

PR #46219 完整报告

vllm-project/vllm

[Rust Frontend] Support echo for token-ID completion prompts

合并时间: 2026-06-23 16:04

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/46219>

执行摘要

- 一句话: 支持 token-ID 类型 prompt 的 echo 回显
- 推荐动作: 此 PR 值得精读, 展示了在 Rust 前端中为 completions 端点添加新功能的清晰模式: 调整验证层、新增转换函数、注入 tokenizer 依赖, 并辅以集成测试。这种模式可复用于其他 feature parity 任务。

功能与动机

Python 前端支持使用 `echo=true` 与 token-ID prompt 组合, 而 Rust 前端之前返回错误。此 PR 是 Rust 前端功能对齐路线图 (#44280) 的一部分, 旨在消除该差异, 使用户能够在 Rust 前端中使用 echo 回显 token-ID 方式传入的 prompt。

实现拆解

1. 移除验证限制 (`validate.rs`): 删除对 `echo=true` 与 `Prompt::TokenIds` 组合的拒绝检查, 并新增单元测试 `validate_request_compat_accepts_token_id_prompt_echo` 确保新路径不再被拒绝。
2. 新增解码回显函数 (`convert.rs`): 新增 `completion_echo_text` 函数, 接受 `CompletionRequest` 和 `&dyn Tokenizer`, 根据 prompt 类型和 `return_token_ids` 标志决定回显文本: 文本 prompt 直接克隆; token IDs 若 `return_token_ids=true` 则返回空字符串 (避免重复解码), 否则调用 `tokenizer.decode` 解码。改造 `prepare_completion_request`, 增加 `tokenizer` 参数并调用 `completion_echo_text` 替代简单取文本。
3. 路由入口适配 (`completions.rs`): 从 `state.chat.text().tokenizer()` 获取 `tokenizer` 实例并传入 `prepare_completion_request`。
4. 集成测试 (`tests.rs`): 添加三个 `tokio` 测试覆盖非流式 echo、流式 echo 以及 `return_token_ids` 场景, 验证 prompt 回显和 token ID 分离的正确性。

关键文件:

- `rust/src/server/src/routes/openai/completions/convert.rs` (模块 转换层; 类别 `source`; 类型 `core-logic`; 符号 `completion_echo_text`, `prepare_completion_request`): 核心转换逻辑, 新增 `completion_echo_text` 函数并改造 `prepare_completion_request` 以接受 `tokenizer` 参数。

- rust/src/server/src/routes/openai/completions/validate.rs (模块 验证层; 类别 source; 类型 guard-removal; 符号 validate_request_compat, validate_request_compat_accepts_token_id_prompt_echo) : 移除对 echo+token IDs 的拒绝检查, 并新增单元测试验证新路径。
- rust/src/server/src/routes/openai/completions.rs (模块 路由入口; 类别 source; 类型 entrypoint; 符号 completions) : 路由入口获取 tokenizer 并传递给 prepare_completion_request。
- rust/src/server/src/routes/tests.rs (模块 集成测试; 类别 test; 类型 test-coverage; 符号 non_stream_completions_echo_decodes_token_id_prompt_text, non_stream_completions_token_id_echo_return_token_ids_keeps_prompt_ids_separate, completions_echo_stream_decodes_token_id_prompt_chunk) : 新增三个集成测试覆盖非流式、流式、return_token_ids 场景。

关键符号: completion_echo_text, prepare_completion_request, validate_request_compat

评论区精华

本 PR 无实质性 review 讨论。BugenZhao 直接批准 (评论 "LGTM") , Codex 自动审查未发现问题。

- 整体审查 (other): 无需修改, 直接合并。

风险与影响

- 风险: 主要风险在于 tokenizer.decode 可能失败 (如模型未加载 tokenizer) , 但代码已通过 map_err 返回 ApiError::invalid_request。行为变更 (之前拒绝 echo+token IDs, 现在接受) 可能影响依赖该错误响应的上游, 但属于对齐预期, 风险较低。解码性能影响可忽略。
- 影响: 影响范围为 Rust 前端 /v1/completions 端点。用户现在可以使用 token-ID prompt 并请求 echo, 这对编程调用者更友好。团队维护成本低, 改动集中在 4 个文件, 测试覆盖充分。
- 风险标记: 行为变更 (先前拒绝现在接受) , 依赖 tokenizer decode

关联脉络

- PR #44280 [Roadmap] Rust Frontend Feature Parity: 此 PR 是 Rust 前端功能对齐路线图的具体步骤, 旨在填补 echo+token-ID 功能缺口。