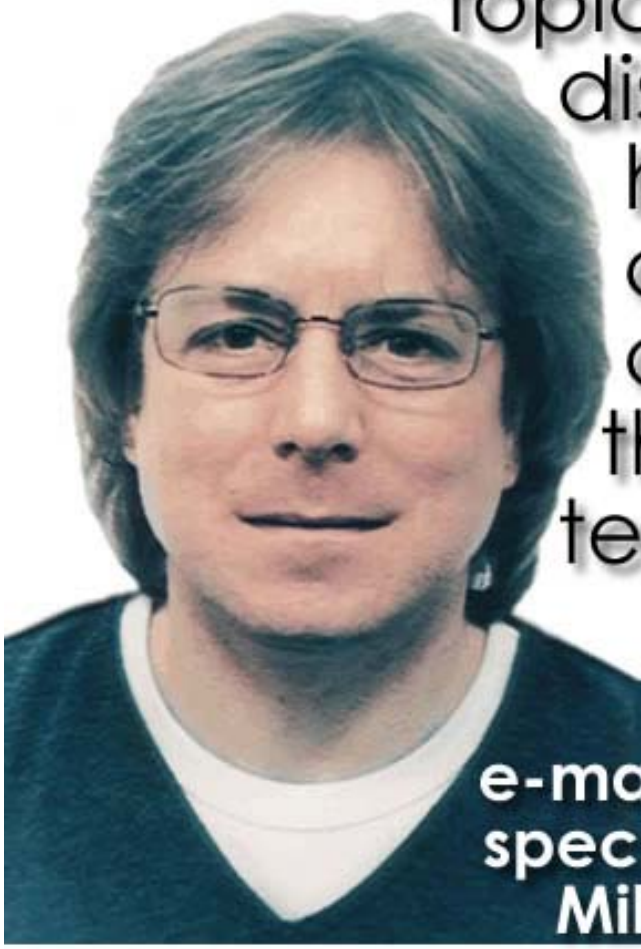
topic distillation: hits, clever, discoweb then teoma.



An e-marketing-news special feature by Mike Grehan.

Includes abstracts from the best selling eBook
Search Engine Marketing: The essential best practice guide.

introduction:

In its short history, data mining the web has come a very long way. Web crawling, web page indexing and keyword or "similarity-based" searching of web contents is a mammoth task. It's tackled on a daily basis by the web's leading search engines such as Google, Inktomi and more recently, Teoma. As the challenge becomes greater, then so does the technology as it scales in terms of both capacity and capability. Hypertext-based machine learning and data mining methods such as clustering, collaborative filtering, supervised learning and semi-supervised learning are the foundation blocks of this rapidly advancing technology. The natural algorithm of the web is based on linkage. After all, that's why it was invented. When applied to the web, the knowledge derived from *social network analysis* can tell more about web pages, than those web pages can tell about themselves.

I'm going to generalise a lot in this paper. It's not a scientific paper, but it does touch on some very technical and scientific aspects. However, the intention is to try and *not* get too technical, but to try and *simply* get across the fundamentals of what is loosely termed as "link popularity" (and why it's so important to search engines). This paper is not at all exhaustive in its content (nowhere near it in fact). It's merely a skim across the surface. Hopefully though, it may help you to understand just a little more about how search engines work, and the way that they take advantage of "information rich" web linkage data.

During the course of researching the second edition of my book, I became fascinated by the work of Professor Jon Kleinberg and an algorithm he developed which has had a major impact on search engine technology. The principle behind the formula has been used as the basis for many experiments in what's known as "topic distillation". Work in this field also had a profound effect on Professor Apostolos Gerasoulis, founder of Teoma.

It's the influence of this work, and the further development work by Jon Kleinberg himself and a team of researchers at IBM's Almaden Research Centre in California, which (in the main) provides the basis for the underlying algorithm at Teoma. So what is it about Teoma which makes it so different? Please allow me to give a very general overview of the important role of link analysis and the algorithm which, some would say, is one of the most influential in the field of information retrieval on the web.

mike grehan.

Published by
Net Writer!
Publishing

background and overview:

In order to make this document more useful to you from the very beginning, it may be easier if I start with one or two explanations. If you've read my book, 'Search Engine Marketing: The essential best practice guide', then this will be much of a "refresher". If you haven't read the book, but you're involved in search engine marketing, then you'll already be aware of the key concepts and principles of which I'm about to give an overview (I still sincerely hope you'll glean one or two useful "nuggets" though). And if you're new to all of this stuff: right here is most certainly the best place to start.

Search engine algorithms are very complex mathematical formulas which govern their entire performance. From crawling the web and indexing pages in their database, to returning relevant results to the queries they receive at the interface, there's a lot of linear algebra and pure math bubbling away under the surface.

