

PR #55283 完整报告

vllm-project/vllm

[Doc] Clarify ITL vs TPOT Prometheus metrics

合并时间: 2026-09-15 14:35

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/55283>

执行摘要

PR #55283 是一次纯文档 + 示例配置变更: 修正 `docs/design/metrics.md` 中把 `vllm:inter_token_latency_seconds` 标注为 'Time Per Output Token (TPOT)' 的错误说法, 补充真正对应按请求 TPOT 的 `vllm:request_time_per_output_token_seconds`, 并在 `examples/observability/prometheus_grafana/grafana.json` 中新增一个 'Time Per Output Token (TPOT)' 面板 (P50/P90/P95/P99/Mean)。没有任何运行时代码或指标采集逻辑被改动, 风险极低, 但对依赖 Prometheus/Grafana 做延迟监控的用户有直接的认知纠偏价值。

功能与动机

Issue #55276 指出文档的 Grafana Dashboard 小节写着:

`vllm:inter_token_latency_seconds` - Inter-token latency (Time Per Output Token, TPOT) in seconds.

这与 benchmark CLI 文档的术语体系冲突。vLLM 实际暴露两个不同指标:

指标	采样时机	语义
<code>vllm:inter_token_latency_seconds</code>	每次流式输出事件	相邻输出的墙钟间隔, 即 ITL
<code>vllm:request_time_per_output_token_seconds</code>	每个请求结束时一次	(end-to-end latency - TTFT) / (num_generation_tokens - 1), 即 TPOT

Issue 强调两个会被实际观测到的差异: 投机解码下一次输出可携带多个 token, 此时 Mean ITL 约等于 TPOT 乘以每次输出 token 数; 以及 token-weighted TPOT 与 mean per-request TPOT 因聚合权重不同而给出不同结果。PR body 明确 **Resolves #55276**, 并声明 'This is a documentation-only change; no runtime code was modified'。

实现拆解

1. 修正文档定义 (`docs/design/metrics.md`, +19/-4)

- 指标总览列表中把 `vllm:inter_token_latency_seconds` 细化为 'Inter-token latency (time between consecutive streamed outputs)', 明确采样粒度是「流式输出事件之间」而非「token 之间」。

- Grafana Dashboard 指标子集小节删掉 '(Time Per Output Token, TPOT)' 后缀——这正是 Issue 点名的错误标注。
- 文末迭代流程说明中的 'The inter-token intervals (Time Per Output Token, TPOT)' 也改回 'The inter-token intervals'，消除同一文档内的口径冲突。

2. 新增 TPOT 条目与差异说明段落

- 新增 `vllm:request_time_per_output_token_seconds` 条目，写明公式与「单 token 请求记为 0」的行为。
- 新增一段解释性文字：说明两者在输出捆绑多 token（投机解码）或聚合权重不同时发散；`vllm bench serve` 会排除单 token 请求，因此 Prometheus 直方图统计与 benchmark TPOT 不可直接等同；定义来源统一指向 `docs/benchmarking/cli.md#understanding-the-latency-metrics`。

3. Grafana 面板改造 (`examples/observability/prometheus_grafana/grafana.json`, +167/-1)

- 改写既有 'Inter Token Latency' 面板的 description，说明它衡量连续流式输出间隔、仅在单 token 输出时近似 TPOT。
- 新增 'Time Per Output Token (TPOT)' 面板：id=17, gridPos {h:8, w:12, x:12, y:48}（右半屏，紧随 ITL 面板），单位 s。
- 5 条 target: `P99/P95/P90/P50` 用 `histogram_quantile(q, sum by(le) (rate(..._bucket[$__rate_interval])))`；Mean 用 `rate(..._sum) / rate(..._count)`，即 request-weighted 平均。

4. 验证与配套

- 无运行时代码改动，`vllm/v1/metrics/loggers.py` 与 `vllm/v1/metrics/stats.py` 未触碰。
- JSON 合法性由作者本地 `python -c "import json; json.load(...)"` 校验；CI 仅跑文档构建（`label build-docs`，Read the Docs 预览已生成）。
- 无新增测试：Grafana 面板语义不在现有测试体系覆盖范围内；PR body 依 `AGENTS.md` 说明跳过 model eval。
- 演进上作者侧只有 3 个内容 commit（初版 → 'fixing documentation in metrics.md' → 'Align ITL and TPOT metric definitions with benchmark docs'），branch 上的 244 个 commit 主要是反复 merge main 带来的噪声。

