

PR #46175 完整报告

vllm-project/vllm

[Bugfix] Accept logprobs=-1 in the Completion API

合并时间: 2026-08-18 16:22

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/46175>

执行摘要

- 一句话: 放宽 Completion API 校验, logprobs=-1 合法化
- 推荐动作: 值得精读。虽是小 bugfix, 但 review 中关于“协议层是否应放行引擎已支持的语义”的论证很有价值, 展示了如何用引擎源码定位协议层误判; 可作为小型前端协议修复的规范示例。关注点: 错误消息统一、测试正反路径、与 Chat API / prompt_logprobs 的语义对照。

功能与动机

PR body 指出: Completion API 拒绝 logprobs=-1, 而同一 check_logprobs 校验器中 prompt_logprobs 允许 -1, SamplingParams.__post_init__ 也接受 -1, Chat API 同样接受 top_logprobs=-1。因此 {"prompt": "...", "logprobs": -1} 会在协议层被拒, 根本到不了引擎, 这是协议校验与引擎语义不一致导致的 400 误报。

实现拆解

1. 定位校验入口: vllm/entrypoints/openai/completion/protocol.py 中的 CompletionRequest.check_logprobs (@model_validator(mode="before"))。
2. 修改校验条件: 原条件 (logprobs := data.get("logprobs")) is not None and logprobs < 0 会拒绝 -1, 现加入 logprobs != -1 例外, 并把错误消息从 "must be a positive value." 改为 "must be a positive value or -1.", 与 prompt_logprobs 保持一致。
3. 补充测试: 在 tests/entrypoints/openai/completion/test_completion_error.py 新增 test_logprobs_minus_one_allowed (构造 logprobs=-1 的 CompletionRequest 并通过) 和 test_logprobs_below_minus_one_rejected (断言 -2 抛错且匹配新消息)。
4. 语义确认: review 中确认 logprobs 直接流入 SamplingParams.logprobs, 引擎在 vllm/v1/sample/sampler.py 与 vllm/v1/worker/gpu/sample/states.py 中实现 -1 语义, 协议层应放行。
5. 无配置、部署或 schema 配套变更, 仅源码与测试联动; 测试沿用现有直接构造 CompletionRequest 的风格。

关键文件:

- vllm/entrypoints/openai/completion/protocol.py (模块 协议校验; 类别 source; 类型 core-logic; 符号 check_logprobs): 协议校验主逻辑所在文件, check_logprobs 是拒绝 logprobs=-1 的直接原因, 本 PR 的核心修改就在这里。

- tests/entrypoints/openai/completion/test_completion_error.py (模块 测试用例; 类别 test; 类型 test-coverage; 符号 test_logprobs_minus_one_allowed, test_logprobs_below_minus_one_rejected): 新增两个测试用例覆盖 logprobs=-1 放行与 -2 拒绝, 防止回归并锁定新错误消息, 是验证修复有效性的配套证据。

关键符号: check_logprobs, test_logprobs_minus_one_allowed, test_logprobs_below_minus_one_rejected

关键源码片段

vllm/entrypoints/openai/completion/protocol.py

协议校验主逻辑所在文件, check_logprobs 是拒绝 logprobs=-1 的直接原因, 本 PR 的核心修改就在这里。

```
# vllm/entrypoints/openai/completion/protocol.py —— check_logprobs 关键分支 (head 版本精简)
# mode="before" 校验器直接面对原始请求数据, 先拒绝非数值类型可避免 TypeError 变成 HTTP 500.
for field_name in ("prompt_logprobs", "logprobs"):
    field_value = data.get(field_name)
    if field_value is not None and not isinstance(field_value, (int, float)):
        raise VLLMValidationError(
            f"`{field_name}` must be an integer.",
            parameter=field_name,
            value=field_value,
        )

# prompt_logprobs 原本就允许 -1 (表示 " 返回全部 "), 此处行为不变
if (prompt_logprobs := data.get("prompt_logprobs")) is not None:
    if data.get("stream") and (prompt_logprobs > 0 or prompt_logprobs == -1):
        raise VLLMValidationError(
            "`prompt_logprobs` are not available when `stream=True`.",
            parameter="prompt_logprobs",
        )
    if prompt_logprobs < 0 and prompt_logprobs != -1:
        raise VLLMValidationError(
            "`prompt_logprobs` must be a positive value or -1.",
            parameter="prompt_logprobs",
            value=prompt_logprobs,
        )

# 修复点: logprobs 增加 != -1 例外, 错误消息与 prompt_logprobs 对齐
if (
    (logprobs := data.get("logprobs")) is not None
    and logprobs < 0
    and logprobs != -1
):
    raise VLLMValidationError(
        "`logprobs` must be a positive value or -1.",
```

```
        parameter="logprobs",
        value=logprobs,
    )
```

