Critical Rendering Path Optimizations to Reduce the Web Page Loading Time

Pratiksha H. Shroff
Research Scholar
Department of Computer Science and Engineering
Marathwada Institute of Technology,
Aurangabad, India.
pratikshashroff@gmail.com

Seema R. Chaudhary
Department of Computer Science and Engineering
Marathwada Institute of Technology,
Aurangabad, India.
seema.chaudhary@mit.asia

***Abstract*—In recent times, the websites have become complex. They are made up of a lot of content like videos, images, Javascript, Cascading Style Sheets(CSS).Because of too much content, the website gets heavy and takes more time to load. On the other hand, users prefer faster loading websites. Slower websites means unhappy users, who in turn abandon the sites in preference to faster sites. To overcome this, the websites should be optimized. Research shows that the complexity of websites is characterized at two levels – content and service. The focus of this work is to optimize the content level complexity. It is observed that reduction in critical rendering path length of a page results in less page load times. Critical rendering path is the set of processing steps a browser goes through for completely rendering a page. So, the idea of the project is to optimize the parameters impacting the critical rendering path length. The parameters under consideration are images, CSS and Javascript. These optimizations are tested on a website having images and text both in proportion.**

***Index Terms*—Google, optimization, page, time, website.**

I. Introduction

Nowadays, websites have become complex. Different types of websites cater to a range of present day needs like blogs, e-commerce sites, photo galleries, videos etc. This has become possible due to several different types of content supported by the websites. In turn rendering a single web page has become complicated due to the different content types in use. Due to this, the website becomes slow and loading of a web page takes more time. Surveys like [1],[2] provide the statistics of the impact of a slow loading website on users.Lower satisfaction means the slow-loading pages aren't just impacting one customer visit page load time but it can prevent customers from wanting to return to the site or recommend it to their friends [1].In addition to the relation of a website loading time to the user retention, Google is considering the page loading time in their search rankings. Slower websites get less rankings. Page speed score is a measure assigned by Google to every website, it is negatively correlated to the Page loading time, i.e. if the page loading time is more, the Page speed score will be less. And less Page speed score attributes towards less page rank[3].All the prior mentioned factors raise a need for the study and optimization of Web pages. Research shows that the Complexityof Web pages can be characterized and divided into two main categories- Content level and Service level complexity[4].Content level complexity deals with the number and types of objects required to load the content on a Web page. And Service level complexity includes the number of multiple servers involved in loading of a site and the number of non-origin services like advertisements, analytics services etc. However, the impact of non-origin contenton the download time is low[4].This project has two main focus areas.

A. Compare and Analyze application

The first focus is to create an application to compare the Page speed scores of two websites. And this application will give detailed suggestions for optimizations of a website.It acts as a supporting functionality to the optimization process. For an existing website, it can provide suggestions, based on which the website at hand can be optimized.The change in the loading performance i.e, the page speed score of the optimized website can be once again checked in the same application.

B. Optimization Process

Page loading time is the time required to download and display the entire content of a web page in the browser window. Displaying the entire content on a page is also called as rendering of a page. Rendering a single web page today involves fetching several objects like images, Javascripts, CSS with varying characteristics resulting in heavy Web pages and in turn more loading times. So,the second focus of this project is to create a web site, apply the optimizations parameter wise and record the impact of optimizations on the page loading time.

Critical Rendering Path is the steps that the browser goes through to convert HTML(Hyper Text Markup Language), CSS, Javascripts into the pixels of the screen[5].In [5],Google suggestsvarious optimizations parameters for improving the loading times. In this study, each parameter is optimized and its impact on the page load time is recorded. These optimizations are applied on a website having images and text in balanced proportion. In addition to the page loading time, page speed score of the websites before and after optimization can be measured using the Compare and Analyze application created in the first part of the project.

II. RELATED WORK

A. Historical Background

The past work involved the study of Web graphs, website popularity, and the nature of Web traffic. The web has kept evolving exponentially since many years, the graph of web has been an interesting topic of study, it has several hundred millions of nodes, over a billion of links and keeps growing[6].Further studies lead to the observation that the macroscopic structure of the web is in fact more intricate[7].With the evolving web, the websites are having rich applications involving heavy client side interaction and hence nature of the web traffic changes as well[8] and it impacts the response times of websites. In [9], authors have observed that one of the most often-discussed complaints about the Web experience is the delay users frequently encounter while browsing. These delays impact the business, customer satisfaction, Page speed scores etc. So, it is important to understand the impact of page loading time on users. The results of the experiments in [9] indicate that users prefer websites with short delays over sites with long delays. It provides valuable insights on web users tolerable wait time and that performance and behavioral intentions began to stabilize at 4-second delay. And another such research reviews the literature on computer response time and users, it suggests that tolerable waiting time for information retrieval is approximately 2 seconds[10].

