Performance test report - Mar 19, 2024 (#4)

Open in Postman

Postman collection: Sentimetry API

Report exported on: Mar 19, 2024, 23:45:59 (GMT+8)

Test setup

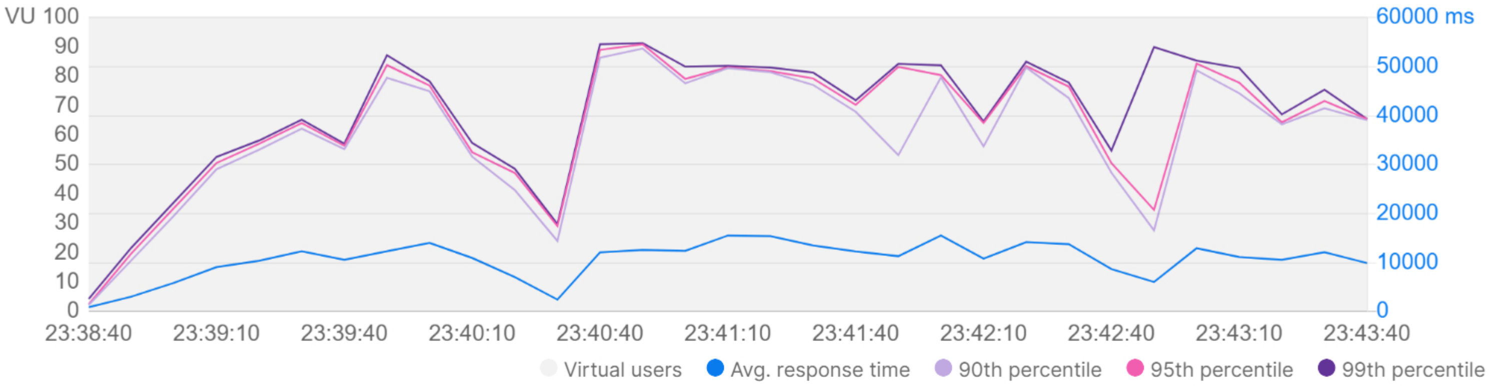
Virtual users 100 VU	Start time Mar 19, 23:38:35 (GMT+8)	Load profile Fixed
Duration 5 minutes	End time Mar 19, 23:43:41 (GMT+8)	Environment -

1. Summary

Total requests sent	Throughput	Average response time	Error rate
2,545	8.32 requests/second	10,054 ms	1.30 %

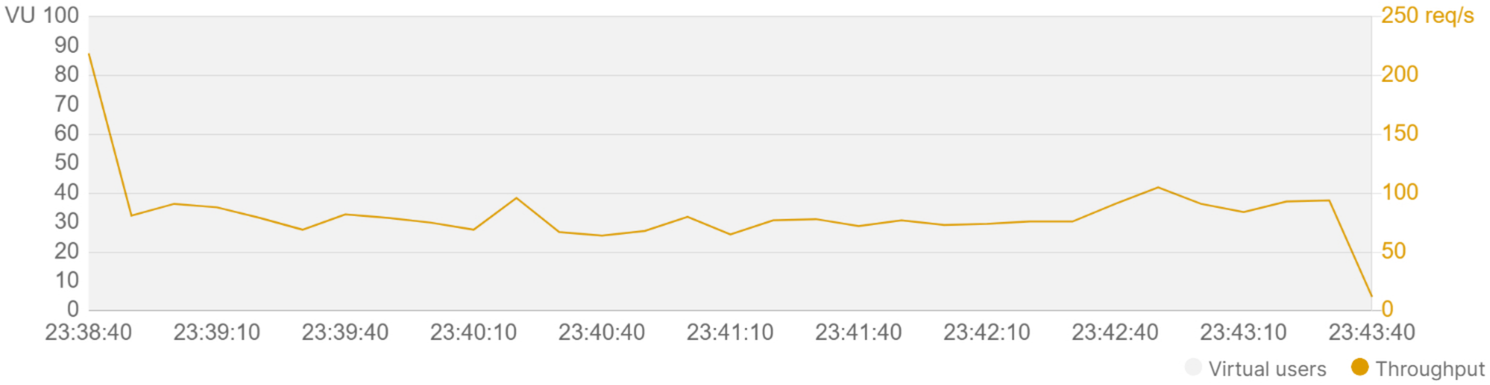
1.1 Response time

Response time trends during the test duration.



