PR #44301 完整报告

vllm-project/vllm

[Feature][Parser] Support include_reasoning param for non-Harmony models

合并时间：2026-07-10 15:34

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/44301

执行摘要

- 一句话：支持 non-Harmony 模型 include_reasoning 参数
- 推荐动作：建议阅读该 PR 以理解 include_reasoning 的完整实现分层：解析器层负责抑制推理文本，serving 层负责抑制关联元数据。尤其关注 元数据泄漏的修复提交（如 e6fdc09 和 23e716c），以及 depthfirst-app 指出的多个安全边界。该设计可复用至其他需要按需过滤响应内容的场景。

功能与动机

基于 Issue #33915 和 PR 描述，OpenAI Responses API 支持 include_reasoning 参数用于过滤推理内容。vLLM 已有的 harmony 模型支持基于通道的 token 标记，但 non-Harmony 模型缺失此功能。社区需求包括网络流量优化（推理 token 常是最终答案的 10 倍）和安全部署（避免推理过程泄露系统提示）。

实现拆解

1. 协议层新增字段：在 vllm/entrypoints/openai/responses/protocol.py 的 ResponsesRequest 中添加 include_reasoning: bool = True，并根据 review 要求补充字段描述。
2. 解析器层抑制推理文本：在 DelegatingParser.parse_delta（vllm/parser/abstract_parser.py）、ParserEngine.parse_delta（vllm/parser/engine/parser_engine.py）和 HarmonyParser.parse_delta（vllm/parser/harmony.py）中，生成 delta_message 后检查 request.include_reasoning，若为 False 则设置 delta_message.reasoning = None；若仅有推理无其他内容则丢弃整个消息。同时修复 content is None 检查以处理空白字符串场景。
3. 流式路径元数据抑制：在 vllm/entrypoints/openai/chat_completion/serving.py 的 chat_completion_stream_generator 中引入 hide_stream_metadata 标志，当 include_reasoning=False 且 parser is not None 时清空 logprobs、条件隐藏 token_ids，防止通过解码元数据泄漏推理内容。
4. 非流式路径元数据抑制：在同一文件的 chat_completion_full_generator 中添加 suppress_metadata 逻辑，条件与流式对齐（基于 parser is not None），抑制 logprobs 和 token_ids，并统一抑制条件避免因 reasoning 为 None 而跳过抑制。

5. Responses API 适配：在 vllm/entrypoints/openai/responses/serving.py 和 context.py 中，构建响应项时根据 include_reasoning 抑制 reasoning 字段，确保 Responses API 的非流式和 ParsableContext 路径同样生效。
6. 测试与文档：新增 tests/parser/test_include_reasoning.py（456 行）覆盖 Parser 单元测试，包括非流式 / 流式 / 混合工具调用场景；新增 tests/entrypoints/openai/chat_completion/test_include_reasoning.py（158 行）提供 E2E 测试，启动 Qwen3-0.6B 验证 Chat Completions 流式 / 非流式、默认行为。更新 docs/features/reasoning_outputs.md 文档说明用法。

关键文件：

- vllm/entrypoints/openai/chat_completion/serving.py（模块 服务层；类别 source；类型 core-logic；符号 chat_completion_stream_generator, chat_completion_full_generator）：核心 serving 层变更，实现流式和非流式路径的元数据抑制逻辑。
- vllm/parser/abstract_parser.py（模块 解析器；类别 source；类型 core-logic；符号 parse_delta, _extract_tool_calls）：DelegatingParser 的 parse_delta 方法中添加推理抑制逻辑，作为所有非 engine-based 解析器的统一入口。
- vllm/parser/engine/parser_engine.py（模块 引擎解析；类别 source；类型 core-logic；符号 parse_delta）：ParserEngine.parse_delta 中添加抑制逻辑，覆盖 Qwen3、DeepSeek-V4 等 engine-based 解析器。
- tests/parser/test_include_reasoning.py（模块 单元测试；类别 test；类型 test-coverage；符号 ThinkReasoningParser, tokenizer, make_responses_request, make_chat_request）：全面的单元测试，覆盖非流式、流式及混合工具调用场景下的 include_reasoning 行为。
- tests/entrypoints/openai/chat_completion/test_include_reasoning.py（模块 集成测试；类别 test；类型 test-coverage；符号 server, client, test_include_reasoning_true_non_streaming, test_include_reasoning_false_non_streaming）：端到端测试，启动真实模型验证 Chat Completions API 的 include_reasoning 行为。

关键符号：DelegatingParser.parse_delta (vllm/parser/abstract_parser.py), ParserEngine.parse_delta (vllm/parser/engine/parser_engine.py), HarmonyParser.parse_delta (vllm/parser/harmony.py), chat_completion_stream_generator (vllm/entrypoints/openai/chat_completion/serving.py), chat_completion_full_generator (vllm/entrypoints/openai/chat_completion/serving.py), _make_response_output_items (vllm/entrypoints/openai/responses/serving.py)

