Automated SEO

A Market Brew White Paper

Abstract

In this paper, we use the term “Reach” to suggest the forecasted traffic to a particular webpage or website. Reach is a traffic metric that describes an expected traffic value *before it actually occurs*.

Using Market Brew, steps can be taken to automatically determine the optimal set of changes to any website that will give us the maximum amount of reach increase with the fewest amount of changes.

In order to determine these steps, we must:

1. Determine rankings of each search term, including the overall query score and all individual scores that go into the overall query score.
2. Simulate all possible ranking changes for every webpage / keyword combination.
3. Order simulations by most efficient (i.e. least amount of changes for most gain in reach).
4. Recommend best set of optimizations, based on user preferences.

Step 1: Determine Rankings

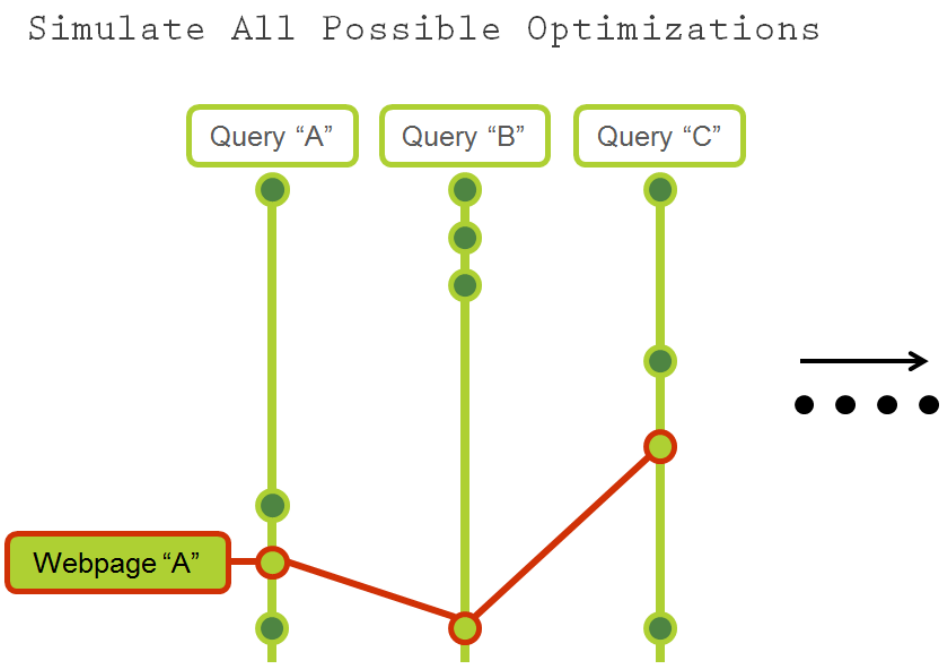
Market Brew’s search engine results are unique. Not only do users see the order of the search results for a particular query, but they also see the distances between those results, and how those distances were calculated.

Search Results with Query Scores (Distances)

Knowing the distances between search results, in terms of query score, is an important step towards automating SEO. We can now simulate all possible moves with great precision.

