PR #42660 完整报告

vllm-project/vllm

[Bugfix] Fix incorrect chat template format for Qwen3.5

合并时间：2026-05-15 11:52

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/42660

执行摘要

- 一句话：修复 Qwen3.5 聊天模板格式检测错误
- 推荐动作：建议精读。该 PR 虽小，但展示了 Jinja2 AST 解析的边界情况处理，对理解 vLLM 的聊天模板自动检测机制有参考价值。

功能与动机

PR body 指出 Qwen3.5 的聊天模板内容格式被错误检测为 string 而非 openai，导致多模态输入与文本间多出换行，且交错显示默认不生效（除非手动设置 --interleave-mm-strings）。作者在调试 speculators 仓库的 token 不匹配问题时才发现此问题。

实现拆解

1. 修复 _iter_nodes_assign_content_item 函数（vllm/renderers/hf.py）：在原有的 for content in message['content'] 检测逻辑之后，新增一条分支：当循环迭代器是一个 jinja2.nodes.Name 节点且其名称恰好为 "content" 时，也将其识别为内容项循环，并 yield 对应的循环目标变量名。这样就能捕获类似 {%- for item in content -%} 的写法。
2. 添加测试用例（tests/renderers/test_hf.py）：在 test_resolve_content_format_hf_defined 的参数化列表中增加 ("Qwen/Qwen3.5-4B", "openai")，确保模板格式检测对该模型返回 "openai"。

关键文件：

- vllm/renderers/hf.py（模块 渲染器；类别 source；类型 core-logic；符号 _iter_nodes_assign_content_item）：核心修复文件，新增对 for item in content 形式的 AST 检测分支。
- tests/renderers/test_hf.py（模块 测试；类别 test；类型 test-coverage；符号 test_resolve_content_format_hf_defined）：增加 Qwen3.5-4B 的测试用例，确保回归覆盖。

关键符号：_iter_nodes_assign_content_item

关键源码片段

vllm/renderers/hf.py

核心修复文件，新增对 for item in content 形式的 AST 检测分支。

```
# vllm/renderers/hf.py
def _iter_nodes_assign_content_item(root: jinja2.nodes.Node):
    message_varnames = [
        varname for _, varname in _iter_nodes_assign_messages_item(root)
    ]

    # Search for {%- for content in message['content'] -%} loops
    # or {%- for item in content -%} loops
    for loop_ast in root.find_all(jinja2.nodes.For):
        loop_iter = loop_ast.iter
        loop_target = loop_ast.target

        for varname in message_varnames:
            if _is_var_or_elems_access(loop_iter, varname, "content"):
                assert isinstance(loop_target, jinja2.nodes.Name)
                yield loop_ast, loop_target.name
                break

        # 新增：处理直接遍历 content 变量本身的循环
        # 例如 Qwen3.5 模板中的 {%- for item in content -%}
        if isinstance(loop_iter, jinja2.nodes.Name) and loop_iter.name == "content":
            assert isinstance(loop_target, jinja2.nodes.Name)
            yield loop_ast, loop_target.name
```

tests/renderers/test_hf.py

增加 Qwen3.5-4B 的测试用例，确保回归覆盖。

```
# tests/renderers/test_hf.py
# 在 test_resolve_content_format_hf_defined 的参数列表中添加一行
@pytest.mark.parametrize(
    ("model", "expected_format"),
    [
        ("microsoft/Phi-3.5-vision-instruct", "string"),
        ("Qwen/Qwen2-VL-2B-Instruct", "openai"),
        ("Qwen/Qwen2.5-VL-3B-Instruct", "openai"),
        ("Qwen/Qwen3.5-4B", "openai"), # 新增：确保 Qwen3.5 检测为 openai
        ("fixie-ai/ultravox-v0_5-llama-3_2-1b", "string"),
        ("Qwen/Qwen2-Audio-7B-Instruct", "openai"),
        ("meta-llama/Llama-Guard-3-1B", "openai"),
    ],
)
def test_resolve_content_format_hf_defined(model, expected_format):
    # ... 测试逻辑不变
```

评论区精华

review 过程中，gemini-code-assist[bot] 指出了一个潜在问题：新增分支中的 `yield loop_ast, loop_iter.name` 本意是将循环迭代器的名称（即 "content"）作为第二个元素返回，这与函数定义“应返回内容项变量名”的意图不一致，也与已有分支的 `loop_target.name` 不统一。

但该评论未得到作者回复，且该 PR 已经合并。实际上，观察最终合并的代码发现，提交的第二版已将 loop_iter.name 改为 loop_target.name，解决了该问题。

- 新分支 yield 的值应为 loop_target.name 而非 loop_iter.name (correctness): 作者在最终提交中已将 loop_iter.name 修正为 loop_target.name，与函数意图一致。

风险与影响

- 风险：变更范围极小（仅 6 行源码），且逻辑只影响 _detect_content_format 函数的 AST 解析路径。风险较低，主要风险是新分支可能误匹配某些不规范的模板，但错误匹配为 openai 比误判为 string 更安全（因为 string 格式会丢失多模态交错功能）。
- 影响：直接影响 Qwen3.5 系列模型及使用类似 for item in content 模板的其他模型，使其聊天模板格式被正确识别为 openai，从而修复多模态输入换行和交错问题。不影响其他模型。
- 风险标记：极小变更，仅影响模板检测路径

关联脉络

- PR #39337 [Model Runner v2] Oracle for model runner v2 - qwen3 dense model by default [1/N]: 同为 Qwen 系列模型的支持 PR，展示了 vLLM 对 Qwen 模型的持续适配。