PR #38410 完整报告

vllm-project/vllm

[Transformers v5] fix missing pixtral/voxtral multimodal dispatch

合并时间：2026-03-29 17:59

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/38410

执行摘要

本 PR 修复了 Transformers v5 更新导致的 pixtral 和 voxtral 多模态处理器参数缺失问题，通过显式传递 image_processor 和 feature_extractor 到处理器构造函数，确保图像和音频功能正常运行，避免了运行时错误和测试失败。

功能与动机

由于 Transformers v5 重构了处理器构造逻辑（如 issue #38382 所述），mistral 处理器在初始化时只显示 tokenizer，导致 pixtral 的图像处理器和 voxtral 的特征提取器被遗漏，引发 RuntimeError: Expected there to be 3 image items... 等异常。PR 旨在解决此兼容性问题，恢复多模态模型的正常功能。

实现拆解

实现分为两个层面：

- 模型层（vllm/model_executor/models/pixtral.py 和 voxtral.py）：添加了 get_image_processor() 和 get_feature_extractor() 方法，并修改 get_hf_processor() 以显式传递这些实例，替代原有的 self.ctx.init_processor 调用。例如，在 pixtral.py 中：python def get_hf_processor(self, **kwargs) -> MistralCommonPixtralProcessor: return MistralCommonPixtralProcessor(tokenizer=self.get_tokenizer(), image_processor=self.get_image_processor(),)
- 处理器层（vllm/transformers_utils/processors/pixtral.py 和 voxtral.py）：更新了 MistralCommonPixtralProcessor 和 MistralCommonVoxtralProcessor 的构造函数，使其接受 image_processor 和 feature_extractor 作为参数，而非内部实例化。

评论区精华

Reviewer DarkLight1337 提出设计建议：

> "Can you refactor this so that tokenizer and image_processor are initialized in the vLLM multi-modal processor, similar to GLM4VProcessingInfo?" 作者采纳此建议，在提交中调整了初始化位置，优化了代码结构，提高了与现有多模态处理模式的一致性。

风险与影响

- 风险：核心多模态路径变更可能影响其他 Mistral 模型或相关功能，但修改范围小且测试通过（PR body 展示了多个测试成功），风险较低。对 Transformers 版本的依赖性需持续关注兼容性。

- 影响：修复了 pixtral 和 voxtral 模型在多模态场景下的 bug，确保用户能正常使用图像和音频输入，避免了生产环境崩溃。影响范围限于特定模型，但提升了系统稳定性。

关联脉络

与历史 PR 如 #35367（新增 Qwen3 多模态支持）和 #38418（修复多模态缓存问题）相关联，显示了 vLLM 仓库中多模态功能的持续演进和 bug 修复趋势。同时，issue 评论提到 Transformers PR #43514 是导致不兼容的根源，凸显了外部库更新对系统的影响。