Step 2: Simulate All Possible Ranking Changes

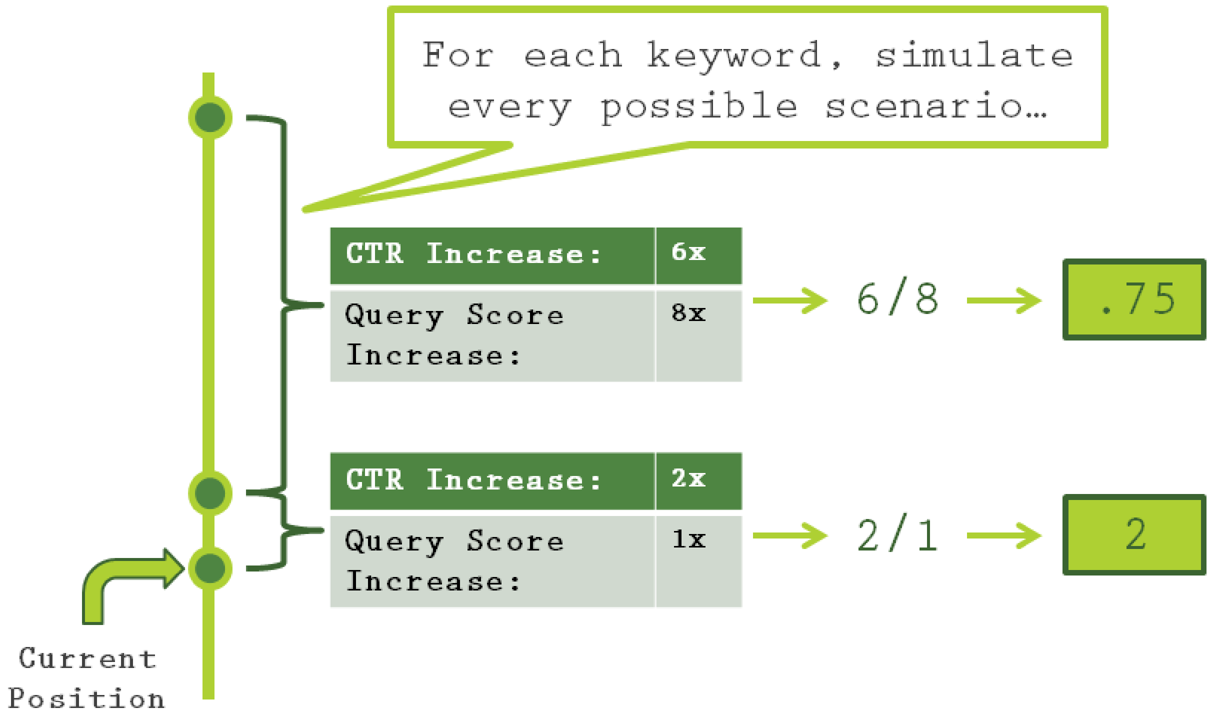
Once Market Brew has established a precise ranking schema, it can begin to simulate, for each webpage and each query, all the possible ranking changes that could occur.



For each webpage, for each query, simulate ranking changes

Within each simulation, we first determine how much efficiency there is within each ranking change. To determine efficiency, we must create a metric, for each simulated ranking change, that allows us to see how much a particular webpage stands to gain in terms of reach vs. how much it must increase its query score (i.e. cost) to gain that additional reach.

Reach Potential (RP)



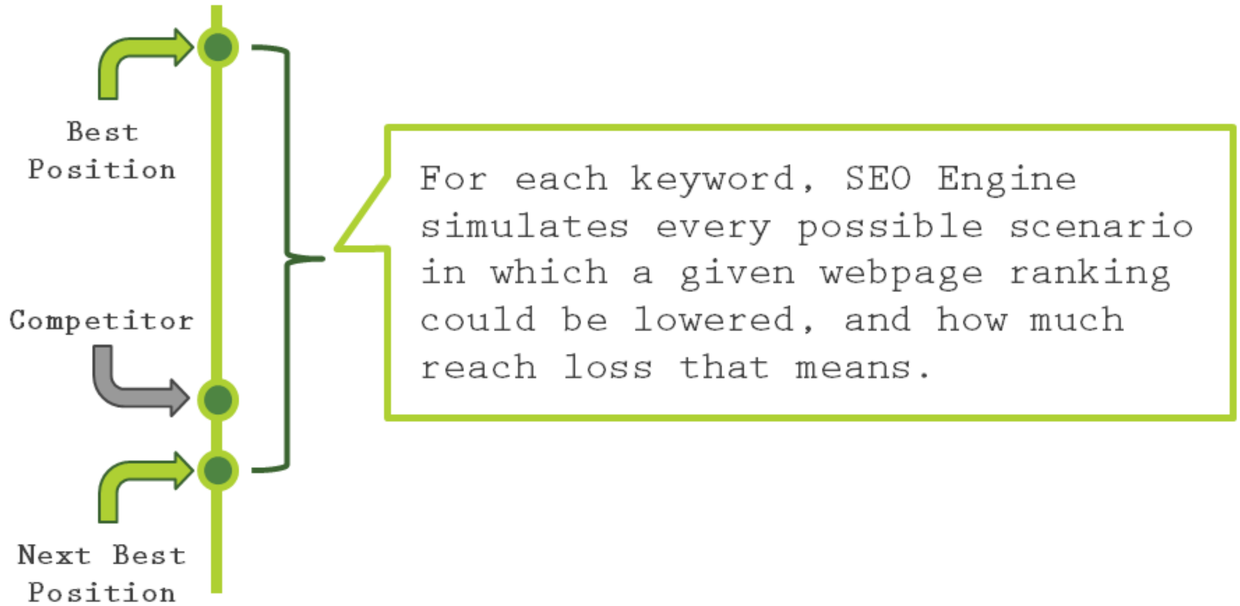
Create an efficiency metric called "Reach Potential"