It's fairly safe to generalise and say that, for ranking purposes, search engines take two major mathematical considerations into account: the composition and characteristics of the text parsed from HTML pages which form the corpus, and the characteristics of linkage data between HTML pages across the fraction of the web they have indexed. Obviously, the text from a page (HTML document) is very important. How else could a machine match a user query against what it has in its database, if it didn't have some indication of what was on the pages it had crawled? And as for linkage data: pages pointing (linking) to other pages can provide a massive amount of information about structure, communities and hierarchy (largely referred to as the web's "topology").

Before I go any further, let me do a couple of quick introductions and explanations to some people, terminology and technology. The title of this paper is: topic distillation: HITS, CLEVER, Discoweb then Teoma. So, to begin with: what is topic distillation? Basically, it means, given a broad topic, distil a small number of high-quality web pages that are most representative of the topic. The term is used within the search engine context of social sciences and bibliometrics which is conventionally concerned with the bibliographic citation graph of academic papers.

There is a major difference between pure search and topic distillation. In pure search, a query such as: what power zoom lens is the Nikon digital five mega pixel Coolpix 5700 - can be handled quite easily by a straight "term index" (and even more easily from a user experienced in query construction). That's because it's a specific query about a specific item. Whereas the query: what is digital logic - a much more broad query, would need to be distilled to the most generally relevant pages on the subject. By distilled, this means (in terms of bibliometrics as referenced above) finding the most authoritative web pages (please note the word PAGES).

Okay, so who's Jon Kleinberg. Well, he's a very, very smart guy (Professor of computer science at Cornell University, Ithaca) who developed an information retrieval algorithm for search engines which (like Google's PageRank) is related to the Pinski-Narin influence weights bibliometrics formulation. And what does HITS stand for? It stands for "hyperlinked induced topic search."

All of this is explained in a little more detail as you get further into this document. Of course, the word algorithm is used over and over again, and I (probably like 90% of the people I know involved in this field) assume you already know what it means. But for the benefit of those who don't, here's how I approached an explanation in my book:

I wonder if the great mathematician Al-Khwarizmi [Born 770 Uzbekistan] would ever have expected that his name would be bandied around as much as it is in the 21st century. As it's from his name that the term algorithm is derived. As I noted earlier about the crawler module being referred to in the singular, the same happens here with algorithm. But as you can see, there are many algorithms used by search engines. Just the use of the word algorithm can strike awe into the uninitiated. For sure, an algorithm developed by a search engine scientist reads like Greek to a non mathematician, but when it is explained in its simplest form, it's not too hard to grasp at all.

Algorithms are the fundamental basis for the performance of computer programmes. An algorithm is a set of instructions to automatically complete a task. In fact the word algorithm could be used to describe any automated task or list of instructions. Let's see it for what it is. We all use labour saving devices to aid us in what can be simply intensive and boring. A washing machine can now be programmed to wash, spin and dry to save us the tedious bother. Yes - it uses an algorithm to perform a routine set of tasks. How can I even more easily describe an algorithm? Here's an algorithm I frequently use myself:

Go into the lounge.

Find a small black plastic object with buttons which can be held in the hand.

Point it at the TV and press button number five for football game.

Go to the fridge and take out a cold beer.

Sit down in armchair and remove opening device from can.

Place can to lips and drink.

Of course, you could do the same thing by getting the beer first and then putting its contents into a glass before you go to the lounge and sit down to press button number five on the hand held device. Which is the best algorithm? Well that just depends on the person and the circumstances. Do you prefer to drink out of a can or a glass: and would you put the TV on first – or get the beer first?

Now, let me quickly introduce Larry Page and Sergey Brin, founders of Google. [As my wife is Russian, needless to say, we're both proud supporters of Muscovite Sergey Brin] Larry and Sergey get a mention because they too, as already briefly mentioned, developed an algorithm which is based on linkage data (PageRank).

And interestingly, in the original research presentation for Google, you'll see a mention of Jon Kleinberg's work: just as you'll see a reference to Page/Brin in Jon Kleinberg's presentation. PageRank is based on linkage data, but it's only one of many important factors which make up the entire ranking algorithm for Google (as HITS would be only one of a number of determining factors for another search engine). Let's just take a brief look at how conventional text based information retrieval has been initially integrated into web search, and how linkage data has further developed to almost supersede its importance.

flat corpus text retrieval Vs html pages:

All search engines use some form of hyperlink analysis as it significantly improves the relevance of search results. Classic information retrieval methods have used algorithms which are based only on the words in a document i.e. automatic text retrieval. Perhaps the most distinguished, is Salton's vector space model. This approach, an explanation of which is beyond the scope of this document (although covered quite extensively in my book), has been integrated by web search engines since Brian Pinkerton developed WebCrawler, the web's first full-text retrieval crawler based search engine and Michael Mauldin developed Lycos.

