

PR #41198 完整报告

vllm-project/vllm

[Bugfix] DSV32/V4 add missing type conversion for non-streaming tool calls

合并时间: 2026-04-29 17:55

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/41198>

执行摘要

- 一句话: 修复 DSV32/V4 非流式 tool call 类型转换缺失
- 推荐动作: 值得精读参考: 该 PR 展示了如何定位并修复一个因缺少类型转换导致的非流式 tool call bug, 代码改动清晰, 测试与源码联动紧密。开发者在实现类似 parser 时可参考其对 schema 类型转换的处理方式。

功能与动机

Issue #41122 报告 DeepSeek V4 非流式 tool calling 中 boolean 参数被当作字符串 (如 "true") 返回, 导致下游代理 (如 ClaudeCode) 输入校验失败。PR body 中通过对比 before/after 的 arguments 值 ("false" → false) 清晰展示了问题。

实现拆解

1. 核心逻辑修改: 在 vllm/tool_parsers/deepseekv32_tool_parser.py 的 extract_tool_calls 方法中, 原来直接对 _parse_invoke_params 返回的 dict 执行 json.dumps, 缺少类型转换。现在先调用 _convert_params_with_schema(invoke_name, param_dict) 将参数按工具 schema 转换为正确类型后再序列化。
2. 测试补充: 在 tests/tool_parsers/test_deepseekv32_tool_parser.py 的 TestExtractToolCalls 类中新增 test_type_conversion_in_non_streaming 方法, 构造一个包含 boolean (enabled) 和 integer (count) 参数的工具, 验证非流式提取后参数类型正确。

关键文件:

- vllm/tool_parsers/deepseekv32_tool_parser.py (模块 工具解析器; 类别 source; 类型 core-logic; 符号 extract_tool_calls): 核心 bug 修复所在, 在 extract_tool_calls 中增加 schema 类型转换步骤
- tests/tool_parsers/test_deepseekv32_tool_parser.py (模块 工具解析器; 类别 test; 类型 test-coverage; 符号 test_type_conversion_in_non_streaming): 新增测试用例验证类型转换正确性, 确保布尔值和整型被正确转换

关键符号: extract_tool_calls, test_type_conversion_in_non_streaming

关键源码片段

[vllm/tool_parsers/deepseekv32_tool_parser.py](#)

核心 bug 修复所在，在 `extract_tool_calls` 中增加 `schema` 类型转换步骤

```
# vllm/tool_parsers/deepseekv32_tool_parser.py
# 修改前的 extract_tool_calls 方法，未使用 schema 转换
# 现在在构建参数前插入 _convert_params_with_schema 调用

for invoke_name, invoke_content in self.invoke_complete_regex.findall(tool_call_match):
    param_dict = self._parse_invoke_params(invoke_content)
    # 新增：根据工具 schema 转换参数类型（如 "true" → True, "42" → 42）
    params = self._convert_params_with_schema(invoke_name, param_dict)
    tool_calls.append(
        ToolCall(
            type="function",
            function=FunctionCall(
                name=invoke_name,
                # 之前直接使用 param_dict，现在使用转换后的 params
                arguments=json.dumps(params, ensure_ascii=False),
            ),
        )
    )
```

tests/tool_parsers/test_deepseekv32_tool_parser.py

新增测试用例验证类型转换正确性，确保布尔值和整型被正确转换

```
# tests/tool_parsers/test_deepseekv32_tool_parser.py
# 新增测试方法，验证非流式提取时参数类型转换正确性

def test_type_conversion_in_non_streaming(self):
    """Non-streaming extraction must convert params using the tool schema."""
    # 构造一个包含 boolean 和 integer 参数的工具
    tool = ChatCompletionToolsParam(
        function=FunctionDefinition(
            name="toggle",
            parameters={
                "type": "object",
                "properties": {
                    "enabled": {"type": "boolean"},
                    "count": {"type": "integer"},
                },
            },
        ),
    )
    parser = make_parser(tools=[tool])
    # 模拟模型输出：参数以字符串形式出现
    model_output = build_tool_call("toggle", {"enabled": "true", "count": "42"})
    result = parser.extract_tool_calls(model_output, None)
    assert result.tools_called
    assert len(result.tool_calls) == 1
    args = json.loads(result.tool_calls[0].function.arguments)
    # 验证参数被正确转换为原生类型，而不是字符串
```

```
assert args == {"enabled": True, "count": 42}
assert isinstance(args["enabled"], bool)
assert isinstance(args["count"], int)
```

评论区精华

该 PR 没有 review 评论讨论。自动审核机器人无反馈，且 reviewer jeejeelee 已直接批准。

- 暂无高价值评论线程

风险与影响

- 风险：变更范围小（仅改 2 个文件，+26/-1），风险低。修改集中于非流式路径，且已有测试覆盖；但测试只覆盖了单一工具和简单参数，未测试多工具调用或复杂嵌套参数场景。另外，`_convert_params_with_schema` 对未知参数的处理（不匹配 schema 时是否静默失败）未在变更中体现，可能需要更多边界测试。
- 影响：影响范围：仅限于 DeepSeekV32/V4 模型的非流式 tool calling 功能，不会影响其他模型或流式模式。影响程度：低。修复了一个用户感知明显的 bug（下游 agent 解析失败），且改动极小。
- 风险标记：测试覆盖较简单，边界场景未充分测试

关联脉络

- PR #41122 [Bug] DeepSeek V4 tool calling returns boolean parameters as quoted strings which causes subagent failed in ClaudeCode: 直接关联的 issue，该 PR 修复该问题