建议精读 MOVA 的 AMD 编译错误修复，理解 AMD 平台上的兼容性限制，并在未来引入类似精度优化时注意测试 AMD CI。

功能与动机

PR body 指出 Qwen-Image checkpoint 的 VAE 权重本身就是 bf16，但 pipeline 默认设为 fp32 导致不必要的精度提升，浪费显存。先前用户需手动设置 --vae-precision bf16，不够直观。作者测量峰值显存从 64098 MB 降至 58518 MB（降 8.71%），且推理速度几乎不变。

实现拆解

1. 修改 Qwen-Image pipeline 配置(qwen_image.py)：在 QwenImagePipelineConfig 中添加 vae_precision: str = "bf16"（覆盖默认的 fp32）。
2. 修改 LTX-2 pipeline 配置(ltx_2.py)：在 LTX2PipelineConfig 中添加 vae_precision: str = "bf16"，并将 audio_vae_precision 从 "fp32" 改为 "bf16"。
3. 修改 MOVA pipeline 配置(mova.py)：在 MOVAPipelineConfig 中添加 vae_precision: str = "bf16"，并将 audio_vae_precision 从 "fp32" 改为 "bf16"。
4. 修改 FLUX pipeline 配置(flux.py)：在 FluxPipelineConfig 中添加 vae_precision: str = "bf16"。
5. 修改 Z-Image pipeline 配置(zimage.py)：在 ZImagePipelineConfig 中添加 vae_precision: str = "bf16"。
6. 更新测试数据版本(test_utils.py)：调整 CI 测试用 ground truth 数据版本号（GIT commit hash）以匹配新精度下的输出。
7. 修复 AMD CI 编译错误：MOVA 的 audio_vae_precision: "fp32" -> "bf16" 行在 AMD 上触发了 Snake1d.forward 中的 uint32_t 类型未定义错误（与 HIP 后端兼容性有关），通过将所有 audio_vae_precision 改为 bf16 间接规避。

关键文件：

- python/sglang/multimodal_gen/configs/pipeline_configs/qwen_image.py（模块 扩散管道；类别 source；类型 core-logic）：核心变更起点：为 Qwen-Image pipeline 添加默认 bf16 VAE 精度，触发全 PR 的讨论和扩展。

- python/sglang/multimodal_gen/configs/pipeline_configs/ltx_2.py（模块 扩散管道；类别 source；类型 core-logic）：LTX-2 的 VAE 和 audio VAE 精度均改为 bf16，是扩展变更的一部分。
- python/sglang/multimodal_gen/configs/pipeline_configs/mova.py（模块 扩散管道；类别 source；类型 core-logic）：MOVA 的 VAE 和 audio VAE 精度改为 bf16，同时修复了 AMD CI 上的 HIP 编译问题。
- python/sglang/multimodal_gen/configs/pipeline_configs/flux.py（模块 扩散管道；类别 source；类型 core-logic）：FLUX pipeline 添加默认 bf16 VAE 精度，是扩展变更的一部分。
- python/sglang/multimodal_gen/configs/pipeline_configs/zimage.py（模块 扩散管道；类别 source；类型 core-logic）：Z-Image pipeline 添加默认 bf16 VAE 精度，是扩展变更的一部分。
- python/sglang/multimodal_gen/test/test_utils.py（模块 测试；类别 test；类型 test-coverage）：更新 CI 测试数据版本 hash，确保精度变更后的测试通过。

关键符号：未识别

关键源码片段

python/sglang/multimodal_gen/configs/pipeline_configs/qwen_image.py

核心变更起点：为 Qwen-Image pipeline 添加默认 bf16 VAE 精度，触发全 PR 的讨论和扩展。

```
# python/sglang/multimodal_gen/configs/pipeline_configs/qwen_image.py
# 在 QwenImagePipelineConfig 类的 vae_sp 之后插入一行，覆盖默认 fp32
@dataclass
class QwenImagePipelineConfig(QwenImageRolloutPipelineMixin, ImagePipelineConfig):
    ...
    vae_sp: bool = False
    vae_precision: str = "bf16" # 新加：默认使用 bf16，节省显存且 checkpoint 本身就是 bf16
    dit_config: DiTConfig = field(default_factory=QwenImageDitConfig)
    ...
```

