Mining the Link Structure of the World Wide Web

Soumen Chakrabarti* Byron E. Dom* David Gibson†
Jon Kleinberg‡ Ravi Kumar* Prabhakar Raghavan*
Sridhar Rajagopalan* Andrew Tomkins*

February, 1999

Abstract

The World Wide Web contains an enormous amount of information, but it can be exceedingly difficult for users to locate resources that are both high in quality and relevant to their information needs. We develop algorithms that exploit the hyperlink structure of the WWW for information discovery and categorization, the construction of high-quality resource lists, and the analysis of on-line hyperlinked communities.

1 Introduction

The World Wide Web contains an enormous amount of information, but it can be exceedingly difficult for users to locate resources that are both high in quality and relevant to their information needs. There are a number of fundamental reasons for this. The Web is a hypertext corpus of enormous size — approximately three hundred million Web pages as of this writing — and it continues to grow at a phenomenal rate. But the variation in pages is even worse than the raw scale of the data: the set of Web pages taken as a whole has almost no unifying structure, with variability in authoring style and content that is far greater than in traditional collections of text documents. This level of complexity makes it impossible to apply techniques from database management and information retrieval in an "off-the-shelf" fashion.

Index-based search engines for the WWW have been one of the primary tools by which users of the Web search for information. The largest such search engines exploit the fact that modern storage technology makes it possible to store and index a large fraction of the WWW; they can therefore build giant indices that allow one to quickly retrieve the set of all Web pages containing a given word or string. A user typically interacts with them

*IBM Almaden Research Center, 650 Harry Road, San Jose CA 95120.

†Computer Science Division, Soda Hall, UC Berkeley, CA 94720. This research was performed in part while visiting the IBM Almaden Research Center.

‡Department of Computer Science, Cornell University, Ithaca NY 14853. Supported in part by an Alfred P. Sloan Research Fellowship and by NSF Faculty Early Career Development Award CCR-9701399. This research was performed in part while visiting the IBM Almaden Research Center.

by entering query terms and receiving a list of Web pages that contain the given terms. Experienced users can make effective use of such search engines for tasks that can be solved by searching for tightly constrained keywords and phrases; however, these search engines are not suited for a wide range of equally important tasks. In particular, a topic of any breadth will typically contain several thousand or several million relevant Web pages; at the same time, a user will be willing to look at an extremely small number of these pages. How, from this sea of pages, should a search engine select the "correct" ones?

Our work begins from two central observations. First, in order to distill a large search topic on the WWW down to a size that will make sense to a human user, we need a means of identifying the most "definitive," or "authoritative," Web pages on the topic. This notion of authority adds a crucial second dimension to the notion of relevance: we wish not only to locate a set of relevant pages, but rather the relevant pages of the highest quality. Second, the Web consists not only of pages but of *hyperlinks* that connect one page to another; and this hyperlink structure contains an enormous amount of latent human annotation that can be extremely valuable for automatically inferring notions of authority. Specifically, the creation of a hyperlink by the author of a Web page represents an implicit type of "endorsement" of the page being pointed to; by mining the collective judgment contained in the set of such endorsements, we can obtain a richer understanding of both the relevance and quality of the Web's contents.

There are many ways that one could try using the link structure of the Web to infer notions of authority, and some of these are much more effective than others. This is not surprising: the link structure implies an underlying social structure in the way that pages and links are created, and it is an understanding of this social organization that can provide us with the most leverage. Our goal in designing algorithms for mining link information is to develop techniques that take advantage of what we observe about the intrinsic social organization of the Web.

Searching for Authoritative Pages. As we think about the types of pages we hope to discover, and the fact that we wish to do so automatically, we are quickly led to some difficult problems. First, it is not sufficient to first apply purely text-based methods to collect a large number of potentially relevant pages, and then comb this set for the most authoritative ones. If we were trying to find the main WWW search engines, it would be a serious mistake to restrict our attention to the set of all pages containing the phrase "search engines." For although this set is enormous, it does not contains most of the natural authorities we would like to find (e.g. Yahoo!, Excite, InfoSeek, AltaVista). Similarly, there is no reason to expect the home pages of Honda or Toyota to contain the term "Japanese automobile manufacturers," or the home pages of Microsoft or Lotus to contain the term "software companies." Authorities are often not particularly self-descriptive; large corporations for instance design their Web pages very carefully to convey a certain feel, and project the correct image — this goal might be very different from the goal of describing the company. People outside a company frequently create more recognizable (and sometimes better) judgments than the company itself.

