

PR #46382 完整报告

vllm-project/vllm

[Bugfix] fix: stream Minimax m2 tool call string arguments

合并时间: 2026-06-25 11:12

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/46382>

执行摘要

- 一句话: 修复 MiniMax M2 流式工具调用参数解析
- 推荐动作: 建议合并。该 PR 修复了明确的 bug, 实现遵循了已有的 Qwen 3 模式, 且经过手动验证。建议后续增加对应的单元测试以避免回归。

功能与动机

跟进 PR #46351 的修复, 解决 MiniMax M2 模型流式输出工具调用时字符串参数无法正确解析的问题。核心问题是当参数值在流式过程中被截断时, 原有的正则表达式要求找到闭合的才匹配, 导致不完整的参数被忽略。

实现拆解

1. 新增正则表达式: 在 `vllm/parser/minimax_m2.py` 中添加 `_PARTIAL_PARAM_RE`, 该表达式匹配未闭合的 `<parameter name="...">...` 模式, 用于捕获流式传输中末尾的不完整参数。
2. 修改已有正则表达式: 将 `_PARAM_RE` 的末尾从强制闭合的 `</parameter>` 改为可选闭合或匹配下一个 `<parameter` 开头, 以便在流式场景中能提前结束一个参数并开始下一个。
3. 扩展参数转换函数: 在 `_minimax_m2_arg_converter` 中, 当 `partial=True` 时, 先用 `_PARAM_RE` 提取完整参数, 然后用 `_PARTIAL_PARAM_RE` 从剩余文本中提取不完整的末尾参数。
4. 注册解析引擎终端: 在 `minimax_m2_config()` 中增加 `'PARAM_START': PARAM_START` 和 `'PARAM_END': PARAM_END` 终端, 并添加从 `TOOL_ARGS` 状态到 `TOOL_ARGS` 状态的 `PARAM_START` 和 `PARAM_END` 转换, 触发 `ARG_VALUE_CHUNK` 事件, 使流式参数能够增量输出。

关键文件:

- `vllm/parser/minimax_m2.py` (模块 解析器; 类别 `source`; 类型 `core-logic`; 符号 `_PARTIAL_PARAM_RE`, `_minimax_m2_arg_converter`, `minimax_m2_config`, `PARAM_START`): 唯一变更文件, 包含所有核心逻辑修改: 新增正则、修改转换函数、更新解析引擎配置。

关键符号: `_minimax_m2_arg_converter`, `minimax_m2_config`

关键源码片段

vllm/parser/minimax_m2.py

唯一变更文件，包含所有核心逻辑修改：新增正则、修改转换函数、更新解析引擎配置。

vllm/parser/minimax_m2.py - 流式工具调用参数修复

新增正则：匹配未闭合的 <parameter> 标签，用于流式部分解析

```
_PARTIAL_PARAM_RE = re.compile(
    r"<\s*parameter\s+name\s*=\s*"
    r"(?:\"(?P<dq_name>[^\"]*)\"|'(?P<sq_name>[^\']*')|(?P<bare_name>[^\s]+))"
    r"\s*>"
    r"(?P<value>.*)$", # 匹配到行尾，无需闭合标签
    re.DOTALL,
)
```

def _minimax_m2_arg_converter(raw_args: str, partial: bool) -> str:

"""将原始参数 XML 转换为 JSON。partial=True 时额外处理末尾不完整参数"""

params: dict[str, object] = {}

先用 _PARAM_RE 提取完整的参数（现已支持未闭合或下一个参数开头作为结束）

for match in _PARAM_RE.finditer(raw_args):

```
    name = (
        match.group("dq_name")
        or match.group("sq_name")
        or match.group("bare_name")
        or ""
    ).strip()
    if not name:
        continue
    params[name] = match.group("value").strip()
```

if partial:

用 _PARTIAL_PARAM_RE 尝试匹配剩余文本中的不完整参数

remaining = _PARAM_RE.sub("", raw_args)

match = _PARTIAL_PARAM_RE.search(remaining)

if match:

```
    name = (
        match.group("dq_name")
        or match.group("sq_name")
        or match.group("bare_name")
        or ""
    ).strip()
    if name:
        params[name] = match.group("value").strip()
```

return json.dumps(params, ensure_ascii=False)

在解析引擎配置中注册 <parameter name= 和 </parameter> 作为终端，以触发增量事件

添加转换：TOOL_ARGS -> PARAM_START -> ARG_VALUE_CHUNK

添加转换: TOOL_ARGS -> PARAM_END -> ARG_VALUE_CHUNK

评论区精华

审核人 bbrowning 确认该修复与 Qwen 3 增量流式字符串参数的修改模式一致，并附带一个验证脚本证明了 before/after 的行为变化（在 PR 评论中显示流式增量从 7 个 delta 变为预期行为）。

- 流式参数解析行为验证 (correctness): 变更验证通过，行为正确。

风险与影响

- 风险：风险较低。变更仅影响 MiniMax M2 模型的解析器，且逻辑清晰，已有审核验证。但缺少单元测试覆盖新增加的正则表达式和 partial 分支，回归风险主要通过手动验证控制。
- 影响：影响范围仅限于使用 MiniMax M2 模型并启用工具调用 (tool calling) 的用户，特别是流式输出场景。修复后，流式工具调用参数将能正确递增输出，提升用户体验。对非流式场景无影响。
- 风险标记：缺少测试覆盖

关联脉络

- PR #46351 [Bugfix] fix: stream Mimimax m2 tool call string arguments: 本 PR 是 #46351 的跟进，修复相同问题但可能包含不同实现或额外修正（#46351 未在历史记录中提供细节）。