PR #40830 完整报告

vllm-project/vllm

[MM][CG] Support ViT CG for Qwen2.5-VL

合并时间：2026-05-02 11:10

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/40830

执行摘要

- 一句话：为 Qwen2.5-VL 启用 ViT CUDA 图支持
- 推荐动作：该 PR 设计清晰，遵循了已批准的 Qwen3-VL 实现模式。核心决策是将元数据计算与模型前向分离，实现了跳帧和图模式的代码复用。值得关注的是文中对视频修剪与 CUDA 图交互的处理策略（直接禁用），以及通过参数覆盖保证图形状安全的方法。建议阅读 *prepare_encoder_metadata* 和 *get_encoder_cudagraph_config* 的实现。

功能与动机

减少 Qwen2.5-VL 视觉编码器的执行延迟，特别是对于编码器密集的前缀填充（prefill）场景。PR 描述中引用了之前 Qwen3-VL 的 PR #35963 作为前例，并通过基准测试（300 个请求，20 个图像 / 请求）展示了 CUDA 图启用后 TTFT 从 37869.81ms 降至 36690.01ms，TPOT 从 704.43ms 降至 688.73ms 等改进。

实现拆解

1. 提取编码器元数据方法（qwen2_5_vl.py）：将原先在 forward 方法中通过 grid_thw 计算旋转变换、窗口索引、累积序列长度等逻辑抽取为新的 *prepare_encoder_metadata* 方法。该方法接受 max_batch_size、max_frames_per_batch、max_window_seqs_per_batch、max_seqlen_override 等可选参数，使得 CUDA 图捕获时能够覆盖最坏情况下的形状，而回放时使用实际输入但保持张量形状一致。
2. 实现 SupportsEncoderCudaGraph 协议：新增 *get_encoder_cudagraph_config*（返回 EncoderCudaGraphConfig 配置，包含 budget 范围、最大视觉项数等）、*get_input_modality*（返回支持的模态列表，视频仅在非修剪模式且后端为 FlashAttn 时启用）、*get_max_frames_per_video*（通过 EVS 配置或采样帧数推断）、*get_encoder_cudagraph_budget_range*（基于图像 / 视频 token 长度范围）等方法。同时添加了辅助方法 *_get_pixel_values_by_modality* 和 *_get_grid_thw_by_modality* 用于按模态提取输入。
3. 修改模型前向逻辑：重写 forward 方法使其调用 *prepare_encoder_metadata*，然后根据是否处于 CUDA 图捕获 / 回放状态选择不同的执行路径（跳帧模式直接调用注意力，图模式则利用预先捕获的图）。在 *embed_multimodal* 中通过新增的模态识别逻辑循环处理图像和视频。
4. 更新测试和示例：在 test_qwen2_5_vl.py 中添加了窗口注意力图像回归测试 *test_qwen2_5_vl_window_attention_image* 和批量图像测试

test_qwen2_5_vl_window_attention_image_batch，它们使用 _encoder_cudagraph_config 辅助函数配置 CUDA 图。同时，在已有的 test_vit_cudagraph.py 中为 Qwen2.5-VL 增加了配置条目，使其能够复用 ViT CUDA 图测试。示例文件 vision_language_offline.py 中将 qwen2_5_vl 加入 MODELS_SUPPORT_VIT_CUDA_GRAPH 列表。

5. 更新文档：在 cuda_graphs_multimodal.md 中添加 Qwen2.5-VL 的条目，并注明仅测试了 FlashAttention 2 和 3 后端。

