PR #44028 完整报告

vllm-project/vllm

[ROCm][CI] Fix failure in the Phi3V pooling test

合并时间：2026-05-30 12:14

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/44028

执行摘要

- 一句话：分离 Phi3V 测试中特殊 token 验证用例
- 推荐动作：建议接受此 PR。变更清晰、动机明确，且拆分后的测试覆盖更精确。可作为测试分离重构的参考案例。

功能与动机

在 ROCm 平台上，test_models_image 测试因合成特殊 token 提示词的嵌入数值在严格阈值下漂移而失败。PR 描述指出原测试混合了两种覆盖：正常的嵌入数值验证和特殊 token 预处理验证。分离后，新测试直接比较 token id，避免数值精度问题，同时仍能捕获缺失的图像 token 扩展问题。

实现拆解

1. 新增常量与辅助函数：在测试文件头部定义 SPECIAL_TOKEN_IMAGE_PROMPT 常量（包含特殊 token 的合成提示词）和 _get_cherry_blossom_image() 辅助函数加载测试图片。
2. 移除旧测试中的特殊 token 分支：从 test_models_image 中删除附加特殊 token 提示词和图片的逻辑，保持其仅测试标准嵌入数值对比。
3. 新增独立测试：test_models_image_special_tokens_processing 使用 ModelConfig 和 MULTIMODAL_REGISTRY 创建 vLLM 处理器，调用处理器处理特殊 token 提示词；同时使用 HF 处理器处理相同输入，并将 HF 的负占位符 token id 映射为图像 token id。然后断言 vLLM 输出的 prompt_token_ids 与 HF 转换后的结果完全一致，并确保图像 token 占位符数量大于 0。
4. 测试配置调整：新测试使用 @pytest.mark.core_model 标记，不包含 @large_gpu_test 装饰器（可能不依赖于大 GPU）。

关键文件：

- tests/models/multimodal/pooling/test_phi3v.py（模块 测试覆盖；类别 test；类型 test-coverage；符号 _get_cherry_blossom_image, test_models_image_special_tokens_processing）：唯一修改的文件，将混合的特殊 token 测试分离成独立测试，并引入处理器比较逻辑。

关键符号：_get_cherry_blossom_image, test_models_image_special_tokens_processing

关键源码片段

tests/models/multimodal/pooling/test_phi3v.py

唯一修改的文件，将混合的特殊 token 测试分离成独立测试，并引入处理器比较逻辑。

```
# 新测试：验证 vLLM 处理器对特殊 token 提示词的 token 化与 HF 一致
@pytest.mark.core_model
@pytest.mark.parametrize("model", MODELS)
@pytest.mark.parametrize("dtype", ["half"])
def test_models_image_special_tokens_processing(
    model: str,
    dtype: str,
) -> None:
    # 使用 ModelConfig 和 MULTIMODAL_REGISTRY 创建 vLLM 处理器
    model_config = ModelConfig(
        model,
        runner="pooling",
        trust_remote_code=True,
        dtype=dtype,
        max_model_len=1024,
    )
    processor = MULTIMODAL_REGISTRY.create_processor(model_config)
    image = _get_cherry_blossom_image() # 加载预定义图片

    # 调用 vLLM 处理器
    processed_inputs = processor(
        SPECIAL_TOKEN_IMAGE_PROMPT,
        mm_items=processor.info.parse_mm_data({"image": image}),
        hf_processor_mm_kwargs={},
    )

    # 获取 HF 处理器并处理相同输入
    hf_processor = processor.info.get_hf_processor()
    hf_inputs = hf_processor(
        SPECIAL_TOKEN_IMAGE_PROMPT,
        images=image,
        return_tensors="pt",
    )

    # 将 HF 的负占位符 token id（如 -1）映射为图像 token id
    image_token_id = hf_processor.get_special_image_token_id()
    hf_prompt_token_ids = [
        image_token_id if token_id < 0 else token_id
        for token_id in hf_inputs["input_ids"][0].tolist()
    ]

    # 断言 vLLM 输出与 HF 转换后的 token id 完全一致
    prompt_token_ids = processed_inputs["prompt_token_ids"]
    assert prompt_token_ids == hf_prompt_token_ids
    # 确保图像 token 占位符数量一致且不为零
    assert prompt_token_ids.count(image_token_id) == hf_prompt_token_ids.count(image_token_
```

```
    id)
    assert prompt_token_ids.count(image_token_id) > 0
```

评论区精华

没有 review 评论或讨论，仅有一位审核者 noooop 批准了变更。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。变更仅为测试用例拆分，不涉及生产代码。新测试不依赖嵌入数值比对，而是直接比较 token id，避免了 ROCm 上的数值精度问题。但需注意：新测试依赖于 ModelConfig 和 MULTIMODAL_REGISTRY 的正确性，如果这些模块有改动可能影响测试稳定性。
- 影响：影响范围限于 ROCm 平台上的 Phi3V pooling 测试。修复了 CI 失败，提高了测试的稳定性和可维护性。对用户无直接影响。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR