

Search Engine Ranking Variables and Algorithms

Sean A. Gollhofer – Publisher, SEMJ.org

ABSTRACT--This paper discusses search engine ranking algorithms and the variables involved, webspam detection, hyperlink analysis, duplicate content issues, and their underlying structure. Additionally, we analyze on-page and off-page variables -- two classes of search engine ranking factors -- and their possible implications for ranking web documents.

Category – Search Engine Optimization

INTRODUCTION

Manipulating text on a webpage was an early form of SEO. The classic document-ranking technique involved viewing the text on a website and determining its value to a search query by using a set of so-called “on-page” parameters. For reasons that will be made obvious, a simple, text-only information retrieval system produces poor search results. In the past, several text-only search engines relied upon on-page ranking factors.

One of the early web crawlers was Wandex, created in 1993 at MIT by Matthew Gray. Webcrawler, released in 1994, is considered the first web crawler to look at the entire text of a web document. When ranking a document, the early companies (and most that followed) focused on what are now called “on-page factors”-- parameters a webpage author can control directly. As you will see, these parameters are of little use in generating relevant search results.

If we were to write a crude ranking algorithm we could create combinations of HTML parameters appearing on a webpage to generate ranking factors. By using on-page HTML parameters, a simple ranking algorithm could generate a list of relevant documents to a given search query. This approach has the built-in assumption that the authors of the webpages we are indexing are honest about the content they are authoring.

An algorithm is simply a set of instructions, usually mathematical, used to calculate a certain parameter and perform some type of data processing. It is the search engine developer’s job to generate a set of highly relevant documents for any search query, using the available parameters on the web. The task is challenging because the available parameters usable by the algorithm are not necessarily the same as the ones web users see when deciding if a webpage is relevant to their search.

“ON-PAGE” VARIABLES

Looking at the available parameters in an HTML document, one can derive a list of *potential* on-page variables for ranking web documents. For example, in the early 1990s, a search engine called Veronica used the index results from a program called Gopher to look at webpage titles and URLs to determine the topic and relevance of a webpage. Because the document’s author can easily manipulate the title of a web document and its URL, a good ranking algorithm would require either more variables or rely on factors a webpage author cannot control directly. Using more variables in a ranking algorithm naturally makes the manipulation of its search results more difficult.

The following on-page variables are ones we *could* use when constructing a basic ranking algorithm. An examination of how an HTML document is constructed will reveal the parameters in list 1, and surely others can be found. In this paper, we will not comment on the exact use of any of these parameters in modern search algorithms. This topic has been discussed extensively in the search marketing community. These parameters should be constructed correctly to indicate the intended content of a web page.

LIST 1: POTENTIAL ON-PAGE VARIABLES (FACTORS)¹

1. Description meta tag
2. A website’s URL.
3. The title of a website.
4. Keyword meta tags.
5. Density of a given keyword on a document.
6. Keywords using H1 and similar tags
7. Keywords in alt text for describing graphics.
8. Proximity of keywords defines how close keywords are in relation to each other.
9. Prominence of keywords defines where the keywords are on the HTML page. For example, a keyword with high prominence would be at the top of an HTML document.
10. Keywords using HTML bold and/or italics.
11. HTML validation. Is the HTML coded according to a standard (typically validated at www.w3.org)
12. Overall size of a page.
13. Total number of pages within the website.
14. Number of outbound links.
15. Use of quotes text keywords.
16. Using underscores on text keywords.
17. The uniqueness of the content on your page relative to the other content on the web.

18. Content “freshness.” When was content last updated?
Has it changed since the last time it was crawled? ²
19. Spelling and grammar.

¹. We are not commenting on the use of these parameters in current algorithms. They should be used to best describe the content on a page. These parameters have been discussed ad-nauseam in the industry.

². This variable is technically not static because it requires time as a variable and is more difficult to track.

A one-dimensional search algorithm might calculate the density of a keyword on a page, and use that keyword density as a measure of relevance. This type of search can quickly lead to text manipulation if the web authors are aware that they need simply to change the keyword density of their web document to indicate to a search engine what their document is about. We could also combine parameters in list 1 with each other and assign weighting factors to any of them. By combining parameters with statistical data on known “trusted pages,” we can potentially do a better job of filtering out webspam.