The principle of this method is the conversion of documents and queries to "term vectors" in a high dimensional vector space with one dimension per term. Here is a good place to "skip past" the point in my book which describes 'how term vectors are created' and refer to the simplified explanation of 'how they're used' by web search engine innovator Brian Pinkerton:

Perhaps the best way to understand what's going on here is to think of the process of answering a query. Simplifying a bit, it's:

1) normalize the query

2) find the total set of documents that match the query

3) rank the elements of that set according to some rules

4) get info about the top results, and display a results page

This is the process for most search engines. Search engines differ most notably on step 3: how they rank the results. For example, say the searcher is looking for "Greenpeace and France." Providing that they have comparable crawls, most search engines will generate a similar unranked set of results for this query.

For instance, it'll probably include the home page of Greenpeace France, and some articles on that nasty business in the South Pacific. The difference is how the search engines rank this set and determine the top 25 results.

With the vector-space retrieval model (classic Salton and as I used it in the first WebCrawler) is actually pretty simple: documents in the result set are ranked based on how close words in the query match words in the documents. The more closely they match, the higher the rank of the document. Typically, though not necessarily, a word is more important in one document than another if it occurs more frequently in that first document.

This model works really well for situations in which the searchers use long queries, or where there are only a few documents that are good matches for the query. For instance, the average query length in Lexis (the big legal database) was 60 words at one point!

In situations where the query is small (the Web average is still about 3 words), the vector model doesn't distinguish among the resulting documents very well. To continue on the Bill Clinton example you gave [MG: this was an example I quoted from the original research document prepared by Sergey Brin and Larry Page of Google in 1998] many documents are matches for this query, and it's hard to tell which documents match it better than others. For example, suppose I have a "Bill Clinton Sucks" page and the White House has a "Bill Clinton, President" page. Further assume that the pages are the same length and both mention the phrase "Bill Clinton" the same number of times. How is the vector space model going to know that more people would prefer the latter page to the former? It wouldn't.

This example shows where the vector model breaks down. It's especially bad on the Web because the number of pages is huge and the query size is so small.

So, we need some new way of ranking documents that can help us rank the set better, or simply help winnow the list down to a useful size.

This is where the link data (Web structure) comes in. It can be used to assist (or even take over) the ranking of documents, to determine a subset of documents that are worth querying, to expand a small set of search results, and many other tasks.

I used link data in a prototype of WebCrawler in what I still think is the most useful form: in combination with a full-text retrieval model.

Google uses it purely, other search engines use it in some way.

That's a very simplistic overview of what is a very complicated process, but I think it does suffice for the purpose of this document. It may not be totally obvious, but this "classic Salton" approach does not scale well with the web. And more to the point, any algorithm which is based purely on the content of a page, is susceptible to manipulation. Maybe we should just use the given term for it here: Spamming!

So what's a search engine to do? This is where the power of hyperlink analysis comes in. Hyperlink analysis uses the content and linkage data of other pages to provide connectivity based ranking. This means, in strictly layman's terms, that to a search engine, it's more important what other pages say about you, than what you say about yourself.

Social networks, bibliometrics, citation analysis:

In the main, the two algorithms developed to ‘data-mine’ and analyse link structures on the web are HITS [Kleinberg 1998] and PageRank [Brin, Page 1998]. PageRank is explained in detail by Chris Ridings in his excellent document PageRank Uncovered (you can get this document free with my book). And of course, for the purpose of this paper, I'm dealing (mainly) with HITS.

There are many, many papers on the subject of information retrieval on the web which reference both Jon Kleinberg and HITS as well as making reference to "hubs and authorities". So why is HITS so significant? Well, for one thing, it was a major leap forward from relying on text based retrieval only. As has already been discovered, text based retrieval methods are good at finding relevant documents following a query (in the case of the web - millions and millions of them). However, just because they are relevant doesn't mean they are the most useful, or for that matter, the most important. Kleinberg himself calls this the "abundance problem" and states that the number of pages that could reasonably be returned as relevant is far too large for a human user to digest.

