

PR #49963 完整报告

vllm-project/vllm

[Bugfix] Restore truncate_prompt_tokens for Jina rerank/score online

合并时间: 2026-07-27 22:52

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/49963>

执行摘要

- 一句话: 修复 Jina rerank/score 在线路径 truncate_prompt_tokens 字段丢失
- 推荐动作: 该 PR 是典型的回归修复案例, 值得阅读以理解代理模式下的字段透传陷阱。讨论了 _truncate_scoring_data 与 truncate_prompt_tokens 的职责分离, 对理解 vLLM 的提示处理管线有参考价值。

功能与动机

truncate_prompt_tokens 是 Jina rerank/score 请求的重要参数, 用于控制提示截断。PR #49153 重构引入的代理模式漏掉了该字段的转发, 导致用户在请求中指定 truncate_prompt_tokens 时被 HTTP 400 拒绝。作者在 PR body 中明确指出这是一个回归。

实现拆解

1. 在 vllm/entrypoints/pooling/scoring/io_processor.py 的 get_request_factory_online 方法中, 将 PoolingCompletionRequest 构造调用从仅传 task 和 input 扩展为也传入 truncate_prompt_tokens 和 truncation_side, 这两个值直接从原始 request 对象中读取。
2. 新增单元测试文件 tests/entrypoints/pooling/scoring/test_jina_ranking_io_processor_unit.py, 使用 monkeypatch 将基类的 get_request_factory_online 替换为 spy 函数, 捕获代理请求中的 truncate_prompt_tokens 和 truncation_side 断言其与原始请求一致。

关键文件:

- vllm/entrypoints/pooling/scoring/io_processor.py (模块 入口层; 类别 source; 类型 core-logic): 核心修复文件, 在 proxy 构造中补传 truncate_prompt_tokens 和 truncation_side。
- tests/entrypoints/pooling/scoring/test_jina_ranking_io_processor_unit.py (模块 测试; 类别 test; 类型 test-coverage; 符号 test_online_forwards_truncate_prompt_tokens_to_proxy, _spy_base): 新增单元测试, 确保 proxy 携带了正确的 truncation 字段。

关键符号: JinaRankingIOProcessor.get_request_factory_online

关键源码片段

[vllm/entrypoints/pooling/scoring/io_processor.py](#)

核心修复文件, 在 proxy 构造中补传 truncate_prompt_tokens 和 truncation_side。

```

# vllm/entrypoints/pooling/scoring/io_processor.py
# 修改后的 ctx.request 构造: 转发 truncate_prompt_tokens/truncation_side
# 防止基类工厂读取 ctx.request 时丢失这些字段
ctx.request = PoolingCompletionRequest(
    task="token_embed",
    input=prompts,
    # 必须从原始 request 转发, 否则基类工厂读取 ctx.request 时
    # truncate_prompt_tokens/truncation_side 默认为 None
    truncate_prompt_tokens=request.truncate_prompt_tokens,
    truncation_side=request.truncation_side,
)
requests = PoolingIOProcessor.get_request_factory_online(self, ctx)
ctx.request = request # 恢复原始请求
return requests

```

tests/entrypoints/pooling/scoring/test_jina_ranking_io_processor_unit.py

新增单元测试, 确保 proxy 携带了正确的 truncation 字段。

```

# tests/entrypoints/pooling/scoring/test_jina_ranking_io_processor_unit.py
def test_online_forwards_truncate_prompt_tokens_to_proxy(monkeypatch):
    """验证代理请求携带 truncate_prompt_tokens/truncation_side"""
    proc = JinaRankingIOProcessor.__new__(JinaRankingIOProcessor)
    # Mock 必要方法和属性
    proc.valid_inputs_online = MagicMock(
        return_value=ScoringData(data_1=["query"], data_2=["doc"]))
    proc._get_token_limits = MagicMock(return_value=(0, 0))
    proc.ensure_str = MagicMock(side_effect=lambda data: list(data))
    proc.format_docs_prompts_func = MagicMock(return_value="formatted prompt")

    captured: dict[str, object] = {}

    def _spy_base(self, ctx):
        # 捕获代理请求的字段
        captured["truncate_prompt_tokens"] = ctx.request.truncate_prompt_tokens
        captured["truncation_side"] = ctx.request.truncation_side
        return []

    monkeypatch.setattr(
        PoolingIOProcessor, "get_request_factory_online", _spy_base)

    # 构造一个带有截断参数的请求
    request = RerankRequest(
        model="m", query="query", documents=["doc"],
        truncate_prompt_tokens=512, truncation_side="left",
    )
    ctx = MagicMock()
    ctx.request = request
    ctx.prompt_extras = None

    proc.get_request_factory_online(ctx)

```

```
# 验证代理中字段值与原始请求一致
assert captured["truncate_prompt_tokens"] == 512
assert captured["truncation_side"] == "left"
# 验证 ctx.request 最终恢复为原始请求
assert ctx.request is request
```

评论区精华

审核者 noooooop 在代码行 795 处提出疑问: `_truncate_scoring_data` 是否已经覆盖了截断逻辑? 作者 umut-polat 详细解释了 `_truncate_scoring_data` 只处理 `max_tokens_per_query/max_tokens_per_doc` 的文档级截断, 而 `truncate_prompt_tokens` 作用于格式化后的完整提示, 两者正交, 不能互相替代。此讨论澄清了设计意图, 验证了修复的必要性。

- 校验 `_truncate_scoring_data` 是否已覆盖 `truncation (design)`: 确认需要本修复, 二者分工明确。

风险与影响

- 风险: 本 PR 只改动了一行核心代码 (传递两个额外参数), 且类型匹配, 回归风险极低。新增的单元测试直接验证了修复的正确性。但缺少端到端集成测试 (需要 Jina 模型在 CI 中), 无法覆盖运行时全链路。
- 影响: 修复了 Jina rerank/score 在线路径的回归 bug, 影响所有使用 `truncate_prompt_tokens` 参数的 Jina scoring 请求。其他嵌入路径 (`embed`、`bi/cross-encoder`) 不受影响, 因为它们在之前的重构中已正确处理了这些字段。
- 风险标记: 缺少端到端集成测试, 回归修复

关联脉络

- PR #49153 [Refactor] Use proxy PoolingCompletionRequest in JinaRankingIOProcessor: 本 PR 修复的回归由 PR #49153 引入, 该 PR 引入了代理模式但漏掉了 `truncation` 字段。