python/sglang/multimodal_gen/configs/pipeline_configs/ltx_2.py

LTX-2 的 VAE 和 audio VAE 精度均改为 bf16，是扩展变更的一部分。

```
# python/sglang/multimodal_gen/configs/pipeline_configs/ltx_2.py
# 在 LTX2PipelineConfig 中同时修改视频和音频 VAE 精度
@dataclass
class LTX2PipelineConfig(PipelineConfig):
    ...
    vae_config: LTXVideoVAEConfig = field(default_factory=LTXVideoVAEConfig)
    vae_precision: str = "bf16" # 新加：视频 VAE 默认 bf16
    audio_vae_config: LTXAudioVAEConfig = field(default_factory=LTXAudioVAEConfig)
    audio_vae_precision: str = "bf16" # 原为 "fp32"，改为 bf16
    ...
```

python/sglang/multimodal_gen/configs/pipeline_configs/mova.py

MOVA 的 VAE 和 audio VAE 精度改为 bf16，同时修复了 AMD CI 上的 HIP 编译问题。

```
# python/sglang/multimodal_gen/configs/pipeline_configs/mova.py
# 在 MOVAPipelineConfig 中同时修改视频和音频 VAE 精度
@dataclass
class MOVAPipelineConfig(PipelineConfig):
    ...
    vae_config: WanVAEConfig = field(default_factory=WanVAEConfig)
    vae_precision: str = "bf16" # 新加：视频 VAE 默认 bf16
    audio_vae_config: DacVAEConfig = field(default_factory=DacVAEConfig)
    audio_vae_precision: str = "bf16" # 原为 "fp32"，改为 bf16；AMD CI 避免编译错误
    ...
```

评论区精华

Review 中主要讨论了是否扩大变更范围：作者 qimcis 发现 Hunyuan3D、LTX-2、MOVA 也存在类似的上转换问题，提议一并修改；维护者 mickqian 要求检查其他 VAE。作者随后在 1xH100 上测试了 7 个模型的 fp32 与 bf16 对比，显存节省 4%-8%，视觉质量可接受，并将修改扩展到 LTX-2、MOVA、FLUX、Z-Image。AMD CI 失败由 bingxche 指出，根因是 MOVA 的 audio_vae_precision: "fp32" -> "bf16" 行触发 HIP 编译问题，最终通过将所有音频 VAE 精度改为 bf16 解决。

- 是否将修改扩展到其他模型 (design): 作者在 1xH100 上测试了 7 个模型，显存节省 4%-8%，视觉质量可接受，将修改扩展到 LTX-2、MOVA、FLUX、Z-Image。
- AMD CI 编译错误 (correctness): 通过将所有音频 VAE 精度改为 bf16 解决，该行被修改后编译通过。

风险与影响

- 风险：
  1. 回归风险（低）：仅修改默认精度，未改动核心推理逻辑；作者提供了前后视觉对比图和性能数据，无明显退化。但由于修改覆盖 5 个模型，个别模型在特定硬件（如 AMD）上可能因 bf16 支持不完善产生精度或性能问题（AMD CI 已修复）。
  2. 兼容性风险（低）：如果用户依赖 fp32 的默认行为并手动设置 --vae-precision fp32，本次变更不影响显式配置。
  3. 编译风险（已解决）：AMD 上 MOVA 的 HIP 编译问题已解决。
- 影响：
  - 用户影响：使用这些扩散模型的用户将自动获得更低的显存占用（-4%~-9%）和几乎相同的推理速度与质量，无需额外配置。
  - 系统影响：减少 GPU 显存压力，有利于大 batch 或长序列场景。
  - 团队影响：为后续其他模型默认精度优化提供了参考模式（只需修改 pipeline config 中的字段）。同时也揭示了 AMD 平台上 Snake1d.forward 与 HIP 后端的兼容性隐患。
  - 风险标记：AMD 平台编译兼容性，仅单位测试，缺少集成测试覆盖

关联脉络

- PR #25510 [diffusion] tighten selected perf baselines: 同为 diffusion 领域的性能优化 PR，修改了性能基准数据和配置，与本 PR 的精度调整可能关联。
- PR #25517 [diffusion] feat: configure encoder as layerwise-offload by default: 同为 diffusion 领域的默认配置优化，将编码器默认改为 layerwise offload 以节省显存，与本 PR 目标一致。