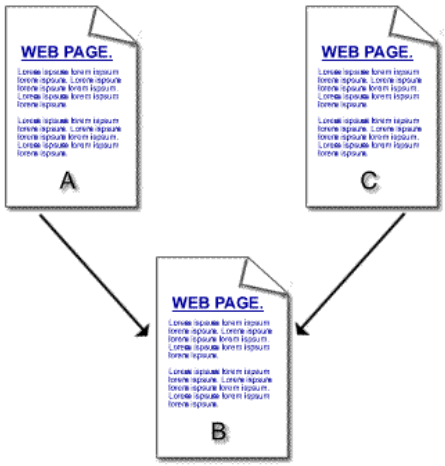
So, the dilemma encountered by search engines is twofold: how do we get a better set of results? And how do we protect and prevent those results from being "manipulated" by external forces? Well, the obvious way to do it, is to take the focus away from the words on a web page i.e. what a page says about itself, and look at what other people say about it in the form a of a vote, or citation. In short, let's look and see who links/points to it. The power of this kind of data is based on a simple assumption: web page creators are most likely to place links to other pages on the same/similar topic. It's also assumed that these other web page creators are motivated to point to other "quality" pages on the given subject matter.

I still laugh when I think about a conversation I had with Andrei Broder, Chief Scientist at Alta Vista (at the time), when he said to me: "It's not very often you'll find a webmaster saying, those are the worst pages I've ever seen on the subject so I'll link to them!"

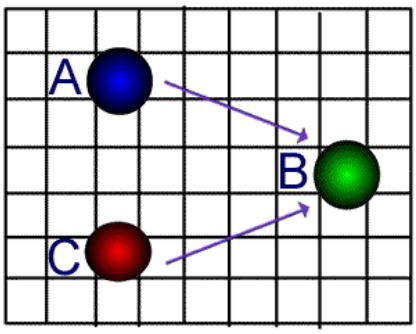
Both research papers by Kleinberg and Page/Brin refer to "bibliometrics" and "citation analysis" which is a tool developed in information science to identify the core sets of articles, authors, or journals of particular fields of study.

Again, this is covered in more detail in my book, but for the moment, I'll give just a simple overview. Read any research paper of any significance and at the end of the paper you'll find the bibliography. This is where the author of the paper will "cite" the work of other researchers/scientists in the field. By tracking these "citations" in a particular field of study, one can usually discover the person who is largely regarded as the *expert* in the field. The likelihood is that his work is "cited" by more researchers/scientists than any other.

It's also possible to discover "co-citation" i.e. more than one author or work mentioned together with regularity in many other documents. Search engines view the web as a graph. The same applies when looking at the "topology" (linkage data) of the web via a link graph. Taking the citation co-citation principle, as used in conventional bibliometrics, hyperlink analysis algorithms can make either one or both of these basic assumptions: A hyperlink from page a to page b is a recommendation of page b by the author of page a and creates a ‘directed edge’ in the (web) link graph {A,B}



Set of linked pages.

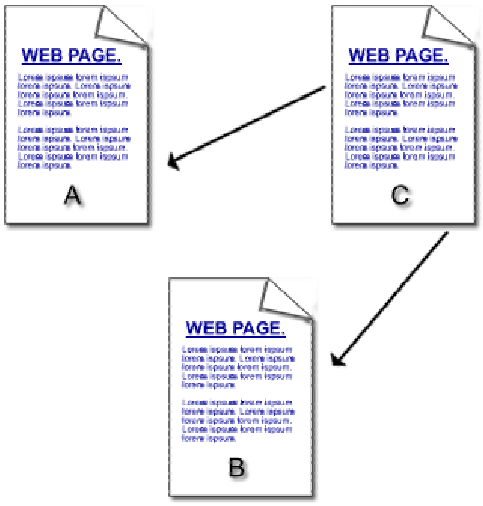


Link graph.

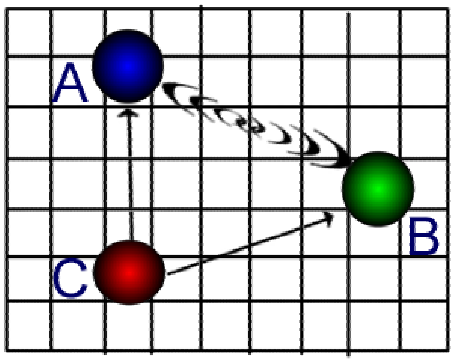
Web pages linked together are nodes in the web graph.

When web page A links to web page B this is a 'directed edge'.

If page a and page b are connected by a hyperlink, then they may be on the same topic. Some algorithms also use an undirected co-citation graph. A and B are connected by an undirected edge, if and only if there is a third page C which links both to A and B



Set of linked pages.



Link graph.

