PR #25366 完整报告

sgl-project/sglang

[auto-detect] match Ring-2.6/Ling XML kv tool-call format via vocab signature

合并时间：2026-05-21 14:34

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/25366

执行摘要

- 一句话：添加 XML KV 格式的词汇表自动检测
- 推荐动作：推荐快速合并。这是一个设计优雅、测试完备的增量修复，通过简单的词汇特征显著提升兼容性，同时保持精确匹配。

功能与动机

现有 _is_glm45 函数通过 [gMASK]<sop> 等家族标记识别 GLM 模型，但 inclusionAI/Ring-2.6-1T 等模型虽然使用相同 XML-kv 工具调用格式，却具有不同的 family signature，导致 --tool-call-parser auto 无法识别，输出 None 而静默禁用工具调用功能。PR 描述指出这是由用户反馈驱动的修复。

实现拆解

1. 新增检测函数在 python/sglang/srt/managers/template_detection.py 中添加 _is_xml_kv_tool_call(ctx) 函数，检查 ctx.has_vocab("<arg_key>") 和 ctx.has_vocab("<arg_value>")。
2. 注册后备规则在 TOOL_CALL_PARSER_RULES 元组中，在 glm45 规则之后插入新的 DetectionRule，其中 name="xml_kv_tool_call"、value="glm45"、predicate=_is_xml_kv_tool_call。由于顺序敏感，glm45 规则优先匹配真实 GLM 模型。
3. 更新测试套件在 test/registered/unit/managers/test_template_manager.py 中：
   - 在现有参数化测试中添加 xml_kv_tool_call_via_vocab 用例，验证完整检测流程。
   - 新增 test_glm45_rule_precedes_xml_kv_fallback 通过枚举 TOOL_CALL_PARSER_RULES 顺序确认 glm45 在 xml_kv_tool_call 之前。
   - 新增 test_xml_kv_requires_both_arg_tokens 验证仅存在 <arg_key> 或 <arg_value> 之一时不会触发。

关键文件：

- python/sglang/srt/managers/template_detection.py（模块 模板检测；类别 source；类型 core-logic；符号 _is_xml_kv_tool_call）：核心逻辑变更：新增 _is_xml_kv_tool_call 检测函数，并在 TOOL_CALL_PARSER_RULES 中注册新规则。
- test/registered/unit/managers/test_template_manager.py（模块 模板测试；类别 test；类型 test-coverage；符号 test_glm45_rule_precedes_xml_kv_fallback,

test_xml_kv_requires_both_arg_tokens）：测试配套：新增三个测试用例覆盖正常路径、规则顺序和边界条件。

关键符号：_is_xml_kv_tool_call

关键源码片段

python/sglang/srt/managers/template_detection.py

核心逻辑变更：新增 _is_xml_kv_tool_call 检测函数，并在 TOOL_CALL_PARSER_RULES 中注册新规则。

```
# 新增检测函数：基于词汇表的结构性签名
# 当分词器同时包含 <arg_key> 和 <arg_value> 作为新增 token 时匹配
# 例如 inclusionAI/Ring-2.6 使用 GLM 工具调用格式但无 GLM 家族标记
def _is_xml_kv_tool_call(ctx):
    return ctx.has_vocab("<arg_key>") and ctx.has_vocab("<arg_value>")


# 在 TOOL_CALL_PARSER_RULES 中注册新规则，位于 glm45 规则之后
TOOL_CALL_PARSER_RULES = (
    # ... 其他规则
    DetectionRule(name="glm45", value="glm45", predicate=_is_glm45),
    # 新规则：通用后备，用于采用 GLM 格式但无家族标记的模型
    DetectionRule(
        name="xml_kv_tool_call", value="glm45", predicate=_is_xml_kv_tool_call
    ),
    # ... 其余规则
)
```

test/registered/unit/managers/test_template_manager.py

测试配套：新增三个测试用例覆盖正常路径、规则顺序和边界条件。

```
def test_glm45_rule_precedes_xml_kv_fallback(self):
    # 验证 glm45 规则在 xml_kv_tool_call 之前匹配
    # 两者当前都返回 "glm45"，值测试无法检测顺序交换
    rule_index = {rule.name: i for i, rule in enumerate(TOOL_CALL_PARSER_RULES)}
    self.assertLess(rule_index["glm45"], rule_index["xml_kv_tool_call"])


def test_xml_kv_requires_both_arg_tokens(self):
    # 验证单独存在 <arg_key> 或 <arg_value> 不会触发
    template = "Hello {{ user }}"
    force, config = detect_reasoning_pattern(template)
    for vocab in (["<arg_key>"], ["<arg_value>"], []):
        with self.subTest(vocab=vocab):
            result = detect_tool_call_parser(
                template, _DummyTokenizer(vocab), config, force
            )
            self.assertIsNone(result)
```

评论区精华

无 review 评论，无需记录讨论。

- 暂无高价值评论线程

风险与影响

- 风险：低风险。变更集中在自动检测路径，仅影响 --tool-call-parser auto 模式。新增规则仅检查两个特定词汇 <arg_key> 和 <arg_value> 同时存在，误触发概率极低。规则位于 glm45 之后，不影响真实 GLM 模型。测试覆盖了顺序和边界条件。
- 影响：此变更轻微扩大 --tool-call-parser auto 对采用 GLM 样式 XML-kv 格式的第三方模型的自动检测覆盖。用户无需手动指定 --tool-call-parser glm45。现有模型如 Qwen、GLM-4.5 等不受影响。
- 风险标记：暂无

关联脉络

- PR #25824 [Refactor] Encapsulate SWA loc translation inside SWAKVPool with per-batch cache invalidation: 均为 template_detection.py 所在模块的变更，但主题不同（工具调用检测 vs 缓存重构）。