CSC111 Project Report:
Linking to Success: How Backlinks Affect SEO Performance of Websites

John DiMatteo, Kyrylo Degtiarenko, Samuel Joseph Iacobazzi, Arkhyp Boryshkevych

Sunday, March 30, 2025

Introduction

One of the challenges that website owners face is ranking high in search results. There are two main ways to raise website recognition: through paid ads and through optimization of the website for search engines. The second approach is called SEO - Search Engine Optimization. Platforms like Google Analytics can provide general statistics and advice on how to improve SEO performance (*Google Search Central (2025). Search Engine Optimization (SEO) Starter Guide*). There are many techniques, including picking the right keywords, optimizing metadata, adding site maps, using appropriate headers (*Michigan Tech University (2025). Five Ways to Improve Your Site's Ranking (SEO)*). This project will be focusing on SEO through external links to and from the website.

Outbound links are hyperlinks that go from a particular website to an external website. Backlinks are links that go from an external website to the current website.

There is some evidence that proper linking with other websites can enhance the SEO performance (*Michigan Tech University (2025). Five Ways to Improve Your Site's Ranking (SEO)*), however, it is not explored in detail how effective it is or what factors to consider when adding links to other websites. Better understanding of how hyperlinks affect search engine ranking can provide website owners with useful information on how to increase natural traffic to the website. On the other hand, it can also detect if there are possible exploits of hyperlinking that are present on the web nowadays. Our project also aims to provide insight into how the engagement on the different websites that your website has an outbound or backlink to can affect engagement on your own website. Since there is no transparent official documentation on how particular search engines like Google exactly work, obtaining empirical evidence on how links affect the performance of the website can be of high importance for small software companies, startups, and individuals who wants their website to be noticed by public, but are not interested in increasing their traffic via artificial methods such as paid ads.

The research question of this project is: "**Can we predict the engagement and SEO performance of the website based on the backlinks and outbound links of that website?** "

Datasets

1. Top 50 Websites Dataset (Primary Metrics)

Source: `https://www.kaggle.com/datasets/yamaerenay/top50websites`
Format: CSV file containing website engagement metrics
Columns Used:

- `domain_name`: Website domain identifier
- `daily_min`: Average daily minutes per visitor
- `daily_pageviews`: Average daily pageviews per visitor
- `traffic_ratio`: Search engine traffic proportion
- `site_links`: Number of inbound links

Usage: All columns were incorporated into the vertex statistics for engagement calculations. The `domain_name` field serves as the primary key for the identification of graph nodes.

2. Common Crawl Hyperlink Graph (Web Structure)

Source: `https://data.commoncrawl.org/projects/hyperlinkgraph/cc-main-2021-feb-apr-may/`
Format:

- `vertices.txt`: ID-domain mappings (`ID, [domain]`)
- `edges.txt`: Directed edge list (`sourceID, destinationID`)

Usage:

- Vertex IDs filtered to match domains present in the metrics dataset
- Edge connections validated against available vertices (only added if both endpoints have associated metrics)

3. Tranco Ranking (Supplementary Traffic Data)

Source: `https://tranco-list.eu/`
Format: CSV ranking data
Tranco Rank-domain mappings (`Rank, Domain Name`)
Integration: The data set for primary metrics was merged as column through domain matching. `tranco_rank`: Used for cross-verification with Alexa rankings.

Computational Overview

0.1 1. Loading/Cleaning Data

First, we created a smaller sample from our three datasets. From 80 million vertices present in Common Crawl dataset we filtered out those that are present in Tranco dataset. Then we filtered out edges from Common Crawl dataset that contain vertices that were selected in the previous step. Form this set we selected approximately 10000 vertices, along with their edges. For this part we selected websites that where first 10000 in Tranco dataset, to ensure there are enough well known websites. Then we removed any domain names in Top 50 Websites that did not belong to the set vertices we filtered earlier
The project allows to either load the entire data of vertices/edges by setting **load_with_stats_only** argument of the loader function to be False, or only load vertices that has additional stats for them from Top 50 Websites dataset with argument set to True. It is also possible to adjust number of vertices you want to load with **n_vertices** argument
Loading is done by **data_loader.py** class, using csv and txt readers

0.2 2. Computing Statistics (stats.py)

The `stats.py` module implements a comprehensive framework for analyzing website engagement metrics through graph-based computations. The system processes four attributes of the core dataset and derives six additional metrics, with global averages calculated for comparative analysis.

0.2.1 Dataset Metrics

- **daily_min**: Average daily minutes spent on the site per visitor
- **daily_pageviews**: Average daily pages viewed per visitor
- **traffic_ratio**: Proportion of traffic originating from search engines
- **site_links**: Number of external sites linking to this domain

0.2.2 Computed Metrics

1. **min_per_page** ($\frac{\text{daily\_min}}{\text{daily\_pageviews}}$)
   Measures average engagement depth per page view, balancing time spent against content volume.