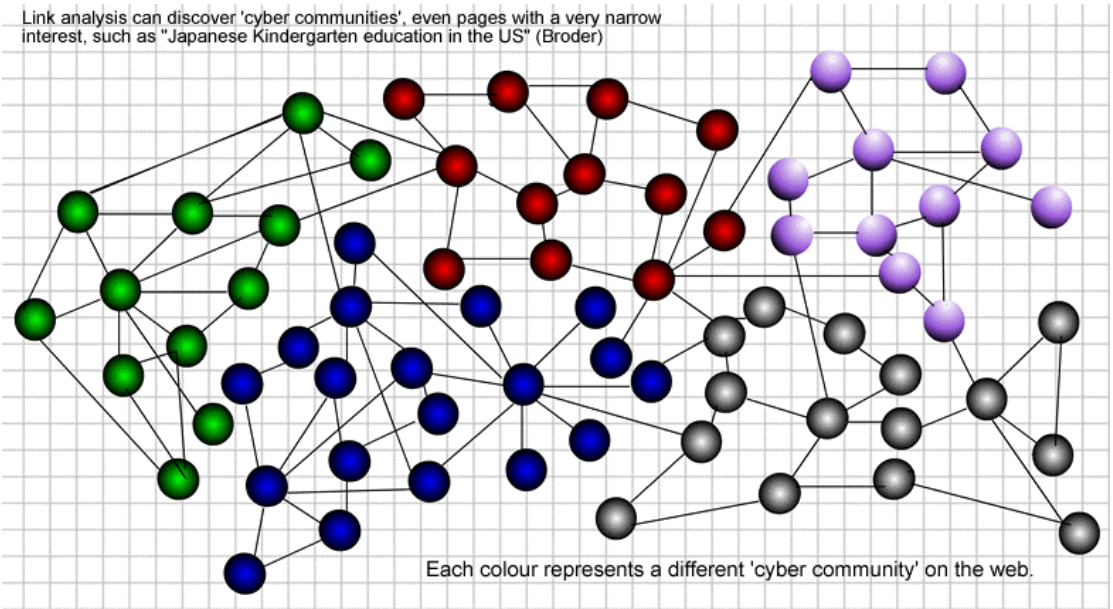
If page C links to both A and B, then A and B are connected by an undirected edge in the graph and are viewed as being co-cited by C.

By using this methodology search engines can attempt to identify the intellectual structure (topology) and social networks (communities) of the web. However, there are many problems with scaling using methods of citation and co-citation analysis to deal with hundreds and hundreds of millions of documents with billions of citations (hyperlinks).

‘Cyberspace’ (as in the web) already has its communities and neighbourhoods. OK – less real in the sense of where you live and who you hang out with. But there is a sociology to the web. Music lovers from different cultures and different backgrounds (and time zones) don’t live in the same geographical neighbourhood – but when they are linked to each other on the web: they do. Just the same as art lovers and people from every walk of life who post their information to the web and form these communities or ‘link neighbourhoods’ in ‘cyberspace’.

If you read the interview with Andrei Broder in my book, you’ll see that, when we are talking about the connectivity server (an experiment to visualise the connectivity of the web) and I mention link popularity, he replies: “it’s about link popularity - but much more than that."

He quotes how he can find pages of a ‘very narrow interest’ and map them: “I could find a small community interested in, say, Japanese Kindergarten education in the US. By dissecting the linkage information, I can find even these types of tiny communities”. It’s about as an obscure example as you could make, but these pages contain information on a variety of other subjects also, including, diet, health and social issues for children – but the linkage itself determines a certain *basic* connection for that subgroup on the web.



By identifying that type of community, it helps not only in the sociological evolution of the web, but also by providing information on people (in detail) with combined focused interests. This is the ‘signature’ of a community on the web. Web communities at their core contain a dense pattern of linkage.
Here we have thematically cohesive web communities: but not essentially thematically cohesive, constrained web sites, as in the web propaganda notion of "themed web sites". The buzz about "page authority" as it’s known within search engine optimisation circles is, within reasonable understanding, relatively new and mainly topical because Google has been so ‘visible’ about it.

Yet this type of experimental research was actually carried out as early as the development of the second phase of WebCrawler, and also with Inktomi in preliminary studies at Berkeley in the early-to-mid nineties.

Just as much attention as has been given to automatic text retrieval and indexing, is now given to the structure and linkage of the web. Web connectivity and its ‘topology’ provides many clues to search engines as to the "importance" and the content of any given web page. An importance which can also be conferred to another.

Of course, the links connecting web pages together, in principle are equivalent. The web itself holds no preference for one link over another. Some links on web pages are simply navigational aids to ‘browse’ a site. Other links may provide access to other pages which augment the content of the page containing them. High quality pages with good, clear and concise information are more likely to have many links pointing to them. Whereas low quality pages will have fewer links or none at all.

hubs and authorities:

At some point, you may have already come across the term "hubs and authorities". This was coined by Jon Kleinberg during the development of his HITS algorithm. As you are now aware, linkage data provides another set of heuristics to take into account when it comes to ranking. Or as it's otherwise referred to by search engine marketers: "off the page criteria."

