Towards Representative Web Performance Measurements with Google Lighthouse

Tjaša Heričko
University of Maribor,
Faculty of Electrical
Engineering and Computer Science,
Koroška cesta 46, 2000 Maribor, Slovenia
tjasa.hericko@um.si

Boštjan Šumak
University of Maribor,
Faculty of Electrical
Engineering and Computer Science,
Koroška cesta 46, 2000 Maribor, Slovenia
bostjan.sumak@um.si

Saša Brdnik
University of Maribor,
Faculty of Electrical
Engineering and Computer Science,
Koroška cesta 46, 2000 Maribor, Slovenia
sasa.brdnik@um.si

Abstract

Web performance testing with tools such as Google Lighthouse is a common task in software practice and research. However, variability in time-based performance measurement results is observed quickly when using the tool, even if the website has not changed. This can occur due to variability in the network, web, and client devices. In this paper, we investigated how this challenge was addressed in the existing literature. Furthermore, an experiment was conducted, highlighting how unrepresentative measurements can result from single runs; thus, researchers and practitioners are advised to run performance tests multiple times and use an aggregation value. Based on the empirical results, 5 consecutive runs using a median to aggregate results reduce variability greatly, and can be performed in a reasonable time. The study's findings alert to potential pitfalls when using single run-based measurement results and serve as guidelines for future use of the tool.

Keywords websites, web performance, performance testing, performance variability, tool, Google Lighthouse

1 Introduction

In software engineering, performance tests are often conducted by software researchers and practitioners to audit a website's quality. The former commonly use web performance measurements to assess web performance on the observed websites [12, 14, 15], to investigate factors (positively or negatively) affecting performance [4, 6, 13], and to improve performance testing [10], while the latter use performance measurements for improving a website's quality to provide a better overall user experience, as web performance influences website traffic, user attrition, user engagement, online revenue, and even rankings in search results greatly [2, 16, 17].

Performance testing can be conducted using various tools, among which Google Lighthouse has gained increasing attention in recent years. It is an open-source tool, providing audits for performance, as well as for accessibility, search engine optimization, and progressive web apps, with indicators on how to improve these aspects of websites if needed [8]. However, when dealing with time-based measurements, the results of such testing can often be inconsistent, as several factors can interfere with the measures and may introduce fluctuations, even if the website has not changed. Most commonly, results tend to vary due to variability in the network, web server, client hardware, and client resource contention [7]. Lighthouse addresses variability by providing vague strategies and recommendations on how to reduce them, though results can still vary. Besides isolating external factors, e.g., using a dedicated device for testing, using a local deployment or a machine on the same network, the most straightforward strategy is to run Lighthouse multiple times and use aggregate values instead of single tests [7].

The research objectives of this paper are: (i) To study how the research community has addressed the challenge of variability in performance measurements when using the tool; and (ii) To demonstrate the strategy of performing multiple runs empirically with their aggregation into a single-value result. To achieve these objectives, we performed a literature review and conducted an experiment.

Our work is broadly related to previous research providing a better understanding and managing of variations in measurements, testing, and benchmarking for timebased performance measurements [3, 5, 10, 11]. These studies are focused primarily on suggesting recommendations for robust testing in the presence of environmental fluctuations, and, as such, are quite different in aim from ours, which is to gain insight into how a specific tool – Google Lighthouse – is used in research for web performance measurement, further investigated with empirical research alerting software researchers and practitioners to potential pitfalls in the future use of the tool.

DOI https://doi.org/10.18690/978-961-286-516-0.9
ISBN 978-961-286-516-0

The contributions of the paper are: (i) Presenting an overview of existing studies using Lighthouse for measuring performance, with an emphasis on how the tool is used, what measuring strategies are employed, and how the authors addressed possible inconsistencies in results; (ii) Providing analysis of the effects of repeating performance measurements to prevent single run's outliers; (iii) Highlighting potential pitfalls for research and practice using single run-based results provided by Lighthouse; and (iv) Serving as a base for research studies on mitigating unrepresentative web performance measurements.

2 Literature review

A literature review was performed to find existing research utilizing Lighthouse as the tool for estimating web performance. A full-text search was conducted using the search string *»Google Lighthouse«* in the following digital libraries: ACM Digital Library[1], Google Scholar[2], IEEE Xplore[3], and Web of Science[4]. The search was carried out on July 28, 2021, and altogether 134 studies were retrieved from the search. Inclusion and exclusion criteria guided the study selection process. Only journal and conference papers were considered. Materials not accessible in English were excluded. Any research that only described Lighthouse theoretically was excluded, and all papers where Lighthouse was not used for performance measurements were excluded as well. After the review process, 8 primary studies were selected.