2. **search_traffic** (traffic_ratio × 100)
   Represents the percentage of traffic originating from search engines, directly converted from the traffic ratio metric. The values range from 0-100%, where higher values indicate a stronger presence of organic search.

3. **links_traffic** ($\frac{\text{site\_links}}{\text{site\_links}+\text{daily\_pageviews}} \times 100$)
   Measures the percentage contribution of inbound links to traffic. Uses a normalized ratio of backlinks to pageviews (diminishing returns model), where values closer to 100% indicate a stronger influence of referral traffic from linked websites. Inspired by the Google Analytics formula for engagement.

4. **engagement_rating**

   $$\sqrt{\text{daily\_min} \times \text{daily\_pageviews}} \times \sqrt{\frac{\text{min\_per\_page} + \text{search\_traffic}}{2}} \times \sqrt{\ln(\text{links\_traffic} + 1)}$$

   This composite index combines:

   - *Overall Activity* (total user-minutes)
   - *Quality Factor* (average of per-page engagement and search validation)
   - *Network Influence* (logarithmic link evaluation)

   Square roots are used to balance the weighting between components, preventing any factor from dominating the score.

5. **predicted_rank**
   Global engagement-based ranking (0-indexed) calculated through descending sort of engagement ratings.

0.2.3 Global Metrics

Each computed metric has a corresponding global average (e.g., `global_daily_min`), calculated by averaging vertex values across the entire graph. These serve as benchmarks for relative performance analysis.

0.3 3. Loading Statistics to Graph

0.3.1 `percentify` Function

Converts absolute metric values into percentage differences from global averages:

$$\text{percentage} = \frac{\text{value} - \text{global\_avg}}{\text{global\_avg}} \times 100$$

This transformation produces intuitive performance comparisons (e.g., +38% above average engagement).

0.3.2 `loader` Function

Automates metric computation across the entire graph using dynamically callable keys for general data calculations:

- Identifies target metric via function name introspection
- Special-cases `predict_rank` for global context requirements
- Updates vertex `stats` dictionaries *in-place* for persistent storage

Program-Use Instructions

1. **Prerequisites**:
   - Python installed.
   - Project ZIP file downloaded from MarkUs:
2. **Setup**:
   (a) Extract project ZIP contents into your directory
   (b) Install dependencies:

   ```
   pip install -r requirements.txt
   ```

3. **Execution**:

   ```
   python main.py
   ```

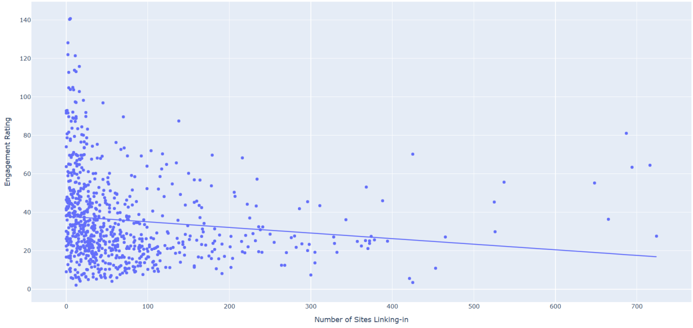
4. **Tips**: For better performance you can increase/decrease **n_vertices** when loading
5. **Interactive Features**:
   - Users are able to use the Tkinter control panel pop-up window to add a website URL to the graph. Our program searches other websites for links to the user's input, and scrapes the input website for links before including this information in the graph for the user to visualize.
   - Users may hover their cursor over individual nodes on the generated plot graph to view the website name and statistics of each domain in the dataset. The stats are also compared to global statistics.