关键源码片段

vllm/entrypoints/openai/chat_completion/serving.py

核心 serving 层变更，实现流式和非流式路径的元数据抑制逻辑。

```
# vllm/entrypoints/openai/chat_completion/serving.py (streaming generator)
# ...
# When reasoning is hidden, suppress per-token
```

```
# metadata (logprobs, token_ids) on every chunk to
# prevent leaking reasoning tokens through decoded
# token text in logprob entries or raw token IDs.
hide_stream_metadata = (
    not request.include_reasoning and parser is not None
)
if hide_stream_metadata:
    logprobs = None

if delta_message is None:
    # ...
    if output.finish_reason is None and (
        not request.return_token_ids or hide_stream_metadata
    ):
        continue
    delta_message = DeltaMessage()

# ...
include_token_ids = (
    request.return_token_ids and not hide_stream_metadata
)
```

vllm/parser/abstract_parser.py

DelegatingParser 的 parse_delta 方法中添加推理抑制逻辑，作为所有非 engine-based 解析器的统一入口。

```
# vllm/parser/abstract_parser.py (after finalize_generation and _flush_engine_parsers)
# Suppress reasoning deltas if not requested
if delta_message and not request.include_reasoning:
    delta_message.reasoning = None

    # If only reasoning was in the message (no content, no tool_calls)
    # skip emitting entirely
    if not delta_message.content and not delta_message.tool_calls:
        delta_message = None

return delta_message
```

评论区精华

- @chaunceyjiang 最初质疑功能实用性（“I have some reservations about this feature. It doesn't seem particularly useful to me”），但后续要求添加 E2E 测试并最终批准。
- @depthfirst-app 多次指出 流式路径的 logprobs/token_ids 泄漏风险：即使推理文本被去除，客户端仍可通过 logprobs 或 token_ids 恢复推理内容。该问题通过引入 hide_stream_metadata 和 suppress_metadata 标志解决。
- 另一争议是 非流式路径抑制条件不一致：流式以 parser is not None 为条件，非流式却依赖 reasoning is not None，在推理提取失败时可能跳过抑制。最终统一为基于 parser 存在性。

- 还发现 ParserEngine.parse_delta() 和 Responses API 默认路径未实现抑制，均已补充。
- 功能实用性疑问 (question): albertoperdomo2 解释用途后，chaunceyjiang 要求添加 E2E 测试并最终批准。
- 流式路径 logprobs/token_ids 泄漏 (security): 在 chat_completion_stream_generator 中添加 hide_stream_metadata 标志，清空 logprobs 并条件隐藏 token_ids。
- 非流式路径抑制条件不一致 (correctness): 统一非流式路径抑制条件为 parser is not None，与流式一致。
- ParserEngine 未抑制推理 (correctness): 在 vllm/parser/engine/parser_engine.py 中添加抑制逻辑。
- Responses API 默认路径泄漏 (security): 在 vllm/entrypoints/openai/responses/serving.py 中适配抑制逻辑。

风险与影响

- 风险：
  1. 兼容性风险：默认值 True 确保现有请求不受影响，新增字段在协议中为可选。
  2. 泄漏风险：初始实现中流式和非流式路径的 logprobs/token_ids 元数据可能泄漏推理内容，已通过 hide_stream_metadata 和 suppress_metadata 修复。
  3. ParserEngine 覆盖不足：基于 ParserEngine 的模型（Qwen3、DeepSeek-V4、Gemma4、KimiK2）需单独添加抑制逻辑，已在 parser_engine.py 和 harmony.py 中补充，但仍需测试验证每个模型行为。
  4. Responses API 路径遗漏：ParsableContext 实验路径和默认 _make_response_output_items 路径未抑制，已适配，但需 E2E 测试覆盖 Responses API。
  5. 性能影响：引入少量条件判断，对吞吐影响可忽略。
- 影响：
  - 用户：引入 include_reasoning 参数，允许灵活控制推理内容输出，减少网络带宽并满足安全过滤需求。默认行为不变。
  - 系统：Parser 和 serving 层新增分支逻辑，但影响范围限于 Chat Completions 和 Responses API 路径。
  - 团队：需维护测试文件和文档，后续新增模型需确保其推理解析器正确继承抑制逻辑。
  - 风险标记：流式元数据泄漏风险，抑制条件不一致，ParserEngine 覆盖缺失，Responses API 初始遗漏

关联脉络

- PR #44391 [Feature][Parser] Support include_reasoning for non-Harmony models (Rust frontend): Rust 前端对应实现，共享相同测试用例，PR 评论中提及已合并。