

# Website Design Ranker

Using Machine Learning

Adhyaksh Guhan - 7 , Anet Eliza Johny - 23 , Dharwish Raj - 47 ,  
Joel J Padayattil - 60

Department of Computer Science and Engineering  
**FISAT**

16 SEPTEMBER 2019

# Introduction

- Our project is aimed at ranking websites in terms of its design which is evaluated based on certain parameters.
- Since a perfect model for website ranking is not in practice this follows ranking according to submissions by critics.
- Thus we compare ranking implementation done by Google that analyses a webpage's content.
- We will be comparing against:
- A Website that ranks other design according to submissions by critics
- A ranking implementation done by Google that analyses a web-page's contents

# Problem Statement

- Our main problem is to evaluate website designs using an algorithm that uses machine learning
- It must take into account various parts of the website to use as parameters.
- for example: people won't be interested to visit a site if they are bothered with ads. Thus ads above a degree will be considered as the parameter for ranking.
- There are no existing algorithms or methods that manage to analyze and evaluate websites in this way.

# Why "Website Design Ranker" ?

- Website Design Ranker will rank set of input websites based on certain parameters.
- It will be helpful to find best website among list of websites which have same content.
- We can compare our website design with other competing websites.
- We can see how a website's design may improve in an area.

# Existing Methods : Website Design Ranking Agencies

- Websites are ranked by Ranking Agencies as per submission on their database.
- Hired critics and analyzing staffs are reviewed and ranked according to their policy.

Example:

<https://www.awwwards.com>

<https://www.cssdesignawards.com>

<https://www.csswinner.com/winners>

<https://thefwa.com>

# Existing Methods : Google Page Layout

- Google introduced Page Layout Algorithm to analyze website readability.[1]
- Looks for the layout of the web page and the amount of content we see in the page once we click on a result.[1]
- Focuses to reduce the difficulty of users to find the actual content.[1]
- The websites which does not have a lot of visible content above-the-fold and dedicates a large fraction (above a normal degree) to ads will be affected.[1]

# Algorithm

- 1 Determine the old design of the website.
- 2 If a website has many advertisements above the fold(the part of page that is visible on the screen when the page first loads before scrolling)then it is considered as the first drawback.
- 3 Else if a website has a large Flash animations or other non-content elements that forces users to scroll to see the content, then that will be the next drawback.
- 4 Else no drawback.
- 5 These drawbacks will affect the ranking process.
- 6 If these conditions are met the website ranks a low rank.
- 7 Else its rank will be considerably high.

# Existing Methods : Google Page Layout

Good example: site layouts that highlight content

Bad example: site layout that pushes content below the fold



Figure: One of the criteria of GPL Algorithm<sup>[Fig:1]</sup>



# Comparison

- **Existing Methods : Website Design Ranking Agencies**
- Website design ranking agencies are not capable of analyzing particular website with different other websites.
- Ranking Varies from each critic or staff, cant be used in large scale, time consuming.
- **Existing Methods : Google Page Layout Algorithm**
- Only looking for layout of the page, more importance for SEO.
- Proprietary code of Google

# Problem Analysis

- Our main effort was to create an algorithm for ranking websites based on the parameters. Even a simple looking websites with exceptional usability and well-structured will cope up with the expectation of the users.
- This will allow users to deviate from the existing manual ranking system of websites design.

# What we proposed?

- We propose a system where an algorithm scrubs through a website, looking for various elements.
- Once we discover the nature of these elements, we check whether the parameters we have set (eg: colour, symmetry, etc) have been met.
- For each parameter met, a website will obtain a mark.
- Once all parameters have been checked, the website receives an overall score (the sum of all marks) that ranks its design.

# Algorithm

- 1 Start
- 2 Using a website scraper to accept the various website addresses
- 3 Scraping through the source code of each website via CSS files
- 4 Find the hex codes of all elements of the website and count them with a count variable
- 5 If count == 0
  - (4.1) Give mark as 0
- 6 else if count > 5
  - (5.1) Give mark as 0
- 7 else if count <= 5
  - (6.1) Give mark as 1
- 8 Stop

# Conclusion

- Here we can see the logical differences in the approaches that our algorithm takes versus any existing methods.
- Our method relies on an objective and automated method that is consistent in nature as opposed to the subjective methods of the existing methods.

# References

1 - Google Page Layout Algorithm: Everything You Need to Know "<https://www.searchenginejournal.com/google-algorithm-history/page-layout/close>"

Fig:1 - <https://cdn.searchenginejournal.com/wp-content/uploads/2017/10/google-algorithm-above-the-fold-380x238.png>