`examples/observability/prometheus_grafana/grafana.json`

新增 'Time Per Output Token (TPOT)' 面板并改写 ITL 面板描述，是本次唯一可执行内容的变更（5 条 PromQL target），也是社区复制参考监控栈的直接来源。

```
// examples/observability/prometheus_grafana/grafana.json
// 新增面板：Time Per Output Token (TPOT)，与既有 Inter Token Latency 面板并排展示
{
  "datasource": { "type": "prometheus", "uid": "${DS_PROMETHEUS}" },
```

```
// 面板描述显式写明口径：每个请求结束时记录一次，即 decode 时间除以（输出 token 数 - 1）
// 并前置提示单 token 请求记为 0、vllm bench serve 会把这些请求排除在 TPOT 统计之外
```

```
"description": "Per-request Time Per Output Token (TPOT) in seconds, computed as (end-to-end latency - TTFT) divided by the number of output tokens minus one. Requests with no more than one output token are recorded as zero; vllm bench serve excludes these requests from its TPOT statistics.",
```

```
// 其余样式字段 (fieldConfig、options、thresholds 等) 与既有 ITL 面板保持一致, 此处省略  
"fieldConfig": { "defaults": { "unit": "s" }, "overrides": [] },
```

```
// x = 12 表示放在右半屏, y = 48 表示紧跟在上一行之后, 避免与既有面板重叠  
"gridPos": { "h": 8, "w": 12, "x": 12, "y": 48 },  
"id": 17,
```

```
"targets": [  
  // P99: 先对 bucket 做 rate, 再按 le 聚合, 最后取直方图分位  
  {  
    "expr": "histogram_quantile(0.99, sum by(le) (rate(vllm:request_time_per_output_token_seconds_bucket{model_name=\"$model_name\"}[$__rate_interval])))",  
    "legendFormat": "P99",  
    "refId": "A"  
  },  
  // P95 / P90 / P50 复用同一条查询骨架, 仅替换分位参数, 这里省略重复项  
  {  
    "expr": "histogram_quantile(0.5, sum by(le) (rate(vllm:request_time_per_output_token_seconds_bucket{model_name=\"$model_name\"}[$__rate_interval])))",  
    "legendFormat": "P50",  
    "refId": "D"  
  },  
  // Mean 使用 request-weighted 平均: sum / count 的 rate 比值, 相当于每个请求先算 TPOT 再取均值  
  {  
    "editorMode": "code",  
    "expr": "rate(vllm:request_time_per_output_token_seconds_sum{model_name=\"$model_name\"}[$__rate_interval])\\n\\nrate(vllm:request_time_per_output_token_seconds_count{model_name=\"$model_name\"}[$__rate_interval])",  
    "legendFormat": "Mean",  
    "refId": "E"  
  }  
],  
  
"title": "Time Per Output Token (TPOT)",  
"type": "timeseries"  
}
```

评论区精华

DarkLight1337 (maintainer) : "Could you provide a screenshot of what the dashboard looks like before vs after your changes?"

ItsRoy69: 贴出 before/after 两张截图。Grafana JSON 没有可自动验证的语义，截图是唯一确认面板布局 (x=12, y=48) 与 id 唯一性的手段。

DarkLight1337: "Maybe we could just copy the definitions from [docs/benchmarking/cli.md](#)?"

ItsRoy69: "Good suggestion. I've aligned the metric definitions with the 'Understanding the latency metrics' section in [docs/benchmarking/cli.md](#)". 这一轮把早期自创的 'Inter-output latency in seconds' 收敛为与 benchmark 文档一致的措辞。

coderabbitai[bot]: 要求补充 `vllm:request_time_per_output_token_seconds` 在 `num_generation_tokens <= 1` 时记为 0，并限定与 `vllm bench serve` 的等价性 (bench 排除这些请求)，同时同步到 dashboard 的 `description`。最终 head 版本已包含该说明。

最终 DarkLight1337 以 "Sorry missed your latest comment" 收尾并 APPROVED，讨论没有遗留阻塞项。

风险与影响

- 运行时风险为零：指标采集路径 (`vllm/v1/metrics/loggers.py`、`vllm/v1/metrics/stats.py`) 完全未动，不影响 serving、精度或性能。
- 观测语义仍有残余歧义：指标名依旧叫 `inter_token_latency_seconds`，已有告警与第三方看板若继续把它当 TPOT，不会被自动纠正，只能靠读者读到新描述。
- 直方图分位精度：新面板 P99 依赖 `histogram_quantile` 与默认 bucket 边界，+Inf 桶内样本会被插值掩盖；同期合入主干的 #48866 正在统一 bucket 默认值，但本 PR 未说明 bucket 影响。
- 面板唯一性与布局：id=17 若与既有面板冲突会造成导入异常，本次仅靠 JSON 解析校验 + 截图人工确认。
- Issue 未闭环部分：Issue #55276 询问的『canonical TPOT PromQL』只落地了 `request-weighted mean` 一种，`token-weighted` 公式与『哪个指标对应 `vllm bench serve` 百分位』仍未给出明确推荐。

影响面：所有使用官方参考 dashboard 或参照 `metrics.md` 定义 SLO 的团队都会受益，但改动强度低——不改变任何指标数值，只改变读法；已导入旧 dashboard 的实例需手动更新。

关联脉络

- Issue #55276 是本次变更的直接来源，其提出的核心问题 (ITL 与 TPOT 在一次输出捆绑多 token 时的数值差异) 在文档层面得到澄清。
- 同期合入主干的 #55016 (修正 `spec-decode` PromQL 示例使用暴露的指标名) 与 #48866 (统一 Prometheus 直方图 bucket 默认值) 说明 vLLM 的 `metrics` 文档与观测基建正在被系统性梳理，本 PR 是同一波整理中的一环。
- 文档中提到 PR #2316 是最初添加该 Grafana dashboard 的改动，本 PR 可视为其后续的语义修正补丁。

结论

价值集中在知识纠偏而非代码改动：值得负责监控 /SLO 的工程师花 5 分钟读

[docs/design/metrics.md](#) 新增段落，并复制 grafana.json 中的 5 条 PromQL；其余开发者可跳过。后续建议补充 token-weighted TPOT 的 PromQL 并明确标注为不推荐用法。