A finite number of on-page variables are available within a web document, so other parameters must be found to improve the quality of a ranking algorithm. The challenge is in generating relevant search results by examining the underlying structure of a website and its relationship to other websites.

“OFF-PAGE” VARIABLES

PageRank^{*}

Using only the limited set of parameters in list 1, webspam will be difficult to stop because the website optimizer can still control the parameters the search algorithm is using to determine ranking. Brin and Page [1], and also lesser-known researcher Klienber [2] have proposed that the *structure* of Internet hyperlinks is an effective indicator of the relevance and importance of a web document. Ask.com and Teoma use a hyperlink analysis algorithm based on Klienber’s HITS algorithm. Similar to what is found in research papers, the link to (or mention of) another document on the web should be an indication of its importance to web users. Indirectly, hyperlinks are a kind of review or screening of a webpage by a web user.

Google’s success and the concept of pagerank [1] had an enormous effect on many so-called “website optimizers” and the way they constructed web documents. Unfortunately, a lot of abuse occurred as website developers’ understanding of pagerank evolved. From “link farms” to paid links, blog spam, and massive numbers of link exchanges of all kinds, the discovery of pagerank and its use in Google’s ranking algorithm created a wave of pagerank manipulation. The pagerank concept is based on the underlying assumption that the link to another page is an “on-

topic” link and that the link is an unbiased link to a website. Any other intentional manipulation of these assumptions greatly reduces the effectiveness of the pagerank calculation.

Pagerank and a website’s linking structure are still considered the most important measure of a website’s ability to rank on any given search query. In general, and for obvious economic reasons, the exact structure of ranking algorithms is typically kept as a trade secret by major search engines. By studying patents, published papers, and performing experiments, we can better understand the important elements of ranking algorithms. We know that pagerank is an important factor in modern ranking algorithms.

The basic PageRank* formula for a single web page B is shown in Equation 1:

$$R(B) = \delta/n + (1-\delta) \sum_{\text{hyperlink}(A,B)} R(A)/\text{outlinks}(A) \quad (1)$$

*PageRank is a Google trademark.

A and B represent different webpages. In this equation, δ is typically set at 0.85 [2], while “n” is the number of pages collected in the calculation, and the outlinks of A are the total number of hyperlinks going out on page A. The factor δ can be arbitrarily chosen. Equation 1 tells us that the pagerank value of webpage B is determined by the total number of pages collected (the more pages collected, the more a page’s rank value will decrease), and also that each webpage A will reduce its effective pagerank by the amount of the pages it references. This aspect of pagerank is easy to understand because the out-degree of page A is in the denominator, and if this value increases in size, the pagerank of page B decreases. For each webpage, we have to solve this equation to calculate a pagerank. This approach leads to large numbers of equations requiring solutions.

Understanding this equation has led to long discussions within the SEO community on the effectiveness of “pagerank sculpting” by using the nofollow attribute. This concept is heavily debated and has been addressed by Matt Cutts [3] in public many times. The idea behind pagerank sculpting is that one can minimize the passing of pagerank to other pages within a site by using the nofollow attribute on pages under a single URL. In other words, if the pagerank of a homepage needs to be preserved, the nofollow attribute on links coming out of that page needs to be employed. There is no concrete, published evidence that page rank sculpting hurts or helps website rankings. An analysis of the pagerank equation indicates, however, that pagerank sculpting should increase pagerank if implemented correctly.

Arguments can be made that there are plenty of authoritative websites ranking on high-value keywords that do not use pagerank sculpting. It is common to have a healthy fear of “over-optimizing” a webpage. The webpage optimizer should keep in mind that link development structures must look *natural*. One should balance necessity

and risk when considering these types of techniques. Pagerank sculpting is another heavily discussed issue within the SEO community. Major search engines already contain most of the data any optimizer would need to determine the necessity of any optimization technique.