Take the estimated search volume K for a given query and the click-through rate CTR_w for position occupied by webpage w, and the query score QS_w for current webpage w, the Reach Potential (RP) can be represented by:

$$RP(x,t) = K * \frac{CTR_t - CTR_x}{QS_t - QS_x}$$

We must also take into account any negative side effects of this ranking change on the overall traffic to a website. To do this, we must create a metric called "Potential Reach Loss".

Potential Reach Loss (PRL)



Take the estimated search volume K for a given query and the click-through rate CTR w for position occupied by webpage w, we remove the webpage w occupying current position x and now rely on the next best webpage in the website to rank. Given x as the current position and t as the new (lower) position, the Potential Reach Loss (PRL) can be represented by:

$$PRL(x,t) = K * (CTR_x - CTR_t)$$

Expected Reach Potential (ERP)

Once we have calculated the Reach Potential (RP) and Potential Reach Loss (PRL), we can combine these two metrics and all of the simulations, for a specific query, into a statistical expected value.

Take the estimated search volume K for a given query, the reach potential RP and potential reach loss PRL of the webpage we are simulating, and the total number of simulations S off all possible ranking changes, then the Expected Reach Potential (ERP) can be described as such:

$$ERP = \sum_{i=1}^{S} \left(K * \frac{1}{S} * \frac{RP}{PRL} \right) = \sum_{i=1}^{S} \left(\frac{K * RP}{S * PRL} \right)$$

This gives us a metric that we can then order by, giving us the most efficient and least risky ranking changes for a given website.

Step 3: Order Simulations By Most Efficient

Next, we simply order all possible ranking simulations by Expected Reach Potential (ERP).

Expected Reach Potential	Simulated Query	Webpage
3,701.37	law	/lawyer-website-design-portfol...ia/law-firm-tv-shows
3,637.55	law	/lawyer-website-design-portfol...online-video-a-media
3,563.19	law	/sitemap
3,550.09	law	/case-studies
1,793.96	search engine optimization	/sitemap
1,793.96	search engine optimization	/blog/category/case-studies/se...engine-optimization/
1,782.07	search engine optimization	/web_site_design.html
1,735.34	county bar association	/lawyer-web-design/index.html
1,735.34	county bar association	/best-law-firm-websites/best-l...m-web-marketing.html
1,722.52	county bar association	/sitemap

Each row represents a potential ranking change, due to a set of optimizations that can be applied to that webpage. Some example optimizations, which would be directed by the specific situation, would be to change the content or link structure around that webpage.

Step 4: Recommend Best Set of Optimizations

Once Market Brew has identified and ordered all possible ranking simulations, it can begin to simulate all possible optimization paths. This is similar to the famous “Traveling Salesman Problem”, where the “cities” are represented by the different optimization simulations, and the “distance” is represented by the Expected Reach Potential (ERP).

