

# PR #45807 完整报告

vllm-project/vllm

fix: report stop\_sequence stop\_reason in Anthropic Messages API

合并时间: 2026-08-20 11:23

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/45807>

## 执行摘要

- 一句话: 修复 Anthropic Messages API 上报 stop\_sequence
- 推荐动作: 值得精读: PR 体量小但语义清晰, 是 Anthropic/OpenAI 兼容层修复的典型样例。核心设计决策是用类型 (str vs int/None) 而非单一 finish\_reason 判别 stop 语义, 相关 if/else 分支可作为转换层代码的注释范式; 配套测试用 parent commit 复现的方式也值得借鉴。建议结合官方 Anthropic 文档阅读。

## 功能与动机

PR body 明确指出: Anthropic Messages 适配器 (/v1/messages) 从不把 stop\_sequence 作为停止原因上报。OpenAI 后端在命中配置的 stop 字符串时会把匹配文本写入 choice.stop\_reason, 但转换器只看 finish\_reason 并一律映射为 end\_turn, 导致匹配文本被丢弃。按 Anthropic 规范, 命中 stop sequence 应返回 stop\_reason="stop\_sequence" 并回显 stop\_sequence; 依赖该语义分支的客户端 (如 Claude Code) 无法区分自然结束与 stop-sequence 命中。

## 实现拆解

1. 明确语义边界: 在 vllm/entrypoints/anthropic/serving.py 中, 为 finish\_reason == "stop" 分支增加类型判别——choice.stop\_reason 为 str 时表示命中配置的 stop 字符串, 上报 stop\_reason="stop\_sequence" 并把匹配文本写入 stop\_sequence; int (stop-token-id) 与 None (自然 EOS) 仍映射 end\_turn, 避免改动既有行为。
2. 非流式路径: 在 messages\_full\_converter 中直接实现上述分支, length 与 tool\_calls 两个分支保持原逻辑不变。
3. 流式路径: message\_stream\_converter 在遍历中新增 stop\_sequence 局部变量, 随 finish\_reason 一起在结束 chunk 上记录 choice.stop\_reason; 构造 message\_delta 时若 stop\_sequence 为 str 则直接生成 stop\_reason="stop\_sequence" 与 stop\_sequence 字段, 否则沿用原有 stop\_reason\_map 映射。
4. 测试配套: tests/entrypoints/anthropic/test\_anthropic\_messages\_conversion.py 扩展 \_make\_stream\_chunk 支持 stop\_reason 参数, 新增 TestStopSequenceReason 覆盖非流式 stop 字符串、自然 EOS、int stop-token-id 与流式 stop 字符串 4 个场景; 测试基于 MagicMock(spec=...) 模式, 无需 GPU 或引擎。

5. 兼容与演进：PR 经历 5 次提交与多次 rebase，解决了上游 `_build_anthropic_usage` 改为单参签名带来的冲突，最终合并前 CI 通过、ruff 检查干净。

关键文件：

- `vllm/entrypoints/anthropic/serving.py`（模块 响应转换；类别 `source`；类型 `core-logic`；符号 `messages_full_converter`, `message_stream_converter`）：核心修复所在。非流式 `messages_full_converter` 与流式 `message_stream_converter` 两处 `stop_reason` 映射逻辑均在此文件修改，决定了 Anthropic 响应的正确性。
- `tests/entrypoints/anthropic/test_anthropic_messages_conversion.py`（模块 转换测试；类别 `test`；类型 `test-coverage`；符号 `TestStopSequenceReason`, `test_non_streaming_stop_string_maps_to_stop_sequence`, `test_non_streaming_natural_eos_maps_to_end_turn`, `test_non_streaming_stop_token_id_maps_to_end_turn`）：新增 `TestStopSequenceReason` 覆盖非流式 / 流式共 4 个场景，修复前两个 `stop-string` 用例失败、修复后通过，是修复正确性的主要证据；同时扩展 `_make_stream_chunk` 支持 `stop_reason` 参数。