Since the pagerank formula was conceived, there have been other variations and important additions to the idea. Zoltán Garcia-Molina of Stanford University and Jan Pedersen of Yahoo [4] released a paper in 2004 on the “TrustRank” method. TrustRank is an algorithm that can be used to help automatically identify “trusted,” human-reviewed webpages (in other words, a small set of human-selected seed pages). They claim this method can filter out a significant amount of webspam by beginning with a known set of “trusted” authority webpages. For example, the open directory project [5] contains a set of such documents. The authors further claim that with a “seed” of about 200 authority websites, webspam can be significantly reduced by initiating a web crawl from these pages. Thus, the value of obtaining links from known authority sites is increased.

OFF-PAGE FACTORS

Other well-known off-page factors, in addition to pagerank, are also difficult for the webpage optimizer to control. Off-page metrics are more desirable in any ranking algorithm because they allow the search algorithm to determine which pages appear in search queries, rather than by webpage optimizers manipulating webpages.

As can be seen from studying recent patents and published papers, search engines continue to become more sophisticated by factoring time (rate of change) into ranking algorithms. Adding more variables into any ranking algorithm comes at a monetary and efficiency cost. However, given the improved relevance in the search results, end user experience is usually worth the cost.

By examining variables associated with any webpage or URL, we can generate a list of possible “off-page” variables to be used in a ranking algorithm. A possible list of well-known off-page variables follows.

LIST 2: COMMONLY USED VARIABLES: OFF-PAGE AND TIME BASED

1. Number of websites linking back to a website (pagerank).
2. The pagerank of a website (real value not known).
3. The text around the links linking back to your website, and how fast links are accumulated.
4. The number and quality of directories a page is listed in. For example DMOZ or Yahoo.
5. An IP address and its relationship to other IP addresses.
6. How long a URL has been registered.
7. When a registered domain name will expire.
8. When the search engine last cached the URL or content.
9. When the search engine spider last crawled the URL.

10. How many pages of the website were crawled (crawl depth).
11. How fast the pages can be crawled (crawl rate).
12. Pages affecting the priority of a crawl. Where do crawls originate?

Note: Most, but not all, of these metrics are referred to as “off-page” factors because the web author cannot theoretically control them.

One reason for moving to metrics like those shown in List 2 is that they are less obvious to the website optimizer. Major search engines like Google and Yahoo! have a majority of the world’s search queries at their disposal. These search engines also have access to statistical data for how authoritative webpages have evolved over time. Armed with this type of information, search engines can develop algorithms that can detect unnatural webpage behavior.

For example, a search algorithm whose metric is the length of time for which a domain is registered indicates that web spammers do not, on average, register their domains for more than the minimum time required by a registrar. Keep in mind that a search engine does not need any particular website in its search results. If a website falls outside of the natural behavior the algorithm expects to see, the website risks getting penalized or completely dropped from the search results.

Advanced SEO Metrics and Algorithms

Using combinations of the metrics already discussed, algorithms and algorithmic extensions to a search engine’s core algorithm can be developed. By combining the variables of pagerank, time, and many of the metrics discussed previously, new metrics and/or algorithms can be generated to improve the quality of automated search results. The following is a discussion of advances, improvements, and expansions on the pagerank concept. Algorithms and metrics are worthy of discussion as a means to gain insight into more useful optimization parameters.

SPAM MASS

Link Spam Detection Based on Mass Estimation by Gyongyi, Berkhin, Garcia-Molina, and Pedersen (Yahoo!) [6] was published in 2005. In it, the authors define “link spamming” as “an attempt to mislead search engines and trigger an artificially high link-based ranking of a specific targeted webpage.” The authors claim that “spam mass” is a measure of the impact of link spamming on a page’s ranking, further stating that pagerank is fairly exposed to spamming: “a significant increase in score requires a large number of links from low-pagerank nodes and/or some hard-to-obtain links from popular nodes, such as the New York Times.”