These considerations indicate some of the difficulties with relying on text as we search for authoritative pages. There are difficulties in making use of hyperlink information as well.

While many links represent the type of endorsement we discussed above (e.g. a software engineer whose home page links to Microsoft and Lotus), others are created for reasons that have nothing to do with the conferral of authority. Some links exist purely for navigational purposes ("Click here to return to the main menu") or as paid advertisements ("The vacation of your dreams is only a click away"). Our hope is thus that in an *aggregate* sense, over a large enough number of links, our view of links as "conferring authority" will hold.

Modeling the Conferral of Authority. We have already argued that link-based analysis of the Web works best if it is rooted in the social organization of Web pages. How, then, can we best model the way in which authority is conferred on the Web? We noted above that authoritative pages are often not very self-descriptive; it is also the case that authorities on broad topics frequently don't link directly to one another. It is clear why this should be true for any topic with a commercial or competitive aspect; AltaVista, Excite, and InfoSeek may all be authorities for the topic "search engines," but they may well have no interest in endorsing one another directly.

If the major search engines do not explicitly describe themselves as such, and they do not link to one another, how can we determine that they are indeed the most authoritative pages for this topic? We could say that they are authorities because a large number of relatively anonymous pages that *are* clearly relevant to "search engines" have links to each of AltaVista, Excite, and Infoseek. Such pages are a recurring component of the Web: "hubs" that link to a collection of prominent sites on a common topic. These hub pages can appear in a variety of forms, ranging from professionally assembled resource lists on commercial sites to lists of recommended links on individual home pages. Hub pages need not themselves be prominent, or even have any links pointing to them at all; their distinguishing feature is that they are potent conferrers of authority on a focused topic. In this way, they actually have a role that is dual to that of authorities: a good authority is a page that is pointed to by many good hubs, while a good hub is a page that points to many good authorities [1].

This mutually reinforcing relationship between hubs and authorities will serve as a central theme in our exploration of link-based methods for search, the automated compilation of high-quality Web resources, and the discovery of thematically cohesive Web communities.

2 HITS: Computing Hubs and Authorities

We now describe the *HITS algorithm* [1], which computes lists of hubs and authorities for WWW search topics. 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 thousand Web pages 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.

We view the Web as a *directed graph*, consisting of a set of *nodes* with directed *edges* between certain pairs of the nodes. Given any subset S of nodes, they induce a *subgraph* containing all edges that connect two nodes in S. The first step of the HITS algorithm constructs the subgraph in which we will search for hubs and authorities. Our goal is to

have a subgraph that is rich in relevant, authoritative pages; we construct such a subgraph as follows. We first use the query terms to collect a *root set* of pages (say, 200) from an index-based search engine of the type described in the introduction. We do not expect that this set necessarily contains authoritative pages; however, since many of these pages are presumably relevant to the search topic, we expect at least some of them to have links to most of the prominent authorities. We therefore expand the root set into a *base set* by including all pages that are linked to by pages in the root set, and all pages that link to a page in the root set (up to a designated size cut-off). This follows our intuition that the prominence of authoritative pages is typically due to the endorsements of many relevant pages that are not, in themselves, prominent. We restrict our attention to this base set for the remainder of the algorithm; we find that this set typically contains roughly 1000-5000 pages, and that (hidden) among these are a large number of pages that one would subjectively view as authoritative for the search topic.

We work with the subgraph induced by the base set, with one modification. We find that links between two pages with the same WWW domain very often serve a purely navigational function, and thus do not correspond to our notion of links as conferring authority. By "WWW domain" here, we mean simply here the first level in the URL string associated with a page. We therefore delete all links between pages with the same domain from the subgraph induced by the base set, and apply the remainder of the algorithm to this modified subgraph.