tests/entrypoints/openai/completion/test_completion_error.py

新增两个测试用例覆盖 logprobs=-1 放行与 -2 拒绝，防止回归并锁定新错误消息，是验证修复有效性的配套证据。

```
# tests/entrypoints/openai/completion/test_completion_error.py —— 新增两个用例
# logprobs=-1 表示 " 返回全部 logprobs": 采样层、prompt_logprobs / top_logprobs
# 校验器都已接受 -1, 这里验证 Completion API 的 logprobs 也能放行。
def test_logprobs_minus_one_allowed():
    request = CompletionRequest(
        model=MODEL_NAME,
        prompt="Test prompt",
        max_tokens=10,
        logprobs=-1,
    )
    assert request.logprobs == -1
```

```
# 比 -1 更小的负值仍然非法，错误消息与 prompt_logprobs 保持一致
def test_logprobs_below_minus_one_rejected():
    with pytest.raises(Exception, match="must be a positive value or -1"):
        CompletionRequest(
            model=MODEL_NAME,
            prompt="Test prompt",
            max_tokens=10,
            logprobs=-2,
        )
```

评论区精华

“Chat API only allows -1 values for prompt_logprobs and top_logprobs, not logprobs” —— DarkLight1337

“the validator was rejecting a value the engine implements for exactly this field … if the preference is to keep /v1/completions strict to the OpenAI shape (non-negative only) … I am happy to drop this, but then the honest fix is a doc note” —— he-yufeng

“Got it, thanks for the explanation” —— DarkLight1337

另有 CI 归因讨论：作者指出 Buildkite 红色任务属于 entrypoints harness，不导入本 PR 改动的协议模块，判定为环境问题而非 diff 问题，请求重试后合入最新 main 验证通过。

- Completion logprobs 是否应允许 -1（与 Chat API 语义对比）(question): 作者给出引擎语义证据后，DarkLight1337 表示 Got it, 并最终 approve。
- CI 失败归因与重试 (other): DarkLight1337 触发 /ci run 两次，最终冲突解决后通过。

风险与影响

- 风险：协议行为变更：logprobs=-1 从 400 变为合法请求，可能影响依赖旧行为的严格客户端；但旧行为本身与引擎语义矛盾，风险较低。与 OpenAI 规范偏差：-1 是 vLLM 扩展语义，文档未显式强调，存在认知成本。测试覆盖：新增用例覆盖核心正反路径，但未覆盖 stream=True 与 -1 组合，也未见 float 负值用例；现有非数值类型检查可拦截字符串，风险可控。无性能、安全影响，改动仅影响协议层校验分支。
- 影响：影响范围集中在 /v1/completions 端点的 logprobs 字段校验：允许用户请求 logprobs=-1 获取全部 token 概率，从 400 变为有效请求。对依赖引擎语义的客户端是兼容性修复，对严格遵循 OpenAI 形状 of 的客户端是行为放宽。团队侧收益是完成协议层与引擎语义的一致性对齐，降低后续维护中“同值不同判”的认知负担。
- 风险标记：协议行为变更，OpenAI 规范偏差，测试覆盖有限

关联脉络

- PR #52622 [Bugfix] Return 4xx for client-caused errors in /detokenize: 同属 entrypoints 层错误处理规范化：本 PR 让 logprobs=-1 从 400 变为合法请求，52622 将真正的客户端错误从 500 归位为 4xx，二者共同完善前端协议的错误码语义。