PR #18174 完整报告

sgl-project/sglang

[Bugfix] Catch errors when DeepSeek-V3.2 generates malformed JSON

合并时间：2026-03-03 16:10

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/18174

执行摘要

本 PR 修复了 DeepSeek-V3.2 模型在工具调用中生成错误 JSON 格式时导致的解析崩溃问题，通过添加错误捕捉机制确保流式输出稳定，是一个针对特定模块的小范围 bugfix。

功能与动机

根据 PR 描述，LLM 可能生成不正确的 JSON 格式内容，必须捕捉此类错误以保证正常流式输出。动机源于处理工具调用参数解析时的鲁棒性需求，具体表述为："LLMs may generate tool call content that are not correct json format, we must catch this error for normal streaming output."

实现拆解

修改了 python/sglang/srt/function_call/deepseekv32_detector.py 文件中的 _parse_parameters_from_xml 函数。关键改动如下：

```
try:
    parameters[param_name] = _partial_json_loads(param_value, Allow.ALL)[0]
except json.JSONDecodeError:
    parameters[param_name] = param_value.strip()
```

当 JSON 解析失败时，回退到使用原始字符串值，避免解析异常中断处理流程。

评论区精华

- gemini-code-assist[bot] 建议："为了更健壮的错误处理，应同时捕捉 IndexError。"
- yuanshaochen 指出："由于 _partial_json_loads 使用 partial_json_parser 包，实际抛出 MalformedJSON 异常，也应被捕捉。" 讨论焦点在于异常捕捉的完整性，但 PR 未采纳这些建议，仅处理了 JSONDecodeError，可能留下未覆盖的异常风险。

风险与影响

- 风险：
  - 未处理所有可能异常（如 IndexError 和 MalformedJSON），可能导致未捕获异常，影响系统稳定性。
  - 缺少单元测试验证错误处理逻辑的正确性，增加回归风险。

- 影响：
  - 对用户：提升 DeepSeek-V3.2 工具调用模块的流式输出可靠性，避免因解析错误导致的崩溃。
  - 对系统：增强错误处理能力，提高整体鲁棒性。
  - 对团队：小范围变更，易于维护，但需后续关注异常捕捉的完善。

关联脉络

从近期历史 PR 看，无直接相关于 DeepSeek 或 function_call 的 bugfix，但此 PR 体现了在 LLM 工具调用中错误处理的通用模式。例如，可参考其他模型类似修复（如 PR 21004 修复 Kimi K2.5 的 AttributeError），以建立更全面的错误处理策略。