We extract good hubs and authorities from the base set by giving a concrete numerical interpretation to the intuitive notions developed in the previous section. We associate a non-negative *authority weight* x_p and a non-negative *hub weight* y_p with each page $p \in V$. We will only be interested in the *relative* values of these weights, not their actual magnitudes; so in our manipulation of the weights, we apply a normalization so that the their total sum remains bounded. (The actual choice of normalization does not affect the results; we maintain the invariant that the squares of all weights sum to 1.) A page p with a large weight x_p (resp. y_p) will be viewed as a "better" authority (resp. hub). Since we do not impose any *a priori* estimates, we set all x- and y-values to a uniform constant initially; we will see later, however, that the final results are essentially unaffected by this initialization.

We now update the authority and hub weights as follows. If a page is pointed to by many good hubs, we would like to increase its authority weight; thus we update the value of x_p, for a page p, to be the sum of y_q over all pages q that link to p:

$$x_p = \sum_{q \text{ such that } q \to p} y_q, \tag{1}$$

where the notation $q \to p$ indicates that q links to p. In a strictly dual fashion, if a page points to many good authorities, we increase its hub weight via

$$y_p = \sum_{q \text{ such that } p \to q} x_q. \tag{2}$$

There is a more compact way to write these updates, and it turns out to shed more light on what is going on mathematically. Let us number the pages $\{1, 2, \ldots, n\}$ and define their *adjacency matrix* A to be the $n \times n$ matrix whose $(i, j)^{\text{th}}$ entry is equal to 1 if page i links to page j, and is 0 otherwise. Let us also write the set of all x-values as a vector

$x = (x_1, x_2, \ldots, x_n)$, and similarly define $y = (y_1, y_2, \ldots, y_n)$. Then our update rule for x can be written as $x \leftarrow A^T y$ and our update rule for y can be written as $y \leftarrow Ax$. Unwinding these one step further, we have

$$x \leftarrow A^T y \leftarrow A^T A x = (A^T A)x \tag{3}$$

and

$$y \leftarrow Ax \leftarrow AA^T y = (AA^T)y. \tag{4}$$

Thus the vector x after multiple iterations is precisely the result of applying the *power iteration* technique to $A^T A$ — we multiply our initial iterate by larger and larger powers of $A^T A$ — and a standard result in linear algebra tells us that this sequence of iterates, when normalized, converges to the principal eigenvector of $A^T A$. Similarly, we discover that the sequence of values for the normalized vector y converges to the principal eigenvector of AA^T. (See the book by Golub and Van Loan [2] for background on eigenvectors and power iteration.)

In fact, power iteration will converge to the principal eigenvector for any "non-degenerate" choice of initial vector — in our case, for example, for any vector all of whose entries are positive. This says that the hub and authority weights we compute are truly an *intrinsic* feature of the collection of linked pages, not an artifact of our choice of initial weights or the tuning of arbitrary parameters. Intuitively, the pages with large weights represent a very "dense" pattern of linkage, from pages of large hub weight to pages of large authority weight. This type of structure — a densely linked *community* of thematically related hubs and authorities — will be a motivating picture in many of the developments to follow.

Finally, the output of HITS for the given search topic is a short list consisting of the pages with the largest hub weights and the pages with the largest authority weights. Thus we see that HITS has the following interesting feature: after using the query terms to collect the root set, the algorithm completely ignores textual content thereafter. In other words, HITS is a purely link-based computation once the root set has been assembled, with no further regard to the query terms. Nevertheless, HITS provides surprisingly good search results for a wide range of queries. For instance, when tested on the sample query "search engines" that we discussed above, the top authorities returned by HITS were Yahoo!, Excite, Magellan, Lycos, and AltaVista — even though none of these pages (at the time of the experiment) contained the phrase "search engines." This confirms the intuition expressed in the introduction, that in many cases the use of hyperlinks can help circumvent some of the difficulties inherent in purely text-based search methods.

Trawling the Web for emerging cyber-communities.

The Web harbors a large number of communities — groups of content-creators sharing a common interest which manifests itself as a set of Web pages. Though many communities are explicitly defined (newsgroups, resource collections in portals, etc.), many more are implicit. By using a subgraph-enumeration technique called *trawling*, we discover fine-grained communities numbering in the hundreds of thousands — a number substantially larger than the number of topics in portals and newsgroups. The following communities we have extracted from the Web in this way may underscore the point: the community of people interested in *Hekiru Shiina*, a Japanese pop singer; the community of people concerned with oil spills off the coast of Japan; and the community of Turkish student organizations in the U.S.