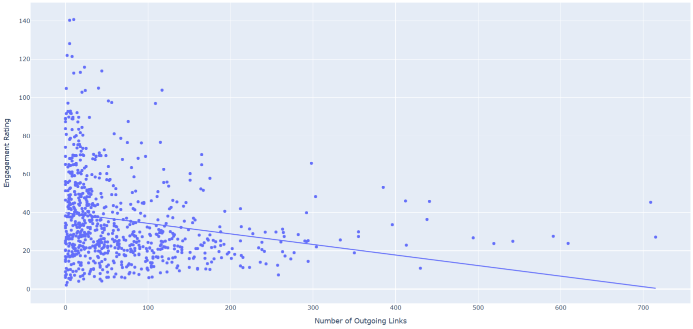
Changes to Original Project Plan

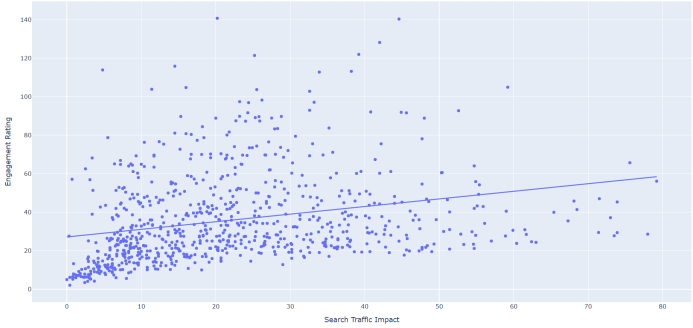
- Changed the datasets for SEO ranking. Since SEO is based on category, we decided to use traffic to estimate performance. For that one we used the Tranco dataset since it has data combined from different search engines
- For visualization of different connected islands, we decided to use community detection in graph instead of using strongly connected components. Now we compute statistics for every community separately instead of only doing that biggest and smallest one, and let users plot that. That way there is no ambiguity and we are users are still able to compare statistics between communities. We display size of each community using a color scale.
- We changed the parameters we want to calculate for each node. Previously, we focused mostly on just stats provided by the graph structure itself. Now we also compute statistics like minutes per page, search traffic, links traffic, based on Top 50 websites dataset, which provides a more complete picture to the user.
- For visualization we changed from having graphs embedded in tkinter to using tkinter as control panel and plotting graphs in plotly. This decision was done after additional research into the compatibility of two libraries. This approach allows for interactivity with buttons on the control panel as well as interactivity inside the graph itself.
- Also for visualization, we added the ability to plot graphs that reflect relations between some statistical parameters.

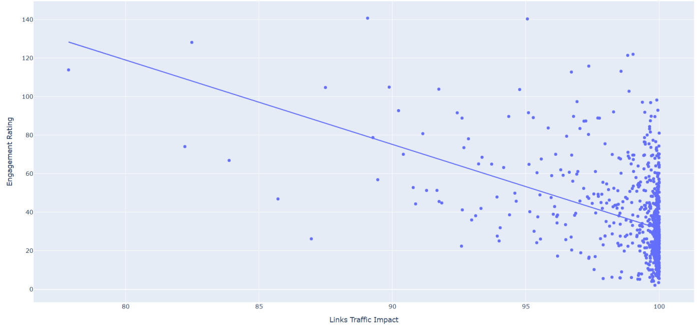
Conclusion

Originally, we hypothesized that specific SEO indicators, especially number of outgoing and incoming links to/from a website, would have a positive correlation with user engagement. However, based on the trend lines on the plots generated by our program, there actually seems to be a *negative* correlation between the number of incoming links and user engagement, and an even stronger negative correlation between the engagement rating and outgoing links on the webpage. However, the "Search Traffic Impact" figure has a positive correlation with engagement rating. We've drawn several. conclusions from this. First, a webpage with many outgoing links is unlikely to keep a user engaged for very long, as they seem to end up leaving to one of the many other sites linked on the page. Additionally, we theorize that more 'popular' websites with many other sites linking in tend to have a very broad target audience and are less 'personalized' compared to lesser-known, niche websites. This could be the reason that more popular websites seem to keep users engaged for less time. Finally, the way users come across a website has a considerable impact on user engagement. Based on the search traffic impact stat, users who actively seek out a website through a search engine are more likely to remain engaged than those who come across the website by a link from another webpage.









References

Arvidsson, Joakim (2023). Top 10 Million Websites 2023. *Kaggle.* https://www.kaggle.com/datasets/joebeachcapital/top-10-million-websites-2023

Bpali26 (2017). Popular websites across the globe. *Kaggle.* https://www.kaggle.com/datasets/bpali26/popular-websites-across-the-globe

Common Crawl (2025). Web Graph Statistics. https://commoncrawl.github.io/cc-webgraph-statistics/

Eren Ay, Yamac (2021). Top 50 Most Popular Websites by Countries. *Kaggle.* https://www.kaggle.com/datasets/yamaerenay/top50websites?select=sites.csv

Goharfar, Ashkan (2020). Effective SEO parameters for all types of websites. *Kaggle.* https://www.kaggle.com/datasets/ashkangoharfar/sites-information-data-from-alexacom-dataset

Google Search Central (2025). Search Engine Optimization (SEO) Starter Guide. https://developers.google.com/search/docs/fundamentals/seo-starter-guide

Kamal Alsyd, Mohammed (2024). SimilarWeb Top Websites [April 2024]. *Kaggle.* https://www.kaggle.com/datasets/mohammedkamalalsyd/similarweb-top-websites

Michigan Tech University (2025). Five Ways to Improve Your Site's Ranking (SEO). *University Marketing and Communications.* https://www.mtu.edu/umc/services/websites/seo/