Expected Reach Potential	Simulated Query	Webpage
3,701.37	law	/lawyer-website-design-portfol...ia/law-firm-tv-shows
3,637.55	law	/lawyer-website-design-portfol...online-video-a-media
3,563.19	law	/sitemap
3,550.09	law	/case-studies
1,793.96	search engine optimization	/sitemap
1,793.96	search engine optimization	/blog/category/case-studies/se...engine-optimization/
1,782.07	search engine optimization	/web_site_design.html
1,735.34	county bar association	/lawyer-web-design/index.html
1,735.34	county bar association	/best-law-firm-websites/best-l...m-web-marketing.html
1,722.52	county bar association	/sitemap

It is a combination of selected optimizations that, combined together, represent the best possible *set* of optimizations and ranking changes for a given website.

Personalization

These sets of optimizations can be personalized to the user. For instance, we could personalize the relative weighting K of all keywords, as well as the click-through rates CTR_w of specific webpages. Additional weighting can be modeled utilizing conversion rates CVR_w of specific webpages.

$$RP(x,t) = K * CVR_x * \frac{CTR_t - CTR_x}{QS_t - QS_x}$$

APPENDIX

The Travelling Salesman Problem

The Travelling Salesman Problem (TSP) is commonly formulated as an integer linear program. Label the "cities" with the numbers 0, ..., *n* and define:

$$x_{ij} = \begin{cases} 1 & \text{the path goes from city } i \text{ to city } j \\ 0 & \text{otherwise} \end{cases}$$

For *i* = 1, ..., *n*, let u_i be an artificial variables, and finally take c_{ij} to be the distance from city *i* to city *j*. Then TSP can be written as the following integer linear programming problem:

$$\begin{aligned} \min \sum_{i=0}^{n} \sum_{j\neq i, j=0}^{n} c_{ij} x_{ij} & \\ 0 \leq x_{ij} \leq 1 & \quad i, j = 0, \cdots, n \\ x_{ij} \in \mathbf{Z} & \quad i, j = 0, \cdots, n \\ \sum_{i=0, i\neq j}^{n} x_{ij} = 1 & \quad j = 0, \cdots, n \\ \sum_{j=0, j\neq i}^{n} x_{ij} = 1 & \quad i = 0, \cdots, n \\ u_i - u_j + n x_{ij} \leq n - 1 & \quad 1 \leq i \neq j \leq n \end{aligned}$$

The first set of equalities requires that each city be arrived at from exactly one other city, and the second set of equalities requires that from each city there is a departure to exactly one other city. The last constraints enforce that there is only a single tour covering all cities, and not two or more disjointed tours that only collectively cover all cities. To prove this, it is shown below (1) that every feasible solution contains only one closed sequence of cities, and (2) that for every single tour covering all cities, there are values for the dummy variables u_i that satisfy the constraints.

To prove that every feasible solution contains only one closed sequence of cities, it suffices to show that every subtour in a feasible solution passes through city 0 (noting that the equalities ensure there can only be one such tour). For if we sum all the inequalities corresponding to $x_{ij} = 1$ for any subtour of *k* steps not passing through city 0, we obtain:

$$nk \leq (n-1)k,$$

which is a contradiction.

It now must be shown that for every single tour covering all cities, there are values for the dummy variables u_i that satisfy the constraints.

Without loss of generality, define the tour as originating (and ending) at city 0. Choose $u_i = t$ if city *i* is visited in step *t* (*i*, *t* = 1, 2, ..., n). Then

$$u_i - u_j \leq n - 1,$$

since u_i can be no greater than *n* and u_j can be no less than 1; hence the constraints are satisfied whenever $x_{ij} = 0.$ For $x_{ij} = 1$, we have:

$$u_i - u_j + nx_{ij} = (t) - (t+1) + n = n - 1,$$

satisfying the constraint.