Identifying these communities helps not only in understanding the intellectual and sociological evolution of the Web but also in providing detailed information to a collection of people with certain focused interests. Owing to their astronomical number, embryonic nature, and evolutionary flux, it is hard to track and find such communities using sheer manual effort. Our approach to uncovering communities can be summarized as follows: We treat the Web as a huge directed graph, use graph structures derived from the basic hub-authority linkage pattern as the "signature" of a community, and systematically scan the Web graph to locate such structures.

The approach begins from the picture discussed earlier, that thematically cohesive Web communities contain at their core a dense pattern of linkage from hubs to authorities. This ties the pages together in the link structure, despite the fact that hubs do not necessarily link to hubs, and authorities do not necessarily link to authorities. It is our thesis that this pattern is a characteristic of both well-established and emergent communities. To put this into more graph-theoretic language, we use the notion of a *directed bipartite graph* — one whose nodes can be partitioned into two sets A and B such that every link in the graph is directed from a node in A to a node in B. Since the communities we seek contain directed bipartite graphs with a large density of edges, we expect many of them to contain smaller bipartite subgraphs that are in fact *complete*: each node in A has a link to each node in B.

Using a variety of pruning algorithms [3], we can enumerate all such complete bipartite subgraphs on the Web on a standard desktop PC in about 3 days of running time. In our experiments to date, we have used an 18-month old crawl of the Web provided to us by `Alexa (www.alexa.com)`, a company that archives snapshots of the Web. The process yielded, for instance, about 130,000 complete bipartite graphs in which 3 Web pages all pointed to the same set of 3 other Web pages. Were these linkage patterns coincidental? Manual inspection of a random sample of about 400 communities suggested otherwise: fewer than 5% of the communities we discovered lacked an apparent unifying topic. These bipartite cliques could then be fed to the algorithms of Sections 2 and 3, which "expanded" them to many more Web pages from the same community. Moreover, about 25% of the communities were not represented in Yahoo!, even today. Of those that do appear in Yahoo!, many appear at as deep as the sixth level in the Yahoo! topic tree. These observations lead us to believe that trawling a current copy of the Web will result in the discovery of many more communities that will become explicitly recognized in the future.

3 Combining content with link information

In the previous section, we discussed some of the advantages of relying extensively on links in searching for authoritative pages. Ignoring textual content after assembling the root set does, however, can lead to difficulties arising from certain features of the Web that deviate from the pure hub/authority view:

(1) On narrowly-focused topics, HITS frequently returns good resources for a more general topic. For instance, the Web does not contain many resources for skiing in Nebraska; a query on this topic will typically generalize to Nebraska tourist information.

(2) Since all the links out of a hub page propagate the same weight, HITS sometimes drifts when hubs discuss multiple topics. For instance a chemist's home page may contain good links to chemistry resources, as well as to resources in her hobbies, as well as regional information on the town where she lives. In such cases, HITS will confer some of the "chemistry" authority onto authorities for her hobbies and her town, deeming these to be authoritative pages for chemistry. How can we combat such drift?

(3) Frequently, a large number of pages from a single Web site will take over a topic simply because many of these occur in the base set. Moreover, pages from the same site often use the same html design template, so that (in addition to the information they give on the query topic) they may all point to a single popular site that has little to do with the query topic. The result can be that such a site gains too large a share of the authority weight for the topic, regardless of its relevance.

The Clever system [4, 5] addresses these issues by replacing the sums of Equations (1) and (2) with *weighted* sums, assigning to each link a non-negative weight. This weight depends on the query terms and the end-points of the link in a number of ways that we now briefly discuss. Together with some additional heuristics that we also discuss, they mitigate the problems mentioned above.

The text that surrounds hyperlink definitions (href's) in Web pages is often referred to as *anchor text*. The idea of using anchor text in our setting, to weight the links along which authority is propagated, is based on the following observation: when we seek authoritative pages on chemistry, for instance, we might reasonably expect to find the term "chemistry" in the vicinity of the *tails* — or anchors — of the links pointing to authoritative chemistry pages. To this end, we boost the weights of links in whose anchor — a window of a fixed width — query terms occur.