1.2 Throughput

Rate of requests sent per second during the test duration.



1.3 Requests with slowest response times

Top 5 slowest requests based on their average response times.

Request	Resp. time (Avg ms)	90th (ms)	95th (ms)	99th (ms)	Min (ms)	Max (ms)
POST EmoRoBERTa {{emoURL}}/emoroberta	32,907	49,345	50,536	54,017	1,418	54,836
POST Logistic Regression {{baseURL}}/logistic-regression	311	893	968	1,100	219	1,278
POST Keras {{baseURL}}/keras	305	340	857	1,450	231	1,545

1.4 Requests with most errors

Top 5 requests with the most errors, along with the most frequently occurring errors for each request.

Request	Total error count	Error 1	Error 2	Other errors
POST EmoRoBERTa {{emoURL}}/emoroberta	33	ESOCKETTIMEDOUT (33)	-	0

2. Metrics for each request

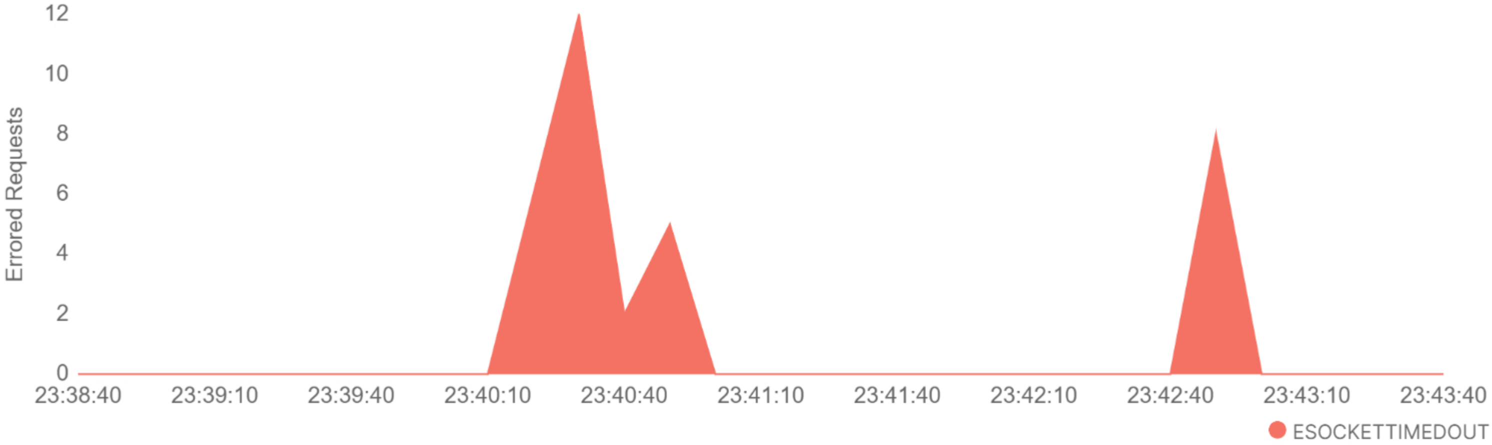
The requests are shown in the order they were sent by virtual users.

Request	Total requests	Requests/s	Min (ms)	Avg (ms)	90th (ms)	Max (ms)	Error %
POST Logistic Regression {{baseURL}}/logistic-regression	881	2.88	219	311	893	1,278	0
POST Keras {{baseURL}}/keras	880	2.88	231	305	340	1,545	0
POST EmoRoBERTa {{emoURL}}/emoroberta	784	2.56	1,418	32,907	49,345	54,836	4.21

3. Errors

3.1 Error distribution over time

Top 5 error classes observed during the test duration.



3.2 Error distribution for requests

Errored requests grouped by error class, along with the error count for each class.

Error class	Total counts
ESOCKETTIMEDOUT	33
POST EmoRoBERTa	33

Testing API performance on Postman

Postman enables you to simulate user traffic and observe how your API behaves under load. It also helps you identify any issues or bottlenecks that affect performance.

Learn more about testing API performance.