

PR #51998 完整报告

vllm-project/vllm

chore: Upstream Cohere parser fixes + tests

合并时间: 2026-08-13 20:05

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/51998>

执行摘要

- 一句话: 上游化 Cohere 解析器修复并补 510 行测试
- 推荐动作: 值得精读。重点看两处设计: 一是 `no_tools()` 与 `start_in_answer()` 如何界定 reasoning parser 与 tool parser 的职责边界, 二是测试中 `_token_deltas` 用 `U+FFFD` 缓冲不完整多字节字符、以字节粒度模拟流式输入的手法, 可作为其他 streaming parser 测试的范本。建议维护者关注 melody 0.11.1 的版本约束是否需要在运行时文档和镜像构建中显式声明。

功能与动机

PR body 仅说明 “Adding some local fixes for Cohere parsers alongside corresponding tests”, 没有关联 issue。从变更内容可以还原动机: vLLM 中 Cohere Command 3/4 的 reasoning parser 在流式模式下未启用 `no_tools()`, 导致工具调用提取职责与 tool parser 重叠; tool parser 未启用 `start_in_answer()`, 可能把推理文本中的 JSON 片段误判为工具调用。这些都是 Cohere 内部已修复的问题, 本 PR 将其合入主线并补齐回归测试, 使 vLLM 解析行为与 Cohere 平台一致。

实现拆解

1. 依赖与提示升级: requirements/test/cuda.in 与 requirements/test/rocm.in 将 `cohere_melody>=0.9.0` 提升到 `>=0.11.1`, 锁文件 `cpu.txt`、`cuda.txt`、`rocm.txt` 同步为 0.11.1; `vllm/reasoning/cohere_command_reasoning_parser.py`、`vllm/tool_parsers/cohere_command_tool_parser.py`、`vllm/renderers/cohere.py` 的 `ImportError` 安装提示统一改为 `pip install 'cohere-melody>=0.11.1'`, 因为 `no_tools()` 与 `start_in_answer()` 等 filter 选项依赖新版本。
2. 推理解析器行为修复: `CohereCommand3ReasoningParser` 与 `CohereCommand4ReasoningParser` 的 `streaming_opts` 从 `stream_non_grounded_answer()` 改为 `no_tools().stream_non_grounded_answer()`, 与既有 `unary_opts` 对齐, 明确 reasoning parser 不负责工具调用提取, 流式输出中 `<ISTART_ACTIONI>` 块原样作为 content 交给下游; 同时 `collect_tool_schema` 生成的 EBNF 中 tools 规则由 `ws "," ws` 调整为 `(ws "," ws)` 语义, 允许逗号前出现空白, 匹配模型实际输出格式。
3. 工具解析器行为修复: `CohereCommand3ToolParser` 与 `CohereCommand4ToolParser` 的 `streaming/unary` 配置均增加 `.start_in_answer()`, 约束工具调用只能从

<|START_ACTION|> 锚点之后开始，避免把推理文本中的 JSON 误判为工具调用。

4. 测试配套：新增 tests/tool_parsers/test_cohere_command_tool_parser.py (510 行)，用 MockCohereTokenizer 按 UTF-8 字节模拟逐 token 流，覆盖单 / 多工具调用、仅推理、引用与 emoji 场景，并校验 streaming 与 non-streaming 结果一致；tests/reasoning/test_cohere_command_reasoning_parser.py 删除 ExpectedToolCall 及工具调用断言，改为断言 content 原样输出且 tool_call_deltas == []。
5. 影响面：所有依赖 Cohere Command 3/4 parser 的推理与 tool-calling 路径行为更新，解析结果与 Cohere 平台对齐；测试与依赖变更同时落地，CI 在 CPU/NVIDIA 平台验证。

关键文件：

- vllm/reasoning/cohere_command_reasoning_parser.py (模块 推理解析器；类别 source；类型 core-logic；符号 collect_tool_schema, CohereCommand3ReasoningParser, CohereCommand4ReasoningParser)：核心行为修复：streaming_opts 增加 no_tools()，统一流式与非流式的工具调用职责；同时修正 collect_tool_schema 生成的 EBNF 语法。
- vllm/tool_parsers/cohere_command_tool_parser.py (模块 工具解析器；类别 source；类型 core-logic；符号 BaseCohereCommandToolParser, CohereCommand3ToolParser, CohereCommand4ToolParser)：为 Command 3/4 tool parser 的 streaming/unary 配置增加 start_in_answer()，限制工具调用只从 answer 区域开始，避免误判推理文本。
- tests/tool_parsers/test_cohere_command_tool_parser.py (模块 工具解析器；类别 test；类型 test-coverage；符号 ExpectedToolCall, ToolCallCase, MockCohereTokenizer, _token_deltas)：新增 510 行测试，用字节级 mock tokenizer 系统性覆盖 streaming/non-streaming 工具调用、引用与 emoji 场景，是本次变更的回归防护核心。
- tests/reasoning/test_cohere_command_reasoning_parser.py (模块 推理解析器；类别 test；类型 test-coverage；符号 ExpectedToolCall, ReasoningCase, MockCohereTokenizer, TestIsReasoningEnd)：删除 ExpectedToolCall 及相关断言，与 reasoning parser 不再输出工具调用的新行为对齐，避免测试断言与新职责边界冲突。
- vllm/renderers/cohere.py (模块 渲染器；类别 source；类型 core-logic；符号 _try_import_melody)：更新 melody 安装提示，与 requirements 中 0.11.1 版本下限保持一致，避免用户安装旧版本后出现难排查的 ImportError。
- requirements/test/cuda.in (模块 测试依赖；类别 test；类型 configuration)：cohere_melody 版本下限从 0.9.0 提升到 0.11.1，这是 no_tools() 与 start_in_answer() 等新 filter 选项可用的前提；rocm.in 与三份锁文件同步更新。