A second heuristic is based on breaking large hub pages into smaller units. On a page containing a large number of links, it is likely that all the links do not focus on a single topic. In such situations it becomes advantageous to treat contiguous subsets of links as mini-hubs, or *pagelets*; we may develop a hub score for each pagelet, down to the level of single links. The thesis is that contiguous sets of links on a hub page are more focused on a single topic than the entire page. For instance a page may be a good hub for the general topic of "cars", but different portions of it may cater to the topics of "vintage cars" and "solar-powered cars".

We mention one other set of modifications to the basic HITS algorithm. Recall that HITS deletes all links between two pages within the same Web domain. We are now working

with weighted links, and so we can address this issue through our choice of weights. First, links within a common domain are given low weight, following as above the rationale that authority should generally be conferred "globally" rather than from a local source on the same domain. Second, when a large number of pages from a single domain participate as hubs, it is useful to scale down their weights so as to prevent a single site from becoming dominant.

Interestingly, it is not hard to implement all of these heuristics without significantly altering the mathematics of Equations (1–4). The sums become weighted sums, and the matrix A now has non-negative real-valued entries rather than just 0's and 1's. As before, the hub and authority scores converge to the components of principal eigenvectors of AA^T and A^TA, respectively. In our experience, the relative values of the large components in these vectors typically resolve themselves after about 5 iterations of power iteration, obviating the need for more sophisticated eigenvector computation methods.

How do the resources computed by Clever compare with those found by other methods? We have conducted a number of user studies in which we compare Clever's compilations with those of AltaVista[1] (a term-index engine), Yahoo![2] (a manually compiled topic taxonomy in which a team of human ontologists create resource lists) and Infoseek[3] (generally believed to be some combination of the above). We now summarize the results of one such study [5] comparing Clever with Yahoo! and Altavista.

In this study, we began with a list of 26 broad search topics. For each topic, we took the top ten pages from Altavista, the top five hubs and five authorities returned by Clever, and a random set of ten pages from the most relevant node or nodes of Yahoo![4] We then interleaved these three sets into a single topic list, without an indication of which method produced which page. A collection of 37 users was assembled; the users were required to be familiar with the use of a Web browser, but were not experts in computer science or in the 26 search topics. The users were then asked to rank the pages they visited from the topic lists as "bad," "fair," "good," or "fantastic," in terms of their utility in learning about the topic. This yielded 1369 responses in all, which were then used to assess the relative quality of Clever, Yahoo!, and AltaVista on each topic. For approximately 31% of the topics, the evaluations of Yahoo! and Clever were equivalent to within a threshold of statistical significance; for approximately 50% Clever was evaluated higher; and for the remaining 19% Yahoo! was evaluated higher.

Note that in masking the source (Clever or Yahoo! or Altavista) from which each page was drawn, this experiment denied Yahoo! of one of the clear advantages of a manually compiled topic list: the editorial annotations and one-line summaries that are powerful cues (in deciding which link to follow). This choice was deliberate — we sought to isolate and study the power of different paradigms for resource finding, rather than for the combined task of compilation *and* presentation. In an earlier study [4] we did not mask these annotations, and Yahoo!'s combination of links and presentation beat an early version of Clever.

[1] www.altavista.com

[2] `www.yahoo.com`

[3] www.infoseek.com

[4] Yahoo! lists pages alphabetically and performs no ranking, hence the random choice.

The semi-automatic construction of taxonomies

Yahoo!, as mentioned above, is a large *taxonomy* of topics: it consists of a tree of subjects, each node of which corresponds to a particular subject and is *populated* by relevant pages. The results discussed above suggest that Clever can be used to compile such large taxonomies of topics automatically; we now explore this theme in more detail. Suppose that we are given a tree of topics, designed perhaps by a domain experts; the tree may be specified by its topology and the labels on its nodes. We wish to populate each node of the tree with a collection of the best hubs and authorities we can find on the Web. The following paradigm emerges: if we can effectively describe each node of the tree (i.e., each topic) as a query to Clever, the Clever engine could then populate the node as often as we please. For instance, the resources at each node could be refreshed on a nightly basis following the one-time human effort of describing the topics to Clever. How, then, should we describe a topic/node to Clever?