关键文件：

- vllm/model_executor/models/qwen2_5_vl.py（模块 模型层；类别 source；类型 core-logic；符号 forward, prepare_encoder_metadata, get_encoder_cudagraph_config, get_input_modality）：核心模型文件，提取了 prepare_encoder_metadata 方法并实现了 SupportsEncoderCudaGraph 协议的所有必要方法，是这次变更的主要载体。
- tests/models/multimodal/generation/test_qwen2_5_vl.py（模块 测试；类别 test；类型 test-coverage；符号 _window_attention_regression_image, _encoder_cudagraph_config, test_qwen2_5_vl_window_attention_image, test_qwen2_5_vl_window_attention_image_batch）：新增了窗口注意力图像回归测试和批量图像测试，验证 CUDA 图与跳帧的一致性。
- tests/models/multimodal/generation/test_vit_cudagraph.py（模块 测试；类别 test；类型 test-coverage）：添加了 Qwen2.5-VL 的配置项，使其能复用已有的 ViT CUDA 图测试框架。
- examples/generate/multimodal/vision_language_offline.py（模块 示例；类别 source；类型 configuration）：将 Qwen2.5-VL 加入支持 ViT CUDA 图的模型列表，使示例能触发该功能。
- docs/design/cuda_graphs_multimodal.md（模块 文档；类别 docs；类型 documentation）：更新文档以反映 Qwen2.5-VL 支持 CUDA 图及其测试后端限制。

关键符号：prepare_encoder_metadata, get_encoder_cudagraph_config, get_input_modality, get_max_frames_per_video, get_encoder_cudagraph_budget_range, _get_pixel_values_by_modality, _get_grid_thw_by_modality, _window_attention_regression_image, _encoder_cudagraph_config, test_qwen2_5_vl_window_attention_image, test_qwen2_5_vl_window_attention_image_batch

关键源码片段

tests/models/multimodal/generation/test_qwen2_5_vl.py

新增了窗口注意力图像回归测试和批量图像测试，验证 CUDA 图与跳帧的一致性。

```
# test_qwen2_5_vl.py — 新增的窗口注意力图像测试

IMAGE_PLACEHOLDER = "<|vision_start|><|image_pad|><|vision_end|>"

# 回归问题图片（issue #15122）
```

```
def _window_attention_regression_image():
    image = ImageAsset("hato").pil_image
    return image.resize((image.width // 2, image.height // 2))


# 辅助函数：生成 CUDA 图编译配置
def _encoder_cudagraph_config(*, max_vision_items: int) -> dict:
    return {
        "cudagraph_mm_encoder": True,
        "encoder_cudagraph_max_vision_items_per_batch": max_vision_items,
    }


@pytest.mark.core_model
@pytest.mark.parametrize("model", models)
@pytest.mark.parametrize("dtype", [target_dtype])
@pytest.mark.parametrize("max_tokens", [128])
@pytest.mark.parametrize("use_bytecode_hook", [True, False])
def test_qwen2_5_vl_window_attention_image(
    vllm_runner,
    model,
    dtype: str,
    max_tokens: int,
    use_bytecode_hook: bool,
    monkeypatch,
) -> None:
    """
    窗口注意力图像的回归测试，同时验证 CUDA 图模式下的正确性。
    使用 _encoder_cudagraph_config 启用编码器 CUDA 图。
    """
    monkeypatch.setenv("VLLM_USE_BYTECODE_HOOK", "1" if use_bytecode_hook else "0")

    prompt = [WINDOW_ATTN_IMAGE_PROMPT]
    images = [[_window_attention_regression_image()]]

    with vllm_runner(
        model,
        runner="generate",
        max_model_len=4096,
        dtype=dtype,
        limit_mm_per_prompt={"image": 1},
        # 传入 CUDA 图配置
        compilation_config=_encoder_cudagraph_config(max_vision_items=1),
    ) as vllm_model:
        outputs = vllm_model.generate_greedy(prompt, max_tokens, images=images)

        assert len(outputs) == 1
        output_ids, output_text = outputs[0]
        assert len(output_ids) > 0
```

```
assert len(output_text) > 0
assert isinstance(output_text, str)
```