Let me attempt to give a brief and simple explanation of how the HITS process works, so that it's easier to understand the principle idea behind "hubs and authorities". HITS begins with a search on a specific query. The first two hundred results or so returned are then used to provide a linkage based ranking order where the actual words used in the query are no longer significant.

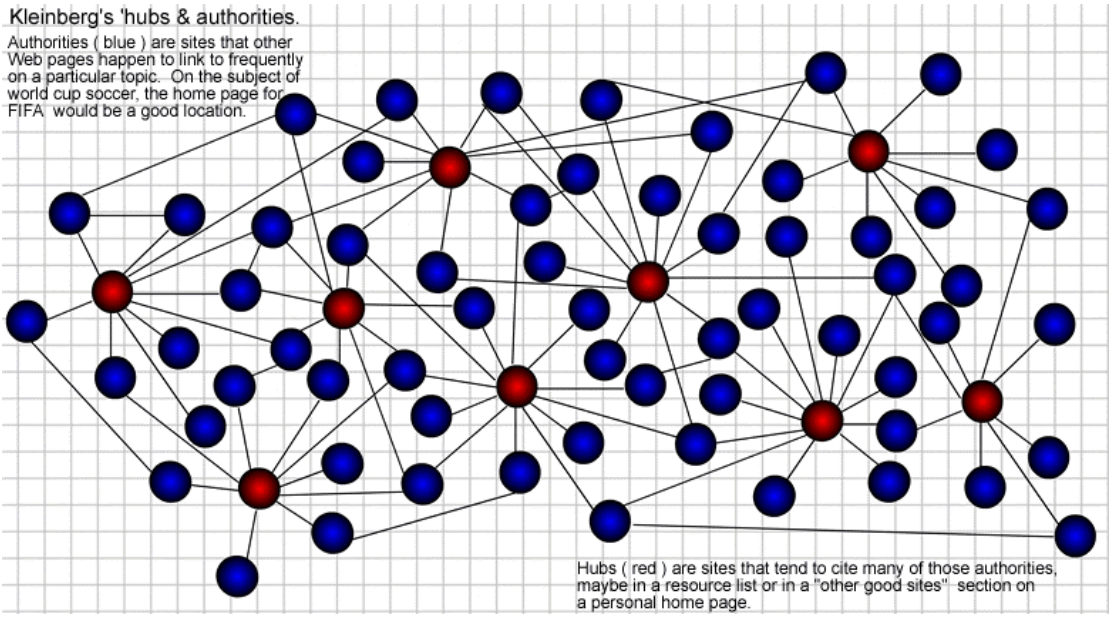
I'll try and explain that a little more clearly. The "hubs and authorities" approach makes it easier to identify a really popular page on a given subject, even if the actual words don't appear anywhere on the page. Again, Kleinberg gives the example that, for instance, there is no reason to expect that the home pages of Toyota or Honda should contain the term "automobile manufacturers", yet these are very much "authoritative pages".

Beginning with a search topic, specified by one or more query terms, the HITS algorithm applies two main steps: a sampling component which constructs a focused collection of several thousands of web pages which are likely to be rich in relevant ‘authorities’. And a weight-propagation component which determines numerical estimates of ‘hub’ and ‘authority’ weights by an iterative procedure. The pages with the highest weights are returned as ‘hubs’ and ‘authorities’ for the search topic.

Hubs and authorities exhibit a mutually reinforcing relationship: a good hub points to many good authorities; a good authority is pointed to by many good hubs (pages can be both good authorities and good hubs). Here's how it goes:

Starting from a user performed query, HITS assembles an initial set of pages: typically, up to 200 pages are returned by a full text search engine on that query. These pages are then expanded to a larger root set by adding any pages that are linked to or from any page in the initial set. HITS then associates with each page *p* a hub-weight h(p) and an authority weight a(p), all initialised to 1. HITS then iteratively updates the hub and authority weights of each page in the root set. First, under the intuition that a page pointing to good authorities should be considered a good hub, it replaces the hub score of each page by the sum of the authorities of the pages it points to. And second, dually, under the intuition that a page pointed to by good hubs should be considered a good authority, it replaces the authority score of each page by the sum of the hub scores of the pages that point to it.

The update operations are performed for all the pages, and the process repeated (normalising the weights after each iteration) for some number of rounds. Following this, the pages with the highest h(p) and a(p) scores are output as the best hubs and authorities. Again, let me try to simplify this: authorities are web pages with good content on a specific topic. And hubs are directory like pages with many hyperlinks to those pages on the topic. So, a page that points to many others should be a good hub, and a page that many pages point to, should be a good authority.



