

PR #48594 完整报告

vllm-project/vllm

[Model] Enable LoRA support for tower and connector in LlavaNextVideo

合并时间: 2026-07-14 20:09

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/48594>

执行摘要

- 一句话: LlavaNextVideo 新增 LoRA 支持 (含视觉塔与连接器)
- 推荐动作: 该 PR 值得精读, 尤其是 `get_num_mm_encoder_tokens` 和 `get_num_mm_connector_tokens` 的实现方式——通过存储网格长度来反推各阶段 token 数量, 为其他具有空间池化的多模态模型 (如 LLaVA-NeXT、LLaVA-Onevision) 的 LoRA 支持提供了可复用的设计模式。

功能与动机

实现 Issue #31479 中关于更多多模态模型支持视觉塔和连接器 LoRA 的需求。

LlavaNextVideo 之前完全没有 LoRA 支持, 即使语言模型部分也不支持。需要实现

`get_num_mm_encoder_tokens` 和 `get_num_mm_connector_tokens` 来桥接 vision tower 与 connector 之间的 token 计数差异 (空间池化导致)。

实现拆解

1. 添加 LoRA 接口混入和模块映射: 在 `LlavaNextVideoForConditionalGeneration` 类继承中加入 `SupportsLoRA`, 并定义 `packed_modules_mapping` (`qkv_proj`、`gate_up_proj`), 使得 LoRA 框架能够识别模型并正确绑定目标模块。
2. 初始化时计算网格长度: 在 `__init__` 中通过 `get_vision_encoder_info` 获取视觉编码器的 `patch_grid_length`, 并根据 `spatial_pool_stride` 向上取整得到 `pooled_grid_length`, 作为实例属性存储。这是后续 token 计数反转的基础。
3. 实现 `get_mm_mapping`: 返回 `MultiModelKeys` 对象, 分别指定 `language_model`、`connector` (`multi_modal_projector`) 和 `tower_model` (`vision_tower`) 的模块前缀, 供 LoRA 框架定位子模块。
4. 实现 token 计数 helper: `get_num_mm_encoder_tokens` 将 LLM 侧的占位 token 数 (`pooled_grid_length2 per frame`) 反算为 vision tower 实际输出的 token 数 (`patch_grid_length2 per frame`); `get_num_mm_connector_tokens` 进行逆操作。注意两者均以帧为单位, 并处理非正整数边界。
5. 新增单元测试: `tests/models/multimodal/processing/test_llava_next_video.py` 使用 `_StubModel` 模拟属性, 测试真实配置 (LLaVA-NeXT-Video-7B-hf) 下的双向转换正确性, 以及非 2 的幂池化比率和零 token 边界情况。
6. 更新文档: `docs/models/supported_models.md` 中该模型的 LoRA 支持列标记为 .

关键文件：

- vllm/model_executor/models/llava_next_video.py（模块 模型定义；类别 source；类型 data-contract；符号 LlavaNextVideoForConditionalGeneration, get_mm_mapping, get_num_mm_encoder_tokens, get_num_mm_connector_tokens）：核心实现文件，添加 SupportsLoRA、packed_modules_mapping、get_mm_mapping、get_num_mm_encoder_tokens、get_num_mm_connector_tokens。
- tests/models/multimodal/processing/test_llava_next_video.py（模块 测试；类别 test；类型 test-coverage；符号 _StubModel, test_num_mm_tokens_match_real_config, test_num_mm_tokens_roundtrip, test_num_mm_tokens_zero）：新增测试文件，验证 token 计数 helper 的正确性，使用 _StubModel 模拟属性避免构造全模型。
- docs/models/supported_models.md（模块 文档；类别 docs；类型 documentation）：更新支持模型表格，标记该模型已支持 LoRA。

关键符号：init, get_mm_mapping, get_num_mm_encoder_tokens, get_num_mm_connector_tokens