The list of primary studies is available in Table 1. All primary studies were published in conference proceedings in recent years, in 2018 or later. From the performance measurements made with Lighthouse, primary studies used the Performance Score (S1-S3, S6) most commonly, a single-value indicator of websites' overall performance, for their further analysis. The following more specific time metrics were also used commonly: Speed Index (S2, S4, S7, S8), First Meaningful Paint (S1, S2, S4, S7) and Estimated Input Latency (S1, S2, S7). Researchers observed between 1 and 21 websites in each study. Less than half of the primary studies (S1, S4, S7) have noted some variance between runs when auditing the same website due to uncontrollable variables, and employed some strategies to mitigate this problem. In two studies (S4, S7), the authors repeated runs consequently, while in one study (S1), researchers ran performance audits multiple times trough the day. The number of runs varied from 5 to 100. Two studies (S1, S4) then used mean for aggregating multiple runs into a single value, while one study (S7) used median.

Table 1: A list of primary studies selected in the literature review with their performance audit strategies.

ID	Year	Performance audit strategy	Ref
S1	2018	5 repetitions trough the day	[6]
S2	2019	1 run	[12]
S3	2019	1 run	[15]
S4	2019	100 consecutive repetitions	[13]
S5	2019	1 run	[10]
S6	2020	1 run	[4]
S7	2020	30 consecutive repetitions	[14]
S8	2021	1 run	[18]

3 Experiment

An experiment was performed to demonstrate further how single performance audits can be unrepresentative in some scenarios, and investigate how the number of runs affects variability. In the experiment, 10 real websites were used, selected randomly from the Alexa Top 500 list, which includes top-ranked websites on the web [1]. The selected websites were: AliExpress[5], Amazon India[6], Bola[7], Freepik[8], IKEA[9], Mercari[10], Shopify[11], Unspash[12], Wix[13], and Zendesk[14], further referred to as *W1*, *W2*, *W3*, *W4*, *W5*, *W6*, *W7*, *W8*, *W9*, and *W10*, respectively. The selected websites ranged from simple static presentation websites to dynamic websites, i.e., e-commerce websites and news sites.

All websites were audited with Google Lighthouse v7.3.0 on a device with macOS 10.15.7 using Headless Chrome. A desktop device was emulated using a broadband network connection. For each website, performance audits were performed 4 times, each time with an increasing number of repeated independent runs (N=1,5,10,100). During the auditing process, external factors were isolated as much as possible. The experiment was conducted on July 29, 2021 between 4:42 and 8:37 PM (GMT+2).

From the collected results, we used the Performance Score for analysis, as this value captures the overall web performance of a website. It is calculated as a weighted average of six metric scores, each metric representing some aspect of a website's performance. The Performance Score can have the following values: *Poor* (0-50), *Needs improvement* (50-89) and *Good* (90-100) [9].

Data analysis was conducted using IBM SPSS Statistics v27. Descriptive statistics were used to present the characteristics of sets of data collected with a different number of consecutive runs, including a description and spread of the data in each set. Mood's median test was performed to estimate if the medians of data sets from different runs on the same website were equal.

[1] https://dl.acm.org/
[2] https://scholar.google.com/
[3] https://ieeexplore.ieee.org/
[4] https://apps.webofknowledge.com/
[5] https://www.aliexpress.com/
[6] https://www.amazon.in/
[7] https://www.bola.com/
[8] https://www.freepik.com/
[9] https://www.ikea.com/
[10] https://www.mercari.com/
[11] https://www.shopify.com/
[12] https://unsplash.com/
[13] https://www.wix.com/
[14] https://www.zendesk.com/

4 Results and Discussion

The distribution of Performance Scores of each website for N=100 is presented with boxplots in Figure 1. It can easily be observed that, for almost all websites (except for *W4*), some performance measurements occurred that were not a typical representative of a website's performance. Suppose one of these outlier measurements was the only assessment run, this can lead to an unrepresentative result. Consequently, wrongful conclusions and decisions can be made, e.g., a developer may think a change he implemented into a code recently made performance worse, yet, instead, this occurred due to fluctuation in the network, web, or client device. An interesting observation is that, due to variability in the measurement results, a website can be interpreted in a different score group, e.g., *W1*, *W2*, *W4*, and *W9* results are dispersed between score groups *Good* and *Needs improvement*, and *W3* between score groups *Poor* and *Needs improvement*.