评论区精华

- 张量设备位置（gemini-code-assist[bot] → johncalesp）：审查者指出 max_seqlen_full 和 max_seqlen_window 可能在 CPU 上创建，需要移至目标设备以避免图回放时的同步问题。作者回应已确保张量在同一设备上。
- 测试复用 vs 新增（shen-shanshan → johncalesp）：建议直接使用 test_vit_cudagraph.py 中的通用测试，只需添加新模型配置。作者同意并移除了专门的匹配测试，改用配置方式。
- 修剪率下的 CUDA 图行为（b-mu → shen-shanshan, johncalesp）：提出当 video_pruning_rate>0 时，后处理仅在跳帧路径运行而不在图路径，这是 Qwen2.5-VL 和 Qwen3-VL 共同的问题。作者通过禁用修剪模式下的 CUDA 图来规避。
- 代码重复（shen-shanshan → johncalesp）：指出 get_encoder_cudagraph_config 等代码与 Qwen3-VL 几乎相同，提议未来消除冗余。作者认为目前仅涉及三个模型，暂时可接受。
- max_seqlen_window 覆盖缺失（b-mu → johncalesp）：审查者询问为何只覆盖 max_seqlen_full 而不覆盖 max_seqlen_window。作者最初以为窗口序列长度受模型几何结构限制，后补充了显式上界。
- 张量设备位置 (correctness): 作者确认已在同一设备上。
- 测试复用 vs 新增专用测试 (testing): 作者同意并移除专用匹配测试，改用配置方式。
- 修剪率下的 CUDA 图行为 (correctness): 作者通过禁用修剪模式下的 CUDA 图来规避。
- 代码冗余 (design): 作者认为仅三个模型，暂时可接受，未来可考虑抽象。
- max_seqlen_window 覆盖缺失 (correctness): 作者最初以为受几何结构限制，后补充了显式上界。

风险与影响

- 风险：
  1. CUDA 图兼容性：如果模型使用的注意力后端不是 FlashAttn 2/3，视频模态的 CUDA 图可能无法正常工作。当前配置仅在 FlashAttn 后端启用视频 CG，但未来新增后端时需要同步修改。
  2. 修剪模式退化：当启用视频修剪（EVS）时，CUDA 图被显式禁用，这可能导致修剪模式下性能较低（回归到跳帧路径）。
  3. 动态形状边缘情况：prepare_encoder_metadata 中的填充逻辑依赖 max_budget_range 等配置，若配置不当（如 budget 小于实际需要）可能导致形状不匹配或图重放失败。
  4. 与 Qwen3-VL 的代码重复：可能导致未来修复或增强时遗漏一处更新。但目前处于可控范围。
  5. 测试覆盖不足：缺少对视频模态与 CUDA 图并用的集成测试（仅在图像测试中覆盖），且缺少对预算整除性等不变量的显式断言，尽管讨论中提及应添加。 - 影响：用户影响：Qwen2.5-VL 用户在启用 --cudagraph_mm_encoder 并正确配置 budget 后，可以在编

码器密集负载下获得小幅到中等的延迟改善（TTFT 降低约 3%，TPOT 降低约 2%）。视频模态仅在特定后端下受益。系统影响：新增的 prepare_encoder_metadata 方法是框架中 CUDA 图协议的一部分，不改变其他模型。示例和文档均更新，降低了使用门槛。团队影响：为后续更多模型（如 Qwen2-VL）的 ViT CUDA 图支持提供了可复用的模式。

- 风险标记：核心路径变更，CUDA 图兼容性，缺少视频图集成测试，配置不当可能导致图回放失败

关联脉络

- PR #35963 [MM][CG] Support ViT CG for Qwen3-VL: 该 PR 是 Qwen2.5-VL 实现的前例和参考基础，实现模式几乎相同。
- PR #40580 [MM][CG] Fix max_budget auto-infer for Qwen3-VL: 讨论中 shen-shanshan 引用了该 PR，指出 max_budget 自动推理存在的问题，需要同步修复。