PR #47008 完整报告

vllm-project/vllm

[XPU] exclude unsupported models for test_tensor_sechma.py

合并时间：2026-06-29 20:30

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/47008

执行摘要

- 一句话：XPU 多模态测试排除不支持的模型
- 推荐动作：该 PR 价值较低，属于特定硬件平台的测试适配。但其中集中管理排除列表、避免 CI 配置与测试代码重复的做法值得借鉴。阅读量不大，建议仅对 Intel XPU 开发和 CI 配置维护者精读。

功能与动机

PR body 指出：某些多模态模型（如 mistralai/Mistral-Large-3-675B-Instruct-2512-NVFP4、baidu/UnlimitedOCR 等）在 XPU 平台上不受支持，应在 test_tensor_schema.py 之类的测试中排除，以避免 CI 失败。

实现拆解

1. 在 test_common.py 中添加 XPU 模型排除列表 - 新增 _XPU_EXCLUDED_MODEL_IDS 集合，列出三个不受支持的模型 ID：baidu/Unlimited-OCR、mistralai/Mistral-Large-3-675B-Instruct-2512-NVFP4、Qwen/Qwen2.5-Omni-7B-AWQ。 - 导入 vllm.platforms.current_platform 以便运行时检测 XPU。
2. 在 _get_model_ids_to_test 中应用过滤 - 函数先按原逻辑收集所有待测试模型 ID，然后若当前平台是 XPU，则循环从列表中移除排除列表中的模型 ID（使用 while ... in ...: remove() 处理可能重复的情况）。 - 此方式集中处理了所有调用 _get_model_ids_to_test 的测试，确保 test_tensor_schema.py 和将来其他测试都能受益。
3. 简化 Intel CI 配置 - 在 .buildkite/intel_jobs/models_multimodal_intel.yaml 的 Multi-Modal Processor 测试命令中，删除了先前用于跳过相同模型的两个 --deselect 参数。现在过滤逻辑已集中在测试代码中，无需在 CI 命令层重复配置，减少维护负担。

关键文件：

- tests/models/multimodal/processing/test_common.py（模块 测试工具；类别 test；类型 test-coverage；符号 _get_model_ids_to_test）：核心变更：添加了 XPU 模型排除列表和过滤逻辑，集中控制多模态 tensor schema 测试的模型矩阵。
- .buildkite/intel_jobs/models_multimodal_intel.yaml（模块 CI 配置；类别 config；类型 configuration）：删除 CI 配置中重复的 --deselect 参数，使过滤逻辑集中于测试代码，减少维护点。

关键符号：_get_model_ids_to_test

关键源码片段

tests/models/multimodal/processing/test_common.py

核心变更：添加了 XPU 模型排除列表和过滤逻辑，集中控制多模态 tensor schema 测试的模型矩阵。

```
# 来源：tests/models/multimodal/processing/test_common.py

from vllm.platforms import current_platform # 新增导入，用于检测当前硬件平台

# ...（其它代码保持不变）

# 定义 XPU 平台不支持的模型 ID 集合（集中管理，方便未来增删）
_XPU_EXCLUDED_MODEL_IDS = {
    "baidu/Unlimited-OCR", # OCR 模型可能缺少 XPU 算子
    "mistralai/Mistral-Large-3-675B-Instruct-2512-NVFP4", # NVFP4 格式在 XPU 上不支持
    "Qwen/Qwen2.5-Omni-7B-AWQ", # AWQ 量化可能有平台依赖
}

def _get_model_ids_to_test(model_arch_list: AbstractSet[str]):
    # 先收集所有候选模型 ID
    model_ids = list(_iter_model_ids_to_test(model_arch_list))

    # 若当前为 XPU 平台，则从列表移除不支持的模型
    # 注意：使用 while-in-remove 处理列表中可能出现的重复 ID
    if current_platform.is_xpu():
        for excluded_model_id in _XPU_EXCLUDED_MODEL_IDS:
            while excluded_model_id in model_ids:
                model_ids.remove(excluded_model_id)

    return model_ids
```

评论区精华

Review 中主要涉及代码风格和改进建议：

- jikunshang 建议使用 list.remove 替代创建新列表（但最终实现采用了 while remove 模式，保留了原有结构）。
- Copilot 指出过滤循环复杂度为 O(n*m)，建议改用列表推导式一次过滤；但该建议未被采纳，作者保持了原始的 remove 循环。
- Copilot 还指出 PR 标题中 test_tensor_sechma.py 拼写错误（应为 schema），但未修正。
- jikunshang 在建议中修复了 baidu/UnlimitedOCR 的拼写（缺少连字符），最终版本已正确使用 baidu/Unlimited-OCR。
- 过滤循环实现复杂度 (performance): 未采纳建议，作者保留了 while-remove 循环。虽然性能不是关键（列表很小），但可读性略差。
- PR 标题拼写错误 (documentation): 未修正，PR 标题保留了 typo。

风险与影响

- 风险：低风险。仅修改测试辅助函数和 CI 配置，不影响核心运行时。主要风险在于：若未来 XPU 支持了这些模型，需要从排除列表中移除；排除列表的维护需要人工同步。
- 影响：
  - 对用户：无直接影响。
  - 对系统：Intel XPU 上的 multimodal tensor schema 测试将跳过不支持的模型，避免因缺少算子或权重格式（NVFP4）导致测试失败。
  - 对团队：测试过滤逻辑集中到一处，CI 配置更简洁，降低维护成本。
  - 风险标记：暂无

关联脉络

- 暂无明显关联 PR