B.Characterizing the web page complexity

As detailed before, all the previous work involved the study of World Wide Web, Web graphs, Web traffic etc., but study of a Web page itself was not done. In [4], authors have made the first attempt towards characterizing the complexity of a Web page and it is most relevant and useful with respect to this project. Researchers had focused on the client side processing on the Web pages and not on the back-end server infrastructure. They characterized the complexity of Web page using its content like the number of objects required, the size of the objects and the types of content[4]. As per their study, the impact of the non-origin content like the analytics, advertisements was low on the download time. Another important finding was that the number of objects fetched has more impact on the download time rather than the number of bytes requested. And Page load times are dominated by the number of round-trip times(RTTs) rather than the number of bytes requested.

The typical load time of a website is strongly dependent on the number of objects requested whereas the variability in the load times is attributed to the number of servers.

Considering all the above findings, it is important to identify a small set of features which can be used for optimized website designing.

C. Critical Rendering Path Optimization:

Google assigns a Page speed score to every website based on how it is performing with respect to the Page loading time[5]. Thus, a higher score is assigned to a website which loads faster i.e. in less time. Study reveals that there is strong negative correlation between the page loading time and the Page Speed score[3].And Google uses this Page speed score as one of the signals in the website rankings.

Now, for the purpose of this project, the focus is narrowed down to the content level complexity. So, the area of interest is the number of objects fetched and RTTs involved in rendering a page. This area has been further explored by Google in their Critical Rendering Path Optimization section of developer tools. Critical Rendering Path is the steps a browser goes through before rendering a page. And Critical Rendering Path length is the number of round trips required to fetch all the critical resources. Hence, the reduction in the number of round trips for critical resources is proposed. The proposed system in the next section, takes into account all the parameters impacting the Critical rendering path length. These parameters are then optimized and applied on the website under study. The resulting optimized site gets loaded faster and have better page speed scores.

III. THE PROPOSED SYSTEM

The Proposed system comprises of two parts:

A. Online application for Comparison and Analysis of websites

Fig. 1 and Fig. 3 together form the Compare and Analyze application.

Compare functionality compares two input URLs and provides the output as their page speed scores and detailed Web page related statistics.

Analyze functionality provides the optimization suggestions for an input URL(Uniform Resource Locator). This functionality gives the details as what parameters can be worked upon and optimized to decrease the page loading time. This application makes use of the pagespeedonline Application Programming Interface (API).

B. Implementing optimizations

Fig. 2 depicts the process of optimizations. It provides a high level overview of parameter names and how each parameter is optimized. As shown in the Fig. 2, we have considered nine types of optimizations in this work.

1) Compress images is to compress the images available in different formats. This compression results in less bytes to be transferred when a Web page is requested. It is done using imagemin API.
2) Combine external Javascript: Too many external files result in multiple RTTs and hence more loading time. So, we have combined external Javascript files into one file using a Java program. And multiple round trips are reduced to just one.

3) Combine external CSSs: External multiple CSS files are combined into one and hence the number of round trips to fetch the CSS files gets reduced.
4) Minify Javascript: Minification is the process of removing all the unnecessary characters from the source code without changing its functionality [11].These unnecessary characters usually include white space characters, new line characters, comments, and sometimes block delimiters, which were used to add readability to the code but are not required for it to execute[11].This results in compressed file and hence less bytes are transferred for a page request. This is done using a Java program.
5) Minify CSS: Similar to Javascript files, CSS files can be minified.
6) Minify HTML: Similar to Javascript and CSS files, HTML can also be minified.
7) Inline small Javascript: It is advised that small Javascripts be inlined. This is done by removing the external call to Javascript file and by adding the code inline into the HTML inside the script tag.
8) Inline small CSS: Small CSS files can be inlined into the code and thus preventing an external call.
9) Asynchronous processing of Javascript: By making the Javascript asynchronous, it can be fetched in parallel to the parsing of the HTML content. This results in reduction of the Critical rendering path length. In effect, the web page can load faster.

