

PR #22624 完整报告

sgl-project/sglang

[Test] Fix flaky test_function_call_specific by adding strict=True

合并时间: 2026-04-12 16:22

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/22624>

执行摘要

该 PR 修复了 `test_function_call_specific` 测试用例的随机失败问题，通过为 `get_weather` 工具添加 `strict=True` 参数，确保在 `tool_choice` 指定具体函数时参数模式被强制约束。这是对 PR #22586 的补充修复，解决了因 #21593 引入的协议语义变更导致的测试不稳定性问题，将测试失败率从约 30% 降至 0%，提高了 CI 可靠性。

功能与动机

问题根源: 在 PR #21593 之后，`tool_choice` 指定具体函数时，如果没有 `strict=True` 参数，系统只会保证生成工具调用，但参数生成是尽力而为的（没有参数模式的约束解码）。

具体表现: 在 Llama-3.2-1B 模型上，当 `temperature=0.8` 时，这导致约 30% 的随机失败率，模型会生成空的 `{}` 参数，导致 `assertIn("city", args_obj)` 断言失败。

修复动机: PR #22586 已经修复了 `test_function_call_required` 测试用例的类似问题，本 PR 是补充修复 `test_function_call_specific` 测试用例，确保测试稳定性。PR body 提供了详细的证据:

- 在 #21593 之前: 10/10 通过
- 在 #21593 之后: 7/10 通过 (30% 失败)
- 本修复后: 11/11 通过

实现拆解

实现非常简单，仅修改了一个文件的一行代码:

文件路径	变更内容	影响
<code>test/registered/openai_server/function_call/test_openai_function_calling.py</code>	在 <code>test_function_call_specific</code> 测试用例中，为 <code>get_weather</code> 工具定义添加 <code>"strict": True</code> 参数	确保参数模式被强制约束，防止模型生成空参数

关键代码片段:

```
{
  "type": "function",
  "function": {
    "name": "get_weather",
    "parameters": {
      "type": "object",
      "properties": {
        "city": {"type": "string"}
      },
      "required": ["city"],
    },
    "strict": True, # 新增行
  },
}
```

评论区精华

review 讨论非常有限，只有 `gemini-code-assist[bot]` 的自动评论：

```
"This pull request updates the test_openai_function_calling.py file by adding the 'strict': True parameter to a function definition schema within a test case. This modification ensures that the test covers the strict mode for OpenAI function calling."
```

从 PR body 和关联 Issue 可以看出，这个修复是基于 PR #22586 的相同模式，两者都针对 #21593 引入的协议语义变更导致的测试不稳定问题。

风险与影响

技术风险：

- 风险极低，仅修改测试文件，不影响生产代码
- 如果 `strict=True` 参数的行为在未来发生变化，可能会影响测试的预期行为，但这种可能性很小

影响分析：

- 测试稳定性：修复后，`test_function_call_specific` 测试用例的随机失败率从约 30% 降至 0%
- CI 可靠性：提高了持续集成测试的可靠性
- 用户影响：对最终用户没有直接影响
- 系统影响：不改变系统功能或性能

关联脉络

与历史 PR 的关系：

1. PR #22586：直接前驱，修复了同一个测试文件中的 `test_function_call_required` 测试用例的相同问题

2. PR #21593: 根本原因, 引入了协议语义变更, 导致 tool_choice 指定具体函数时, 如果没有 strict=True, 参数生成变为尽力而为

演进趋势:

- 这反映了工具调用协议中 strict 参数的重要性逐渐被认识
- 小模型在非严格模式下可能产生不符合参数模式的输出, 需要通过测试约束来保证可靠性
- 测试用例需要与协议语义变更保持同步, 避免因协议更新导致的测试不稳定

跨 PR 模式:

- 这是第二个修复工具调用测试随机失败的 PR, 表明团队正在系统性地解决 #21593 引入的测试回归问题
- 类似的修复模式 (添加 strict=True) 可能还需要应用到其他相关的测试用例中