In its basic principle, this innovation (or expansion on citation and link analysis) is an ideal solution to help ease the problems search engines suffer with mainly text based retrieval. But applying it to ‘Cyberspace’ and real world web search has detected its flaws. A lot of further research to ‘improve’ or ‘enhance’ the HITS algorithm has been carried out.

Like all major new developments, in the early stages, the obvious flaws (from a general purpose search engine point of view) with HITS became immediately apparent. The first one is quite obvious, and that's the amount of time taken to collate the data and then return relevant results following a "hard" query. Certainly some of this work can be done "up front" as is the case with Google's PageRank. PageRank uses the "power iterative" method for what are known as "eigenvectors" offline, over the whole web graph. This up-front ranking is then stored in the database. This provides the major advantage that there is no additional run-time link analysis penalty during the query search process.

However, even this approach creates its own problems, in that, rankings can be dominated by "strong" pages which are not relevant to the query. By that, I mean, once the principle eigenvector is established in the link graph (the web community determined by linkage following a keyword search), there are bound to be a number of pages which have an unfair advantage in the ranking because of their greater linkage, yet they may not be relevant to the actual query.

The CLEVER (Clientside Eigenvector Enhanced Retrieval) project, developed at IBM's Almaden Research Centre in San Jose, (of which Jon Kleinberg was a team member as a visiting scientist) uses a version of HITS. Remember that the HITS concept relies on the assumption that if site A is pointed to by many other sites, then they infer authority to A. However, the definition of hubs and authorities as stated is not very helpful in determining who they are, but as already stated, you can use an intuitive alternate definition: good hubs point to many good authorities, and good authorities are pointed to by good hubs.

This “frustratingly circular definition” as it has been referred to, was solved in the CLEVER project, which used spectral filtering techniques to find the best hubs and top authorities on any given topic. The improved algorithm doesn't merely count links to make its distinctions it also considers clues within the pages, such as whether the query term is located within or near the link, to ultimately re-rank the original list of sites and present a more accurate measure of relevancy. Users in an IBM-sponsored study found CLEVER’s results as good or better than Yahoo!'s 81 percent of the time.

CLEVER assumes that two pages on the same website were created by the same company or individual, and so should not be allowed to confer authority to one another. To address this, CLEVER varies the weights of the links between pages based on the domains of their end-points. CLEVER seeks a final set of hubs and authorities that provide good access to a wide variety of information. For instance, two pages that are extremely high quality but contain virtually identical information would not both be returned. To this end, after CLEVER outputs a hub, it diminishes the scores of pages that are very similar to that hub.

CLEVER returns only a single point-of-entry into a particular internet resource. This quote from the research team behind CLEVER describing a routine within their experimenting is something I found to be quite fascinating:

"We often encounter situations in which a good hub page, for instance, appears with a different level of generality than the query for which it would be useful. As an example, consider the query "mango fruit." A high-quality hub page devoted to exotic fruit might have a section of links on papaya, another section on mango, and finally a section on guava. However, if we consider the page to be a universally good hub, the unrelated destination pages about papaya and guava will be considered to be good authorities. To address this, we identify interesting (physically contiguous) sections of web pages and use these sections to determine which other pages might be good hubs or authorities in their entirety."

Monica Henzinger (gets mentioned quite a lot in my book) is Director of Research at Google and presides over a group of 10 computer scientists in her research team (at the time of publication of the 2nd edition of my book). A German born PhD she works on improving Google's search functionality and moving Google into new areas such as mobile phone and voice-activated searching. In fact, a couple of years ago, Google was approached by the German car manufacturer BMW who wanted to put a voice-activated search into their 7 series cars (presumably drivers would be expected to stop the car in order to do this and not crash on the highway using a mobile phone whilst viewing a small monitor to check their stocks and shares!).

Formerly with Digital Equipment Corporation (DEC) Systems Research Centre, she has conducted much research with other computer scientists (including Andrei Broder, also formerly with DEC Systems Research Centre) into the web's connectivity. She worked with Andrei Broder on (among others) Alta Vista's Connectivity Server project [Bharat, Broder, Henzinger et al - 1999].

In a further experiment into 'topic distillation' [categorisation and then classification] with Krishna Bharat [Bharat, Henzinger – 2000] they discovered three problems with connectivity analysis as suggested by Kleinberg with this 'links only' approach. The first they describe as: Mutually Reinforcing Relationships Between Hosts. Further described as "where certain arrangements of documents 'conspire' to dominate the computation" (I think we could simply refer to this as 'link Spamming' – 'hub' and 'authority' look-alikes). The second problem they refer to as: Automatically generated Links. This is further described as "where no human opinion is expressed by the link" (think web authoring tools, database conversion tools, or a hypernews system which turns news articles into web pages and then automatically inserts links to the site). The third problem is referred to as: non relevant nodes. Further described as "documents in the neighbourhood graph which are not relevant to the query topic (here they give an example of a query for 'jaguar and car' where the algorithm drifts more towards the general topic of car and returns pages from different car manufacturers as top 'authorities' and lists of car manufacturers as the best 'hubs').