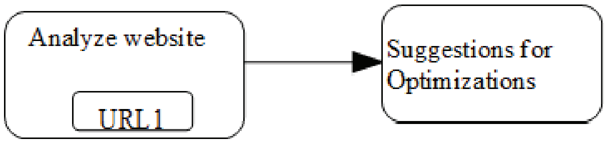

Fig. 1. Analyze screen

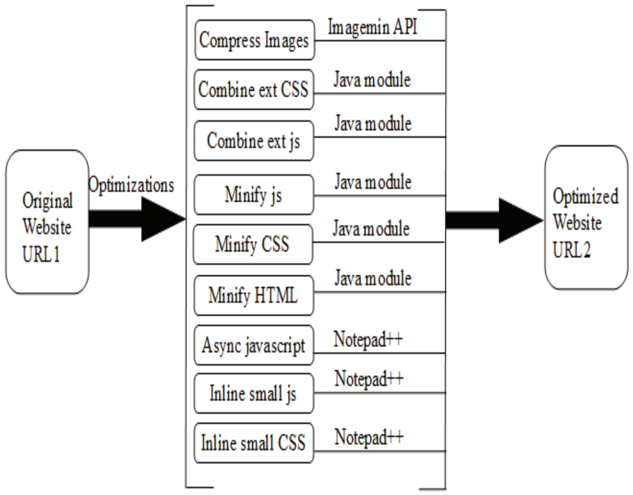

Fig. 2. The Optimization Process

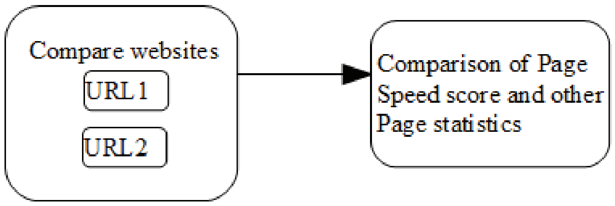

Fig. 3. The Compare screen

IV. EXPERIMENTAL SETUP

The project is implemented on a system having 2 GHz processor speed, 6 GB RAM, 64 bit Operating System, 100 GB Hard Disk and 12 Mbps internet speed.

A website is created with lot of images and its related text, the images and text are in balanced proportion. Each parameter optimization is applied one by one, then in combination and finally all of them together. Performance of the website is evaluated for all the scenarios mentioned in TABLE I. Google chrome browser is used in the in-cognito mode for the Performance evaluation. Before each run of the evaluation, it is made sure that all the instances of the in-cognito browsers are closed. Otherwise, the browser will retain the in-memory cache and the performance will not be effective. For each run, the Network tab inside the developer tools window is opened. Load time is displayed in red inside the Network tab. Thus, for each optimized parameter, several rounds of evaluation are done and hence multiple readings are noted for the original and the optimized site. The average percentage decrease in load time is calculated for every parameter. The process is repeated for all the parameters and their different combinations. The results are in the TABLE I.

TABLE I

PERFORMANCE EVALUATION RESULTS FOR IMAGE AND TEXT WEBSITE

Sr. no.	Scenarios for Page loading time			
	Scenario title	Avg. load time before Optimization in ms	Avg. load time after Optimization in ms	Percentage Decrease in load time
1	Compress images	385	338.75	12.20
2	Combine external Javascripts	395.5	319.5	11.12
3	Combine external CSS	393.5	330.25	16.07
4	Minify Javascripts	387.33	380	1.89
5	Minify CSS	403.67	400	0.009
6	Minify HTML	431.5	400.5	7.18

Sr. no.	Scenarios for Page loading time			
	Scenario title	Avg. load time before Optimization in ms	Avg. load time after Optimization in ms	Percentage Decrease in load time
7	Inline small CSS	389.75	367	5.83
8	Inline small Javascript	386.25	343.25	9.95
9	Asynchronous Javascript	381	336	11.81
10	Compress images and combine Javascripts	440	361.25	17.89
11	Compress images and combine CSS	388.75	359.25	7.58
12	Optimize images, Javascripts and HTML	388.75	334.25	14.019
13	Optimize all Parameters	362.2	273.2	24.57

V. Conclusion

It is observed that the page loading performance is significantly improved after applying the various optimizations. For the website under consideration with images and text in balanced proportion, an overall 24.57% reduction in load time is recorded when all the optimizations are applied.When hosted online, the Page speed score for the optimized site increased from 57 to 87. It is observed that the compression of images, combining external JavaScript, CSS and asynchronous processing of Javascript has strong impact (10%+) on reducing the Web page loading time.

References

[1] http://www.prnewswire.com/news-releases/customers-are-won-or-lost-in-one-second-finds-new-aberdeen-report-65399152.html

[2] https://blog.kissmetrics.com/loading-time/

[3] http://www.seochat.com/c/a/search-engine-optimization-help/google-page-speedscore-vs-website-loading-time/

[4] Michael Butkiewicz, Harsha V. Madhyastha, and Vyas Sekar, "Characterizing Web Page Complexity and Its Impact",IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 22, NO. 3, JUNE 2014

[5] https://developers.google.com/web/fundamentals/performance/critical-rendering-path/

[6] J. M. Kleinberg, S. R. Kumar, P. Raghavan, S. Rajagopalan, and A.Tomkins, "The Web as a graph: Measurements, models and methods,"in Proc. COCOON, 1999, pp. 1–17.

[7] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan,R. Stata, A. Tomkins, and J. Wiener, "Graph structure in the Web,"Comput. Netw., vol. 33, no. 1, pp. 309–320, Jun. 2000.

[8] S. Ihm and V. S. Pai, "Towards understanding modern Web traffic," in Proc. IMC, 2011,pp. 295–312.

[9] D.Galletta, R. Henry, S. McCoy, and P. Polak, "Web site delays: How tolerant are users?," J. Assoc. Inf. Syst., vol. 5, no. 1, pp. 1–28, 2004.

[10] F. Nah, "A study on tolerable waiting time: How long are Web users willing to wait?," Behav. Inf. Technol., vol. 23, no. 3, pp. 153–163,May 2004.

[11] https://en.wikipedia.org/wiki/Minification_(programming)