关键符号：`messages_full_converter`, `message_stream_converter`

## 关键源码片段

### `vllm/entrypoints/anthropic/serving.py`

核心修复所在。非流式 `messages_full_converter` 与流式 `message_stream_converter` 两处 `stop_reason` 映射逻辑均在此文件修改，决定了 Anthropic 响应的正确性。

非流式路径核心分支：

```
def messages_full_converter(
    self,
    generator: ChatCompletionResponse,
) -> AnthropicMessagesResponse:
    # 构造响应骨架，usage 由公共函数统一转换
    result = AnthropicMessagesResponse(
        id=generator.id,
        content=[],
        model=generator.model,
        usage=_build_anthropic_usage(generator.usage),
        kv_transfer_params=generator.kv_transfer_params,
        ec_transfer_params=generator.ec_transfer_params,
    )
    choice = generator.choices[0]

    # 核心修复：finish_reason == "stop" 时，vLLM 会把匹配到的 stop 字符串
    # 放入 choice.stop_reason 字段，此时其类型为 str；若为 int 则是 stop-token-id,
    # 若为 None 则是自然 EOS (end-of-turn)，后两者必须保持 end_turn 语义。
    if choice.finish_reason == "stop":
        if isinstance(choice.stop_reason, str):
            result.stop_reason = "stop_sequence"
```

```

        result.stop_sequence = choice.stop_reason
    else:
        result.stop_reason = "end_turn"
    elif choice.finish_reason == "length":
        result.stop_reason = "max_tokens"
    elif choice.finish_reason == "tool_calls":
        result.stop_reason = "tool_use"
    # 随后按 thinking / text / tool_use 组装 content 块, 未受本次语义变更影响

```

流式路径核心分支 (message\_stream\_converter 内部) :

```

# 流式转换器内部: 先记录停止信息, 再在 message_delta 事件中上报原因
first_item = True
finish_reason = None
# stop_sequence 保存匹配到的 stop 字符串; int 与 None 都不是 stop sequence
stop_sequence: int | str | None = None
state = _ActiveBlockState()

# ... 处理每个 OpenAI 流式 chunk, 更新 thinking / text / tool_use 分块 ...

# 命中结束 chunk 时, 同时记下 finish_reason 与 choice.stop_reason
if origin_chunk.choices[0].finish_reason is not None:
    finish_reason = origin_chunk.choices[0].finish_reason
    stop_sequence = origin_chunk.choices[0].stop_reason

# 构造 message_delta: str 才上报 stop_sequence, 否则沿用 stop_reason_map
if isinstance(stop_sequence, str):
    stop_delta = AnthropicDelta(
        stop_reason="stop_sequence",
        stop_sequence=stop_sequence,
    )
else:
    stop_delta = AnthropicDelta(
        stop_reason=self.stop_reason_map.get(finish_reason or "stop")
    )
chunk = AnthropicStreamEvent(
    type="message_delta",
    delta=stop_delta,
    usage=_build_anthropic_usage(origin_chunk.usage),
)

```

## tests/entrypoints/anthropic/test\_anthropic\_messages\_conversion.py

新增 TestStopSequenceReason 覆盖非流式 / 流式共 4 个场景, 修复前两个 stop-string 用例失败、修复后通过, 是修复正确性的主要证据; 同时扩展 \_make\_stream\_chunk 支持 stop\_reason 参数。

```

class TestStopSequenceReason:
    """当配置的 stop 字符串命中时, Anthropic Messages API 必须上报
    stop_reason="stop_sequence", 并在 stop_sequence 中回显匹配文本。
    自然 EOS (stop_reason 为 None) 与 stop-token-id (int) 仍映射 end_turn。
    """