The third problem mentioned, of non relevant nodes is the most common problem when using 'link only' analysis. Which is why it is necessary to also use content analysis in an attempt to keep the computation 'on topic'.

So, what's the difference between HITS and PageRank? Again, here's Monica Henzinger with her official Google hat on:

“The PageRank algorithm differs from HITS in that it computes the rank of a page by weighting each hyperlink to the page proportionally to the quality of the page containing the hyperlink. To determine the quality of a referring page they use its PageRank recursively, with an arbitrary initial setting of the PageRank values. The formula shows that the PageRank of page a – depends on the PageRank of page b pointing to page a [co citation]. Since the PageRank definition introduces one such linear equation per page, a huge set of linear equations need to be solved in order to compute PageRank for all pages. [Henzinger 2001]

A much more in-depth and detailed analysis of the PageRank algorithm has been prepared by Chris Ridings and is distributed as a free supplement when you purchase my book.

from discoweb to Teoma:

In 1999, Apostolos Gerasoulis, Professor of Computer Science at Rutgers University, New Jersey, became intrigued by CLEVER, Google and the work of the web archaeology team at Compaq’s research centre.

Whilst working on a research project exploring how to sift mountains of data with supercomputers for the Defence Advanced Research Projects Agency [DARPA], he sensed a tie-in to search engines. With his own research team at Rutgers, he developed a prototype search engine called DiscoWeb, a play on the word ‘discover’ (because it DISCOvers WEB communities – nothing to do with any Saturday Night Fever connotations!).

By using link analysis, as described so far in this document, DiscoWeb was a further development on HITS to ‘pull together’ highly interconnected web pages that typically share a topic or focus, and then automatically build web directories. Professor Gerasoulis is also the founder of Teoma Technologies, still the "new kid on the block" in the search engine world.

The connection (no pun intended) by Gerasoulis to the work carried out on the CLEVER project is extremely evident even in the name of his search engine, as Teoma is a Gaelic word for EXPERT.

Teoma takes advantage of all of the previous research work carried out and uses compact mathematical modelling of the web’s structure and its ordering and ranking. This is based on multi-parametric analysis to achieve its high degree of relevance and quality. So goes the "blurb" from the press release to go with the launch of Teoma.

The major advancement, of course, is the work carried out on Discoweb to speed up the actual "convergence" which (at that time) took less than a minute to provide results.

However, it may have been a major step forward, but would anybody really sit at a search engine interface and wait for one minute to see the results? Run-time link analysis is still a problem (at this stage of link analysis algorithms) when relating to an interactive search engine which must return results in a matter of seconds. The breakthrough in response time eventually came with the launch of Teoma. How does Teoma do it? I'm afraid for that: you'll have to watch this space ;-)

In September 2001 Teoma Technologies was acquired by Ask Jeeves. Teoma technology replaced Direct Hit. Teoma 2.0 was launched early 2003

www.teoma.com

In 2001 another approach to 'fine tuning' Kleinberg's HITS was presented: SALSA (Stochastic Approach to Link Structure Analysis. At the time of writing the 2nd edition of my book, SALSA had progressed from being part 'anecdotal' part 'research', to another ongoing research project with IBM.

For the purpose of being thorough I should also make reference to 'Hilltop' which is another variation algorithm developed by Krishna Bharat, an expert in the field and a member of the research team at Google.

And for the "one to watch", you'd best keep an eye on Wisenut. Yeogirl Yun graduated from Stanford with an MS in computer science in 1995. In 1998 he founded My Simon the search and compare price portal which he steered to a $700 million acquisition by CNET. He then founded Korea-Wisenut in 1999 and then Wisenut in 2000. The acquisition of Wisenut by Looksmart in a $9.25 million stock deal was intended to be able to provide Looksmart visitors with both directory and context sensitive results. How does Wisenut work? Wisenut's patent applied for technology works on an expert pages, link and link anchor text analysis detail in a similar way to that of Teoma.

www.wisenut.com

This new generation search technology is showing a major leap forward in being able to achieve much more relevant results at the interface following a query at search engine. Teoma now powers Ask Jeeves. Yahoo! has recently purchased Inktomi and its new generation technology. Is it now likely that Looksmart/MSN will drop Inktomi in favour of Wisenut? And what changes will we see at old favourite Alta Vista, following its acquisition by Overture?

Don't you just LOVE this business?