Gyongyi et al. [6] developed a technique to identify target nodes “x” that benefit from so-called “boosting nodes”. Boosting nodes are webpages controlled by a web spammer to artificially inflate the pagerank of the target page. The paper continues that search engines must contain a “whitelist” and/or blacklist of websites manually compiled by editors or algorithms, containing a baseline for calculating spam mass. The implication is that these search algorithms are not completely automated; rather, the search algorithms are a combination of both automation and human editing. In fact, high page-rank sites are of more interest to the human editor because they have more influence on other websites.

Understanding the spam mass concept requires a deeper understanding of the pagerank formula. As stated in [6], a technical issue exists with the pagerank formula. A web surfer or crawler (an automated program that follows links) can theoretically get stuck at a node (webpage) containing no outbound links. A probability distribution connecting these nodes has to be defined. To make the pagerank matrix complete, “virtual” links [6] must be made to connect all other nodes on the web. The pagerank equation is a “Markov chain” (statistical matrix). Pagerank indicates the probability that a web surfer will transition from page A to page B. For this mathematical model to be correct, the concept of “teleporting” or random jumping must be introduced, so that a web surfer or crawler can jump from one webpage to another, in the event it reaches a webpage with no outbound links.

In [6], the authors demonstrate that by looking at the number of incoming links to webpage X and the amount of pagerank each page contributes, the links coming from spam pages can be determined, and thus node X can be labeled as spam. By calculating the contribution of known good and known spam nodes, they can detect a spam page. If the contribution to pagerank by known good nodes is less than the contribution by known bad nodes, the target page is labeled as spam.

Gyongyi et al. [6] define absolute spam mass as a measure of how much the direct or indirect in-neighbor spam nodes increase a node’s pagerank. In this paper they admit to having a directory but do not disclose the URL, only that it contains 16,766 hosts. They also include a set of .gov hosts consisting of 55,320 hosts after URL cleaning. Finally, they include a list of educational institutions worldwide in what’s known as their known “good list.” Based on the university list, they included 434,045 individual hosts belonging to these institutions. This list of hosts indicates the value of links from these types of URLs. They also state that being one of these trusted sites can result in a negative spam mass value.

In general, the spam mass calculation indicates that a website needs to have most of its pagerank coming from trusted URLs, or it risks being identified as spam. This indication leads to the idea that spammers can harm their competitors by generating non-trustworthy links to those competitors. This potential harm would be the case if

spammers controlled the bad set of links that the authors identify as V^- . Use of the spam mass calculation requires a reliable list of trusted nodes V^+ and a reliable black-list of nodes V^- . In general, a web spammer does not know the blacklist of nodes.

The process of gathering a set of blacklist nodes can also be automated to some extent. In another paper by Gyongyi et al. [4], the authors discuss an article entitled “Combating Web Spam with TrustRank.” Remembering that TrustRank helps to automatically identify “trusted” web pages based on a small set of human-selected seed pages, the claim of this paper is that webspam pages can be derived from a seed of only 200 known human-reviewed pages.

NEAR DUPLICATE CONTENT ALGORITHMS AND FOCUSED CRAWLERS

In 2002, Charikar published a paper on the so-called “simhash [8] algorithm.” The SEO community has had long discussions on duplicate and near-duplicate content issues. For databases with billions of web documents, detecting duplicate or near-duplicate content is time-consuming and expensive. A web crawler is an automated program that follows links and gathers information on the web. A generic crawler does not compute any special information about a page, whereas a focused crawler limits the kinds of pages it crawls based on some type of calculation. Eliminating duplicate or near-duplicate content on the web greatly increases the quality of search results and limits storage requirements.

A set of definitions common in information retrieval -- the science of indexing, finding, and recalling information -- is required to gain a basic understanding of these types of algorithms. In this case, we are dealing with textual information. Below are a few useful information retrieval definitions.

Case folding: The process of making all the terms in a text string lower or upper case.

Stemming: The process of reducing a word to a root word. Stemming algorithms are part of the information retrieval process. You may, for example, reduce the word constitution to const.

Phrase detection: Extracting phrases or words from a text document. For example, “open class words” are nouns, adjectives, and verbs. These contain enough information to determine what the document is about, whereas punctuation like commas and periods give no details about content.

Tokenization: The process of breaking text up into chunks, and eliminating unnecessary characters.