```

"""

```
def test_non_streaming_stop_string_maps_to_stop_sequence(self):
    converter = _make_full_converter()
    response = ChatCompletionResponse(
        id="chatcmpl-test",
        model="test-model",
        choices=[
            ChatCompletionResponseChoice(
                index=0,
                message=ChatMessage(role="assistant", content="hello"),
                finish_reason="stop",
                stop_reason="</tool>", # OpenAI 后端回显的匹配字符串
            )
        ],
        usage=UsageInfo(prompt_tokens=5, total_tokens=8, completion_tokens=3),
    )

    result = converter.messages_full_converter(response)

    # 修复前此处为 end_turn / None, 修复后正确标记 stop_sequence
    assert result.stop_reason == "stop_sequence"
    assert result.stop_sequence == "</tool>"

def test_non_streaming_natural_eos_maps_to_end_turn(self):
    converter = _make_full_converter()
    response = ChatCompletionResponse(
        id="chatcmpl-test",
        model="test-model",
        choices=[
            ChatCompletionResponseChoice(
                index=0,
                message=ChatMessage(role="assistant", content="hello"),
                finish_reason="stop",
                stop_reason=None, # 自然结束, 不含 stop 字符串
            )
        ],
        usage=UsageInfo(prompt_tokens=5, total_tokens=8, completion_tokens=3),
    )

    result = converter.messages_full_converter(response)

    # None / int 必须保持 end_turn, 避免回归
    assert result.stop_reason == "end_turn"
    assert result.stop_sequence is None
```

评论区精华

ErenAta16 在评论中给出了独立验证：直接在修复分支运行测试，57 个用例全部通过（无 GPU 环境下 `torch.accelerator.empty_cache()` 的 `teardown` 报错与本次变更无关）；然后切到 `parent commit`、只叠加新测试文件，两个 `stop-string` 用例如预期失败（返回 `end_turn`），证明 bug 真实存在且修复有效。chaunceyjiang 在批准时附上 Claude Messages API 文档链接，确认 `stop_reason="stop_sequence"` 为规范值。另有 `pre-commit` 检查因测试文件末尾多余空行失败，ErenAta16 给出精确 diff，作者在 288152d 修复后转绿。

- 独立验证 bug 真实存在与修复有效 (testing): 修复被独立验证有效，测试用例能复现并验证行为。
- 审批依据 Anthropic 官方 spec (design): 与 Anthropic 规范一致，合并通过。
- `pre-commit` 尾部空行问题 (style): 已修复，`pre-commit` 转绿。
- 多次 `rebase` 与 `merge conflict` (other): 冲突已解决，合并前 CI 通过。

## 风险与影响

- 风险：行为变更：命中 `stop` 字符串时响应由 `end_turn` 变为 `stop_sequence`，依赖旧行为的客户端需要适配；该变更符合 Anthropic 官方规范，属于正确修复而非破坏。`isinstance` 守卫排除了 `int stop-token-id` 与 `None` 两种情形，避免对 `token-id` 场景误报。流式路径的 `stop_sequence` 仅在结束 `chunk` 被记录，若上游把 `finish_reason` 与 `stop_reason` 拆到不同 `chunk` 可能出现丢失，但 OpenAI 流式响应中两者同 `chunk` 到达，风险很低。PR 生命周期较长、经历多次合并 `main`，存在上游漂移风险，但最终 CI 全绿。测试在 MagicMock 层覆盖，未做真实引擎端到端验证，作者也在评论中说明本地无 GPU/torch 环境。整体风险低。
- 影响：对通过 `/v1/messages` 消费响应的客户端可见：`stop` 字符串命中时新增 `stop_sequence` 字段且 `stop_reason` 变为 `stop_sequence`，Claude Code 等依赖该语义的工具可正确区分结束原因。对 vLLM 内部无核心路径影响，改动集中在 Anthropic 转换层；对维护者而言，后续需保持非流式与流式两条路径的一致性。
- 风险标记：行为变更需客户端适配，无 GPU 端到端验证，多次 `rebase` 上游漂移

## 关联脉络

- 暂无明显关联 PR