In the simplest form, we may take the name or label of the node as a query term input to Clever. More generally, we may wish to use the descriptions of other nodes on the path to the root. For instance, if the topic headings along a root-to-leaf path are *Business/Real Estate/Regional/United States/Oregon*, the query "Oregon" is not accurate; we might prefer instead the query "Oregon real estate".

Additionally, we may provide some exemplary authority or hub pages for the topic. For instance, `www.att.com` and `www.sprint.com` may be deemed exemplary authority pages for the topic "North American telecommunications companies". In practice, we envision a taxonomy administrator first trying a simple text query to Clever. In many cases this yields a good collection of resources, but in some others Clever may return a mix of high-quality and irrelevant pages. In such cases, the taxonomy administrator may highlight some of the high-quality pages in the Clever results as exemplary hubs, exemplary authorities, or both. This is akin to the well-studied technique of *relevance feedback* in information retrieval.

We take advantage of exemplary pages through the following link-based approach. An exemplary hub that is supplied is added to the base set, along with all pages that it points to; the weights of the links emanating from the exemplary hub are increased in the iterative computation. The treatment for exemplary authorities is similar, except that instead of adding to the base set any page pointing to an exemplary authority (a heuristic found to pull in too many irrelevant pages), we add in any page pointing to *at least two* exemplary authorities. A similar heuristic is used to delete user-designated "stop-sites" and their link neighborhoods from the base set. This is typically necessary because of the overwhelming Web presence of certain topics. For instance, if our topic is *Building and Construction Supplies/Doors and Windows*, it is difficult to focus away from Microsoft. Stop-siting `www.microsoft.com` takes care of this issue.

Thus, we may envision a topic node being described to Clever as a combination of query terms, exemplified authority and hub pages, and possibly stop-sites. We have developed a Java-based graphical user interface for administering such descriptions of taxonomies, called TaxMan (for Taxonomy Manager). Using this tool, we have been able to construct taxonomies with over a thousand topics. We have benchmarked both the time spent in creating these taxonomies and the quality of the results of using simple text-only queries versus a combination of text queries and exemplified Web pages. In our study, we found that

the average time spent per node grows from about 7 seconds to roughly three minutes when one moves to a combination of text and exemplary page queries. The increase in quality can be quantified as follows: outside users considered about 8% more of the pages generated using exemplaries to be good pages compared to the pages generated by textual queries.

Assigning Web Pages to Categories

In addition to their use as a means for finding hubs, authorities, and communities, hyperlinks can be used for categorizing Web pages. Categorization is a process by which a system learns (from examples) to assign documents to a set of predefined topic categories such as those found in a taxonomy. Hyperlinks contain high-quality semantic clues as to the topic of a page that are lost by a purely term-based categorizer. It is challenging to exploit this link information, however, since it is highly noisy; indeed, we have found that naive use of terms in the link neighborhood of a document can even *degrade* accuracy.

An approach to this problem is embodied in a system called HyperClass [6], which makes use of robust statistical models such as Markov random fields (MRF's) together with a relaxation labeling technique. Using this approach, it obtains improved categorization accuracy by exploiting link information in the neighborhood around a document. The use of the MRF framework derives from the simple observation that pages on the same or related topics tend to be linked more frequently than those on unrelated topics. Even if none of the categories of the linked pages are known initially, significant improvement can be obtained using relaxation labeling, wherein the category labels of the linked pages and of the page to be categorized are iteratively adjusted until the most probable configuration of class labels is found. Experiments were performed [6] using pre-classified samples from Yahoo! and the US Patent Database (`www.ibm.com/patents`). Using HyperClass with hyperlinks cut the patent error rate by half and the Yahoo! (Web documents) error rate by two thirds.

HyperClass is also used in a *focused* Web crawler[7], which is designed to search the Web for only pages on a particular topic or set of topics. By categorizing pages as it crawls, the focused crawler is able not just to filter out irrelevant pages; it also uses the associated relevance judgment, as well as a rank determined by a version of the Clever algorithm, to set the crawling priority of the outlinks of the pages it finds.

4 Conclusion

The mining of WWW link structures has intellectual antecedents in the study of social networks and citation analysis [8]. The field of citation analysis has developed a number of link-based measures of the importance of scholarly papers, including the *impact factor* and *influence weights* [8]. These measures in effect identify "authoritative" sources without introducing a notion of "hubs." The view of hubs and authorities as dual sets of important documents is inspired by the apparent nature of content creation on the Web, and indicates some of the deep contrasts that exist between content on the WWW and content in the scholarly literature.