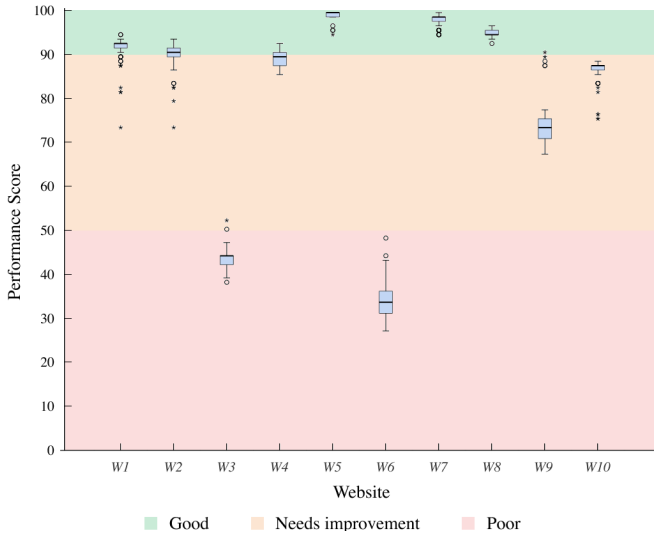

Figure 1: Data distribution of Performance Scores (N=100).

Detailed results for all sets of data are presented in Table 2, providing insight into how the data are spread, how much repeating the test reduces variability and how results stabilize as the number of tests run increases. These results further illustrate the differences between single and multiple runs, which can provide a more reliable estimate of a website's performance; therefore, providing a rationale why addressing intrinsic fluctuations when dealing with time-based metrics should be considered, and why a single run can (in some cases) not be representative enough to provide reliable measurements. We argue that the use of a median value for aggregation is preferred over other measures of central tendency to minimize the impact of outliers.

Mood's median test, performed for N=5, N=10, and N=100, showed that the medians of the Performance Score were the same across all three categories of runs for all websites, except *W5*, where the test could not be performed. These results, presented in Table 3, indicate that 5 runs in comparison to 10 and 100 runs are sufficient to eliminate possible outliers while still performing web performance testing in a reasonable time.

Table 2: Performance scores and their statistics across different numbers of runs.

	Statistics	**N=1**	**N=5**	**N=10**	**N=100**
W1	Mean	87	91	91.2	90.9
	Median	87	92	92	92
	Range	/	6	6	21
	SD	/	2.3	1.9	2.9
W2	Mean	86	86	88.3	89.5
	Median	86	90	90	90
	Range	/	18	20	20
	SD	/	7.5	5.7	2.8
W3	Mean	40	42.8	44.3	43.3
	Median	40	43	44	44
	Range	/	5	10	14
	SD	/	1.9	2.8	1.9
W4	Mean	89	88.4	88.3	88.6
	Median	89	89	88	89
	Range	/	5	5	7
	SD	/	1.9	1.5	1.5
W5	Mean	99	98.6	98.5	98.6
	Median	99	99	98.5	99
	Range	/	1	1	5
	SD	/	0.5	0.5	0.9
W6	Mean	28	35	32.9	33.9
	Median	28	34	31	33.5
	Range	/	15	16	21
	SD	/	7.3	6.1	3.9
W7	Mean	98	98	97.6	97.3
	Median	98	98	98	98
	Range	/	0	2	5
	SD	/	0	0.7	1.2
W8	Mean	94	94.4	94.4	94.5
	Median	94	94	94	94
	Range	/	1	1	4
	SD	/	0	0.5	0.6
W9	Mean	77	72.4	72.1	73.3
	Median	77	72	72	73
	Range	/	7	8	23
	SD	/	2.9	2.5	4.2
W10	Mean	75	83.4	85.3	86.1
	Median	75	86	87	87
	Range	/	12	13	13
	SD	/	4.9	3.9	2.5

5 Conclusion

Several strategies can be employed to reduce random noise, measurement bias and errors when using Lighthouse for web performance measurements. In the paper, we performed a literature review in which we selected studies using Lighthouse for estimating web performance. The results show that more than half of the primary studies did not employ any specific strategy to address variability in web performance measurements. Others use a reasonably straightforward approach to repeat the Lighthouse audit multiple times and summarize repeated runs using a mean or median. However, a large discrepancy was noticed in these works in the number of runs and measures of central tendency used to aggregate multiple

Table 3: Mood's median test by website, comparing groups of N=5, N=10, and N=100.

