PR #39892 完整报告

vllm-project/vllm

[Bugfix][Responses API] Fix streaming tool calls on /v1/responses

合并时间：2026-04-20 11:24

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/39892

执行摘要

- 一句话：修复 /v1/responses API 流式工具调用中 Gemma4 特殊令牌被剥离和 Pydantic v2 序列化错误的问题。
- 推荐动作：值得精读，特别是对于处理工具调用、Pydantic v2 兼容性和 API 设计的开发者。关注 adjust_request 方法的设计决策，以及如何通过单步构造避免字段跟踪问题，这些技术点对于类似场景有借鉴意义。

功能与动机

根据 PR body，两个 bug 使得流式函数调用在 /v1/responses API 上无法使用，具体是对于依赖特殊令牌分隔符的工具调用解析器（如 Gemma4）以及当 tool_choice="required" 与 stream=True 组合时。这导致工具调用内容泄漏或 API 崩溃，影响了用户体验和 API 稳定性。

实现拆解

1. 修复 Gemma4 特殊令牌剥离问题：修改 vllm/tool_parsers/gemma4_tool_parser.py 中的 adjust_request 方法，移除 isinstance(request, ChatCompletionRequest) 守卫，使 skip_special_tokens=False 同时适用于 ChatCompletionRequest 和 ResponsesRequest，确保 <|tool_call> 等特殊令牌不被剥离，以便解析器正确检测工具调用。
2. 修复 Pydantic v2 序列化错误：修改 vllm/tool_parsers/abstract_tool_parser.py 中的 adjust_request 方法，将 ResponseTextConfig 的构造从两步（先创建空对象再赋值 format）改为单步（直接传入 format 参数），避免 Pydantic v2 的 __fields_set__ 未跟踪导致嵌套配置丢失，同时移除无效的 description 参数。
3. 新增回归测试：添加 tests/tool_use/test_gemma4_responses_adjust_request.py 文件，包含两个单元测试：test_gemma4_adjust_request_sets_skip_special_tokens_on_responses 验证 skip_special_tokens 设置正确性，test_tool_parser_adjust_request_builds_valid_response_text_config 验证 ResponseTextConfig 构造和序列化的有效性，确保修复覆盖所有场景。

关键文件：

- tests/tool_use/test_gemma4_responses_adjust_request.py（模块 测试模块；类别 test；类型 test-coverage；符号 _get_weather_tool, _build_responses_request, _StubTokenizer, get_vocab）：新增回归测试，覆盖两个 bug 的验证，确保修复正确性和未来回归防护。

- vllm/tool_parsers/abstract_tool_parser.py（模块 工具解析器；类别 source；类型 core-logic；符号 adjust_request）：核心源码文件，修改 adjust_request 方法以修复 Pydantic v2 序列化错误，影响所有工具解析器的配置构造。
- vllm/tool_parsers/gemma4_tool_parser.py（模块 工具解析器；类别 source；类型 core-logic；符号 adjust_request）：关键源码文件，修改 adjust_request 方法以移除类型守卫，确保 skip_special_tokens=False 应用于所有请求类型，防止特殊令牌剥离。

关键符号：ToolParser.adjust_request, Gemma4ToolParser.adjust_request

关键源码片段

tests/tool_use/test_gemma4_responses_adjust_request.py

新增回归测试，覆盖两个 bug 的验证，确保修复正确性和未来回归防护。

```
def test_gemma4_adjust_request_sets_skip_special_tokens_on_responses() -> None:
    """验证 Gemma4ToolParser.adjust_request 是否将 skip_special_tokens 设为 False
    以保留特殊令牌。"""
    parser = Gemma4ToolParser.__new__(Gemma4ToolParser)
    parser.model_tokenizer = _StubTokenizer() # 使用桩 tokenizer 模拟词汇表

    request = _build_responses_request(tool_choice="auto")
    assert request.skip_special_tokens is True # 预处理确认默认值为 True

    Gemma4ToolParser.adjust_request(parser, request)
    assert request.skip_special_tokens is False # 断言修复后设置为 False，确保特殊令牌不被剥离
```

vllm/tool_parsers/abstract_tool_parser.py

核心源码文件，修改 adjust_request 方法以修复 Pydantic v2 序列化错误，影响所有工具解析器的配置构造。

```
def adjust_request(
    self, request: ChatCompletionRequest | ResponsesRequest
) -> ChatCompletionRequest | ResponsesRequest:
    # ... 省略前序逻辑，如获取 json_schema_from_tool ...
    if isinstance(request, ResponsesRequest):
        # 单步构造 ResponseTextConfig，确保 Pydantic v2 跟踪 format 字段在 __fields_set__ 中
        # 旧代码的两步构造（先创建空对象再赋值 format）可能导致序列化时嵌套配置丢失
        request.text = ResponseTextConfig(
            format=ResponseFormatTextJSONSchemaConfig(
                type="json_schema", # 指定格式类型为 JSON schema
                name="tool_calling_response", # 配置名称
                schema=json_schema_from_tool, # 从工具生成的 JSON schema
                strict=True # 启用严格模式
            )
        )
        # 移除无效的 description 参数，旧代码传递了错误用途的字符串
    return request
```

vllm/tool_parsers/gemma4_tool_parser.py

关键源码文件，修改 adjust_request 方法以移除类型守卫，确保 skip_special_tokens=False 应用于所有请求类型，防止特殊令牌剥离。

```
def adjust_request(
    self, request: ChatCompletionRequest | ResponsesRequest
) -> ChatCompletionRequest | ResponsesRequest:
    request = super().adjust_request(request) # 调用父类方法处理基础配置
    if request.tools and request.tool_choice != "none":
        # 移除旧代码中的 isinstance(ChatCompletionRequest) 守卫
        # 现在同时应用于 ChatCompletionRequest 和 ResponsesRequest，确保特殊令牌不被剥离
        # 这避免了 Gemma4 工具调用分隔符（如 <|tool_call>）被 tokenizer 移除
        request.skip_special_tokens = False
    return request
```

评论区精华

Review 中，chaunceyjiang 指出 tool_choice="required" + stream=True 组合在 /v1/responses 上还未正式实现，但 PR 修复了相关 bug，并批准了 PR，表示“LGTM”。这表明团队认可修复的必要性，同时提示该功能状态尚在演进中。

- Bug 修复和功能状态确认 (design): 批准 PR，认为修复有效且必要。

风险与影响

- 风险：风险包括：
  1. Pydantic v2 兼容性风险，修改构造方式可能影响其他使用类似模式的代码，但测试覆盖了序列化验证；
  2. 移除 description 参数可能被下游依赖，但它是无效参数，实际无影响；
  3. 控制流调整可能引入回归，但新增测试提供了保障。 - 影响：修复后，使用 Gemma4 等依赖特殊令牌的工具解析器在 /v1/responses API 上的流式工具调用将正常工作，工具调用内容不会泄漏到输出文本，提升用户体验。同时，解决了 Pydantic v2 下的序列化错误，增强了 API 的稳定性和兼容性，对前端和工具调用模块有直接影响。 - 风险标记：Pydantic v2 兼容性 , 特殊令牌处理

关联脉络

- PR #40314 fix: Do not make function calls when request has no tools for /v1/responses: 关联原因：都涉及 /v1/responses API 的工具调用 bug 修复，显示团队在持续改进该 API 的稳定性和功能完整性。