The methodology of *influence weights* from citation analysis is related to a link-based search method due to Brin and Page [9], forming the basis of the *Google* search engine on the Web. Brin and Page first compute a score, which they call the *PageRank*, for every page that they index. The score for each page is the corresponding component of the principal eigenvector of a matrix B, which can be viewed as the adjacency matrix A with a very small constant added to each entry (recall Section 2). Given a query, they return pages containing the query term(s), ranked in order of the PageRanks of these pages. (The actual implementation of Google incorporates a number of additional heuristics, similar in intent and spirit to those used for deriving Clever from HITS.)

It is worth drawing some contrasts between Clever and Google. Google focuses on authoritative pages, while Clever seeks good hub pages as well. Note that these hub pages may have few (or no) links into them, so that they would end up with low PageRank scores and seldom be reported by Google. A number of participants in our user studies suggested that good hubs are especially useful when the user is trying to learn about a new topic, but less so when seeking a very specific piece of information. Another way in which the two methods differ in their behavior is especially apparent in topics with a commercial theme. A company may describe itself (on its Web pages) using terms and language that are different from the way a user embarking on a Web search might. Thus, a direct search for "mainframes" would not return IBM's home page (which does not contain the term "mainframes"); but IBM would in fact be pulled in by Clever because there are hub pages describing IBM as a mainframe manufacturer.

In independent work, Bharat and Henzinger [10] have given a number of other extensions to the basic HITS algorithm, substantiating the improvements via a user study. For instance, their paper was the first to describe the modification in which the weights of multiple links from within a site are scaled down (see Section 3).

Our analysis of hyperlink topology has focused on the extraction of densely connected regions in the link structure — hubs, authorities, and communities on a common topic — and it has made of use of techniques from linear algebra and subgraph enumeration. The paper of Karypis et al. in this issue deals with different types of problems that arise in the analysis of link structures — specifically, the partitioning of such structures into sparsely inter-connected pieces — and it approaches this through an interesting combinatorial technique.

We believe the mining of WWW link topology has the potential for beneficial overlap with a number of areas. One of these is the field of information retrieval [11]. Another is the mining of well-structured relational data. It is a considerable challenge to extract structure, of the kind that succumbs to traditional database techniques, from an unstructured medium

such as the Web (see [12]), and we hope that the techniques described here represent a step in the direction of this general goal.

References

[1] J.M. Kleinberg. Authoritative sources in a hyperlinked environment. Proc. 9th ACM-SIAM Symposium on Discrete Algorithms, 1998 and IBM Research Report RJ 10076, May 1997. To appear in Journal of the ACM.

[2] G. Golub, C.F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, 1989.

[3] S. R. Kumar, P. Raghavan, S. Rajagopalan and A. Tomkins. Trawling emerging cyber-communities automatically. Proceedings of the 8th World Wide Web conference, 1999.

[4] S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, P. Raghavan, and S. Rajagopalan. Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text. Proceedings of the 7th World Wide Web conference, 1998. Elsevier Sciences, Amsterdam.

[5] S. Chakrabarti, B. Dom, Ravi Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Experiments in Topic Distillation. SIGIR workshop on hypertext information retrieval, 1998.

[6] S. Chakrabarti and B. Dom and P. Indyk, Enhanced hypertext classification using hyperlinks. ACM SIGMOD Conference on Management of Data, 1998.

[7] S. Chakrabarti and B. Dom, and M. van den Berg. Focused Crawling: A New Approach for Topic-Specific Resource Discovery. Proceedings of the 8th World Wide Web conference, 1999.

[8] L. Egghe, R. Rousseau. *Introduction to Informetrics.* Elsevier, 1990.

[9] S. Brin, and L. Page. The anatomy of a large scale hypertextual Web search engine. In Proceedings of WWW7, Brisbane, Australia, April, 1998.

[10] K. Bharat and M.R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. Proceedings of ACM SIGIR, 1998.

[11] G. Salton and M. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.

[12] D. Florescu, A. Levy and A. Mendelzon. Database Techniques for the World Wide Web: A Survey. SIGMOD Record 27(3): 59-74 (1998).