关键源码片段

vllm/model_executor/models/llava_next_video.py

核心实现文件，添加 SupportsLoRA、packed_modules_mapping、get_mm_mapping、get_num_mm_encoder_tokens、get_num_mm_connector_tokens。

```
def __init__(self, *, vllm_config: VllmConfig, prefix: str = "") -> None:
    super().__init__()
    config = vllm_config.model_config.hf_config
    # ...
    vision_encoder_info = get_vision_encoder_info(config)
    self.patch_grid_length = vision_encoder_info.get_patch_grid_length()
    # 池化后的网格长度由步长决定，向上取整
    self.pooled_grid_length = math.ceil(
        self.patch_grid_length / config.spatial_pool_stride
    )
    # ... 初始化 vision_tower, vision_resampler, multi_modal_projector 等

def get_mm_mapping(self) -> MultiModelKeys:
    """返回多模态模型的模块前缀，供 LoRA 框架定位子模块"""
    return MultiModelKeys.from_string_field(
        language_model="language_model",
        connector="multi_modal_projector",
        tower_model="vision_tower",
    )

def get_num_mm_encoder_tokens(self, num_video_tokens: int) -> int:
    """从 LLM 侧的占位 token 数 (pooled_grid_length^2 per frame)
    反算 vision tower 实际输出的 token 数 (patch_grid_length^2 per frame)。
    """
    pooled_tokens_per_frame = self.pooled_grid_length ** 2
```

```

if num_video_tokens <= 0 or pooled_tokens_per_frame <= 0:
    return 0
num_frames = num_video_tokens // pooled_tokens_per_frame
return num_frames * (self.patch_grid_length ** 2)

def get_num_mm_connector_tokens(self, num_vision_tokens: int) -> int:
    """从 vision tower 输出 token 数反算 connector 实际输入的 token 数
    (即 pooled_grid_length^2 per frame, 因为 projector 保持长度不变)。
    """
    patch_tokens_per_frame = self.patch_grid_length ** 2
    if num_vision_tokens <= 0 or patch_tokens_per_frame <= 0:
        return 0
    num_frames = num_vision_tokens // patch_tokens_per_frame
    return num_frames * (self.pooled_grid_length ** 2)

```

评论区精华

本 PR 无实质性 review 审查讨论；仅收到 claude bot 自动评论（fork 项目无法自动审查）和合并者 nooooop 的批准 ("thanks!")。

- 设计权衡：累加器 token 计数与池化反推 (design): 采用存储网格长度的方式实现 token 计数反转，并通过测试覆盖真实配置和非标准配置，确认设计正确。

风险与影响

- 风险：token 计数逻辑正确性：get_num_mm_encoder_tokens 和 get_num_mm_connector_tokens 依赖于 patch_grid_length 和 pooled_grid_length 的准确推导。如果 get_vision_encoder_info 返回错误值，或 spatial_pool_stride 配置异常，可能导致 LoRA 层输入尺寸不匹配。测试覆盖了真实配置和多种网格组合，降低了风险。无回归风险：变更仅在该模型类内添加新方法和属性，不影响其他模型或模块。性能影响：仅增加两次整除运算，无显著开销。兼容性：新增方法遵循已有接口约定（SupportsLoRA、MultiModelKeys），对旧版本无影响。
- 影响：用户影响：现在可以对 LLaVA-NeXT-Video 系列模型（如 llava-hf/LLaVA-NeXT-Video-7B-hf）使用 LoRA 微调，包括视觉塔和连接器，提升多模态任务的适配灵活性。系统影响：仅影响单个模型类，无跨模块副作用。团队影响：遵循了已有的模式（如 LLaVA #31513），降低了后续维护和扩展相似模型 LoRA 支持的难度。
- 风险标记：新 token 计数逻辑，LoRA 接口首次用于该模型，依赖 get_vision_encoder_info

关联脉络

- PR #31513 Enable LoRA for LLaVA tower and connector: 本 PR 直接复用 #31513 引入的实现模式（如 get_mm_mapping、packed_modules_mapping、token 计数 helper），作为多模态模型 LoRA 支持的参考模板。
- PR #31479 Enable LoRA support for tower and connector in more MM models: 本 PR 旨在完成该 Issue 中列出的 LLaVA-NeXT-Video 项目，是整体计划的一部分。