关键符号：CohereCommand3ReasoningParser.init,
CohereCommand4ReasoningParser.init, CohereCommand3ToolParser.init,
CohereCommand4ToolParser.init, collect_tool_schema, _try_import_melody,
_token_deltas, _run_streaming_over_deltas

关键源码片段

vllm/reasoning/cohere_command_reasoning_parser.py

核心行为修复：streaming_opts 增加 no_tools()，统一流式与非流式的工具调用职责；同时修正 collect_tool_schema 生成的 EBNF 语法。

```

# vllm/reasoning/cohere_command_reasoning_parser.py
# ``no_tools()`` 让 reasoning parser 在流式路径下也不解析工具调用，
# 工具调用统一交给 tool parser 处理，避免两个解析器重复提取。
# ``stream_non_grounded_answer`` 保持 melody 对不完整引用标记的缓冲。
class CohereCommand3ReasoningParser(BaseCohereCommandReasoningParser):
    def __init__(self, tokenizer: TokenizerLike, *args, **kwargs):
        super().__init__(
            tokenizer,
            *args,
            streaming_opts=(
                PyFilterOptions().cmd3().no_tools().stream_non_grounded_answer()
            ),
            unary_opts=PyFilterOptions().cmd3().no_tools(),
            **kwargs,
        )

class CohereCommand4ReasoningParser(BaseCohereCommandReasoningParser):
    def __init__(self, tokenizer: TokenizerLike, *args, **kwargs):
        super().__init__(
            tokenizer,
            *args,
            streaming_opts=(
                PyFilterOptions().cmd4().no_tools().stream_non_grounded_answer()
            ),
            unary_opts=PyFilterOptions().cmd4().no_tools(),
            **kwargs,
        )

```

评论区精华

review 评论为空，自动化 review 因 fork 被禁用：claude[bot] 提示维护者可评论 [@claude review](#) 手动触发。维护者 DarkLight1337 在 issue 评论区两次运行 [/ci run](#) (Buildkite #83560、#83711)，分别验证首个 commit 与合并 main 后的版本，随后直接 approve。核心设计权衡（[no_tools\(\)](#) 与 [start_in_answer\(\)](#) 的职责划分）没有在评论区展开讨论，但从 diff 中可以明确推断。

- fork PR 自动 review 状态 (other): 未触发额外自动 review，由维护者 DarkLight1337 直接 approve。
- CI 验证触发 (testing): 两次 CI 均已触发，PR 最终合并。

风险与影响

- 风险：依赖版本下限提升到 `cohere_melody>=0.11.1`，低于该版本的环境会直接触发 `ImportError`，安装提示虽已更新，但用户若使用缓存安装或旧 Docker 镜像可能出现版本不一致；reasoning parser 流式路径启用 `no_tools()` 后不再输出 `tool_calls`，依赖该行为的上游调用方需要改用 tool parser；`collect_tool_schema` 的 EBNF 语法放宽了逗号前空白，存在结构化输出解析歧义的微小可能，需回归验证；新增测试基于字节级 mock tokenizer

，未覆盖真实 Cohere tokenizer 的分词边界（如特殊 token 被拆成多个子 token 的情况），真实环境仍有回归风险。

- 影响：用户侧：使用 Cohere Command 3/4 的 tool calling 与 reasoning 输出在升级后获得更准确的解析结果，但必须安装 `cohere-melody>=0.11.1`。系统侧：通过 `no_tools()` 与 `start_in_answer()` 明确了 reasoning parser 与 tool parser 的职责边界，降低两类 parser 的耦合，长期利于维护。团队侧：510 行新增测试显著提高 Cohere 解析功能的回归防护，字节级 mock tokenizer 的设计可复用到其他厂商 parser 的流式测试。CI 影响：新增测试在 CPU/NVIDIA 平台运行，依赖锁文件同步更新。
- 风险标记：依赖版本下限提升 (`cohere-melody>=0.11.1`)，streaming 输出行为变更，reasoning/tool 解析职责分离，测试基于 mock tokenizer

关联脉络

- PR #52021 [Bugfix] Preserve Anthropic `disable_parallel_tool_use`: 同为前端 tool-calling 解析链路的修复 PR，可对比不同厂商 parser 的修复模式；无直接代码依赖。