Stop-word removal: There are 400 to 500 types of stop words such as “of”, “and”, “the,” etc., that provide no

useful information about the document's topic. Stop-word removal is the process of removing these words.

The information retrieval terms above give the optimizer an idea of what focused crawlers see when they detect duplicate content. Stop-words account for about 20% of all words in a typical document. These techniques greatly reduce the size of the search engine's index. Stemming alone can reduce the size of an index by nearly 40%. Once the documents have been processed using basic IR techniques, algorithms like simhash can be applied more efficiently.

The simhash algorithm begins with a "hash." A hash is the process of turning information into integers, which typically are put into a matrix, because machines can operate quickly on machines. To compare a webpage with another webpage, all unnecessary content must be removed and the text put into an array of numbers (hashing). The simhash algorithm applies these reduction techniques to a webpage to detect duplicate content violations. It generates small "footprints" (numerical representations) of a webpage, and can detect duplicate content issues for billions of webpages and indexes. These footprints need to happen within a few milliseconds.

Algorithms such as simhash can detect near-duplicate content (with some modifications), plagiarism violations, and spam. This data is also useful for posting search results on related documents. An analysis of these algorithms reveals how webpages should be constructed and what the information retrieval system is using to rank web documents. Future improvements in search technology will bring more advanced techniques in combating webspam. This continuing advancement can only result in a positive outcome for web authors creating valuable web content.

CONCLUSION

The webpage optimizer must deal with many variables at once and be cautious when intentionally optimizing webpages. Remember that search engines also monitor statistics. Keeping within the usual evolution of statistical distributions is important for any webpage. Arguments about which variables a search engine uses to rank documents or what qualities of a document a search engine can calculate can only be solved by digging into the published papers and/or experimenting. Speculation about all the parameters involved in ranking a web document runs rampant in the SEO community. In many cases, the information is based on experiments that don't apply in a real search environment.

Search engines are full of useful data, an analysis of which can yield surprising and valuable insights. Each time we analyze a paper or a patent, or understand a new term, we begin to see how a search engine can create new parameters and detection algorithms to combat webspam. Understanding the potential variables in ranking algorithms can aid the SEO consultant when making decisions on the best way to let a

search engine understand a webpage. Search engine algorithms are a small but important part of search marketing and SEO. We should be cautious when making assumptions about what ranking algorithms can and cannot detect, especially when dealing with a client's website.

The SEO should focus on providing valuable content to the user and, if needed, use available SEO tools to correctly describe the content on a webpage to a search engine algorithm.

REFERENCES

- [1] S. Brin, "The Anatomy of a Large-Scale Hypertextual Web Search Engine" (1998).
- [2] Poalo Boldi, Massimo Santini, Sebastiano Vigna, "Pagerank as a function of damping factor" DSI Università degli Studi di Milano.
- [3] Matt Cutts blog, <http://www.mattcutts.com/blog/seeing-nofollow-links/>
- [4] Gyöngyi, Zoltán; Hector Garcia-Molina, Jan Pedersen "Combating Webspam with TrustRank" Stanford University.
- [5] www.dmoz.org. Human edited directory of the web.
- [6] Gyongyi, Berkhin, Garcia-Molina, and Pedersen, "Link Spam Detection Based on Mass Estimation" technical report, Stanford University, Oct. 31, 2005.
- [7] Moses S. Charikar, "Similarity Estimation Techniques for Rounding Algorithms", ACM 2002.



Sean Gollhier. Founder and Publisher of SEMJ.org.

Sean is a consultant for companies needing help with search engine optimization and paid advertising techniques. He has optimized websites for companies in many different industries and has consulted on search engine optimization for companies like Grubb-Ellis

Commercial. Sean has a B.S. in physics from the University of Washington in Seattle, a masters in electrical engineering from Washington State University, holds four engineering patents, and began studying search engines in 1999. Sean was an affiliate marketer for four years before working as a website developer and owning a Colorado SEO company. His main research interests are search engine ranking algorithms and SEO. He frequently attends search marketing conferences in the U.S.

Note: All trademarks and registered trademarks in this paper are the property of their respective owners..