	W1	*W2*	*W3*	*W4*	*W5*	*W6*	*W7*	*W8*	*W9*	*W10*
Asymp.Sig	0.996	0.723	0.137	0.557	*	0.744	0.927	0.879	0.211	0.758

*All values are less than or equal to the median. Mood's median test could not be performed.

runs into a single-value result. Thus, we investigated this further empirically by conducting an experiment on real popular websites, to demonstrate how the number of runs affects variability and prevents single-run outliers. With this, we highlighted how measurement results from single runs could be misleading and unrepresentative; therefore, we recommend for research and practice to run performance tests multiple times and use an aggregation value. Based on our results, performing performance audits 5 times reduces variability in results greatly in a reasonable time. Our study provides a base for future research studies addressing outliers in web performance testing, and for guidelines for future studies on how to perform representative web performance measurements with Lighthouse.

Acknowledgment

The authors acknowledge the financial support from the Slovenian Research Agency (Research Core Funding No. P2-0057).

References

[1] Alexa Internet, Inc. The top 500 sites on the web, 2021.

[2] Arapakis, I., Bai, X., and Cambazoglu, B. B. Impact of response latency on user behavior in web search. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval* (New York, 2014), SIGIR '14, ACM, pp. 103–112.

[3] Beyer, D., Löwe, S., and Wendler, P. Reliable benchmarking: Requirements and solutions. *International Journal on Software Tools for Technology Transfer 21*, 1 (2019), 1–29.

[4] Chan-Jong-Chu, K., Islam, T., Exposito, M. M., Sheombar, S., Valladares, C., Philippot, O., Grua, E. M., and Malavolta, I. Investigating the correlation between performance scores and energy consumption of mobile web apps. In *Proceedings of the Evaluation and Assessment in Software Engineering* (New York, 2020), EASE '20, ACM, pp. 190–199.

[5] Chen, J., and Revels, J. Robust benchmarking in noisy environments, 2016.

[6] Gambhir, A., and Raj, G. Analysis of cache in service worker and performance scoring of progressive web application. In *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE)* (2018).

[7] Google Developers. Variability, 2019.

[8] Google Developers. Lighthouse, 2021.

[9] Google Developers. Lighthouse performance scoring, 2021.

[10] Johnston, O., Jarman, D., Berry, J., Zhou, Z. Q., and Chen, T. Y. Metamorphic relations for detection of performance anomalies. In *2019 IEEE/ACM 4th International Workshop on Metamorphic Testing (MET)* (2019), pp. 63–69.

[11] Kalibera, T., and Jones, R. Rigorous benchmarking in reasonable time. In *Proceedings of the 2013 International Symposium on Memory Management* (New York, 2013), ISMM '13, ACM, pp. 63–74.

[12] Nurshuhada, A., Yusop, R. O. M., Azmi, A., Ismail, S. A., Sarkan, H. M., and Kama, N. Enhancing performance aspect in usability guidelines for mobile web application. In *2019 6th International Conference on Research and Innovation in Information Systems (ICRIIS)* (2019), pp. 1–6.

[13] Rahman, S., and Wittie, M. P. MR-DNS: Multi-resolution Domain Name System. In *Internet and Distributed Computing Systems* (Cham, 2019), R. Montella, A. Ciaramella, G. Fortino, A. Guerrieri, and A. Liotta, Eds., Springer International Publishing, pp. 191–202.

[14] Riet, J. v., Paganelli, F., and Malavolta, I. From 6.2 to 0.15 seconds – an industrial case study on mobile web performance. In *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)* (2020), pp. 746–755.

[15] Rojas-Mora, J., Lincolao-Venegas, I., and Schneeberger-Leon, F. S3e2: a web-based gis for the visualization and analysis of socioeconomic segregation in chile's elementary education system. In *Proceedings of the 1st International Conference on Geospatial Information Sciences* (2019), O. S. Siordia, J. L. S. Cardenas, A. Molina-Villegas, G. Hernandez, P. Lopez-Ramirez, R. Tapia-McClung, K. G. Zuccolotto, and M. C. Colunga, Eds., vol. 13 of *Kalpa Publications in Computing*, EasyChair, pp. 12–20.

[16] Shivakumar, S. K. *Modern Web Performance Optimization*. Apress, Berkeley, Ca, 2020.

[17] Szalek, K., and Borzemski, L. Conversion Rate Gain with Web Performance Optimization. A Case Study. In *Information Systems Architecture and Technology: Proceedings of 39th International Conference on Information Systems Architecture and Technology – ISAT 2018* (Cham, 2019), Springer, pp. 312–323.

[18] Yu, A., and Benson, T. A. Dissecting performance of production quic. In *Proceedings of the Web Conference 2021* (New York, 2021), WWW '21, ACM, pp. 1157–1168.