

PR #45832 完整报告

vllm-project/vllm

[Bugfix][Gemma4] Fix parsing when thinking is disabled

合并时间: 2026-06-17 10:41

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/45832>

执行摘要

- 一句话: 修复 Gemma4 禁用思考时工具调用解析错误
- 推荐动作: 值得精读, 尤其是关注 `adjust_request` 的设计模式——通过 `super().adjust_request()` 组合父类逻辑, 并在子类中按条件覆盖特定属性。

功能与动机

当 Gemma4 模型禁用推理 (`enable_thinking=False`) 且使用工具时, `<ltool_call>` 分隔符被错误移除, 导致工具调用返回空列表, 原始 `call:fn{...}` 文本泄漏到 `content` 中。PR #45553 引入的早期返回逻辑未考虑工具场景。

实现拆解

1. 修改 `vllm/parser/gemma4.py` 中的 `adjust_request` 方法:
 - 先调用 `super().adjust_request(request)` 让父类处理常规逻辑
 - 检查 `enable_thinking` 和 `tools_active` (工具存在且 `tool_choice` 不为 "none")
 - 仅当 `enable_thinking=True` 且 `tools_active=False` 时, 才设置 `skip_special_tokens=True` (即移除分隔符)
 - 否则保持 `skip_special_tokens=False`, 让工具调用的分隔符保留
2. 修改测试文件 `tests/tool_use/test_gemma4_responses_adjust_request.py`:
 - 将 `_build_chat_request` 函数改为支持可选的 `chat_template_kwargs` 参数
 - 新增 `test_gemma4_keeps_special_tokens_with_tools_thinking_disabled`: 验证禁用推理 + 工具时, `skip_special_tokens` 应为 `False`
 - 新增 `test_gemma4_strips_special_tokens_when_nothing_to_preserve`: 验证禁用推理 + 无工具时, `skip_special_tokens` 应为 `True` (默认行为)

关键文件:

- `vllm/parser/gemma4.py` (模块 解析器; 类别 `source`; 类型 `core-logic`; 符号 `adjust_request`): 核心逻辑修改: 重构 `Gemma4Parser.adjust_request`, 使其在禁用推理时正确保留工具调用分隔符
- `tests/tool_use/test_gemma4_responses_adjust_request.py` (模块 测试; 类别 `test`; 类型 `test-coverage`; 符号 `_build_chat_request`, `test_gemma4_keeps_special_tokens_with_tools_thinking_disabled`, `test_gemma4_strips_special_tokens_when_nothing_to_preserve`)

e)：测试配套：修正辅助函数以支持 chat_template_kwargs，新增两个测试用例覆盖禁用推理时的工具场景

关键符号：adjust_request

关键源码片段

vllm/parser/gemma4.py

核心逻辑修改：重构 Gemma4Parser.adjust_request，使其在禁用推理时正确保留工具调用分隔符

```
# vllm/parser/gemma4.py (部分代码)
def adjust_request(
    self,
    request: ChatCompletionRequest | ResponsesRequest,
) -> ChatCompletionRequest | ResponsesRequest:
    """Keep special tokens when thinking or tool calls need them.

    ``skip_special_tokens`` must stay ``False`` when there is something to
    preserve: reasoning channel tokens (thinking enabled) or tool-call
    delimiters (tools active). Otherwise keep the default so stray
    delimiters do not leak into content (e.g. ``tool_choice="none"`` with
    thinking disabled).
    """
    request = super().adjust_request(request) # 先让父类处理
    chat_template_kwargs = getattr(request, "chat_template_kwargs", None) or {}
    enable_thinking = chat_template_kwargs.get("enable_thinking", True)
    has_tools = bool(getattr(request, "tools", None))
    tools_active = has_tools and request.tool_choice != "none"
    if not enable_thinking and not tools_active:
        # 禁用推理且无工具时，才移除特殊分隔符
        request.skip_special_tokens = True
    # 其他情况（推理启用或有工具时）保持 False，让解析器处理
    return request
```

评论区精华

无 review 评论，PR 被快速批准。bbrowning 在 PR 中确认了问题复现和修复效果。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。变更仅影响 Gemma4 解析器在禁用推理时的行为，且通过新增测试验证。可能的风险是其他依赖 adjust_request 方法的逻辑因调用 super().adjust_request() 顺序变化而受影响，但由于 gemma4.py 中 super().adjust_request() 会调用解析器引擎的处理，而该变更仅在禁用推理 + 无工具时改变 skip_special_tokens，其他场景行为不变。
- 影响：影响范围限于 Gemma4 模型用户在禁用推理（enable_thinking=False）并使用工具的场景。修复后工具调用恢复正常，工具分隔符不再泄漏到 content 中。未禁用推理或未使

用工具的用户不受影响。

- 风险标记：核心路径变更

关联脉络

- PR #45553 [Feature] Add Gemma4 parser support: 本 PR 修复了 #45553 引入的 bug : early return 逻辑未考虑工具场景
- PR #45795 [Bugfix] Gemma4: skip forced JSON for required/named tool choice: 相关修复, 本 PR 基于 #45795 开发并验证