Information Retrieval on the Web

Kiduk Yang
Indiana University

Introduction

How do we find information on the Web? Although information on the Web is distributed and decentralized, the WWW can be viewed as a single, virtual document collection. In that regard, the fundamental questions and approaches of traditional information retrieval (IR) research (e.g., term weighting, query expansion) are likely to be relevant in Web document retrieval[1]. Findings from traditional IR research, however, may not always be applicable in a Web setting. The Web document collection–massive in size and diverse in content, format, purpose, and quality–challenges the validity of previous research findings that are based on relatively small and homogeneous test collections. Moreover, some traditional IR approaches, while applicable in theory, may be impossible or impractical to implement in a Web setting. For instance, the size, distribution, and dynamic nature of Web information make it extremely difficult to construct a complete and up-to-date data representation of the kind required for a model IR system.

To further complicate matters, information seeking on the Web is diverse in character and unpredictable in nature. Web searchers come from all walks of life and are motivated by all kinds of information need. The wide range of experience, knowledge, motivation and purpose means that searchers can express diverse types of information need in a wide variety of ways with differing criteria for satisfying their needs. Conventional evaluation measures, such as precision and recall, may no longer be appropriate for Web IR, where a test collection

[1] For the remainder of this paper, IR will imply text-based retrieval unless otherwise stated.

representative of dynamic and diverse Web data is all but impossible to construct.

Finding information on the Web creates many new challenges for, and exacerbates some old problems in, IR research. At the same time, the Web is rich in new types of information not present in most previous IR test collections. Hyperlinks, usage statistics, document markup tags, and bodies of topic hierarchy such as Yahoo! (http://www.yahoo.com) present an opportunity to leverage Web-specific document characteristics in novel ways that go beyond the term-based retrieval framework of traditional IR. Consequently, researchers in Web IR have reexamined the findings from traditional IR research to discover which conventional text retrieval approaches may be applicable in Web settings, while exploring new approaches that can accommodate Web-specific characteristics.

Web data still consist mostly of text, which means that various text retrieval tools (e.g., term weighting, term similarity computation, query expansion) may be useful in Web IR. As the name implies, Web documents are heavily interconnected. Link analysis approaches, such as PageRank, HITS, and CLEVER have created hyperlinks as implicit recommendations about the documents to which they point (Chakrabarti et al., 1998b; Kleinberg, 1998; Page et al., 1998). Some researchers (Pirolli, Pitkow, & Rao, 1996; Schapira, 1999), as well as commercial search services (e.g., DirectHit[2], Alexa: http://www.alexa.com) have also looked at ways to leverage Web user statistics (e.g., counting link clicks, page browsing time). Another approach popularized by Web search services, though less explored in research, is the organization of quality-filtered Web data into a topic hierarchy. Due to the volume and diversity of data, traditional automatic classification approaches do not seem to fare well in a Web setting, so most such topic hierarchies are constructed manually (e.g., Yahoo!).

---

[2] DirectHit, a "popularity" search technology that powers other search services, no longer maintains its own site. See http://www.searchenginewatch.com/sereport/article.php/2164521 for details.

Two of the most promising Web IR tools, namely Google and CLEVER, seem to be using combinations of these retrieval techniques identified. CLEVER combines topic-dependent link analysis techniques called HITS with term similarity techniques from text retrieval, while Google seems to employ a smorgasbord of techniques to obtain high performance text retrieval influenced by a universal link analysis score called PageRank. Both Google and CLEVER not only leverage heavily off the implicit human judgment embedded in hyperlinks, but also improve on link analysis by combining it with text retrieval techniques.

By way of contrast, there seems to be a shortage of techniques that utilize the considerable body of explicit human judgment (e.g., Web directories[3]) in combination with hyperlinks and text contents of Web documents. Most IR research dealing with knowledge organization focuses on automatic clustering and classification of documents. There is little research that investigates how hierarchical knowledge bases like Yahoo! can be integrated with text retrieval and link analysis techniques in Web IR.

Scope

Information discovery on the Web is challenging. The complexity and richness of the Web search environment call for approaches that extend conventional IR methods to leverage rich sources of information on the Web. We first review findings from research that investigates characteristics of the Web search environment; then we examine key ideas and approaches in Web IR research. Although the emphasis in this chapter is on Web retrieval strategies, a review of research that explores the classification of Web documents is included to highlight the importance of the knowledge organization approach to information discovery. We also examine

[3] Web directories are manually constructed topical taxonomies of Web documents (e.g., Yahoo!).

research that combines traditional and Web-specific IR methods.

Sources of Information

Web IR is an active research area with participants from a variety of disciplines, such as computer science, library and information science, and human-computer interaction. Research results are published in a wide range of journals, such as *Information Processing & Management* and *the Journal of the American Society for Information Science and Technology*, as well as many conference proceedings, such as the International World Wide Web Conference (IW3C), the Association for Computing Machinery (ACM) conferences, and the Text Retrieval Conference (TREC). A wide spectrum of research in Web IR can be found in a variety of ACM conferences such as the Annual International Special Interest Group on Information Retrieval (SIGIR) conference on Research and Development in Information Retrieval and the Hypertext and Hypermedia (HT) Conference, while the TREC conference provides a common ground where cutting-edge Web IR approaches are investigated in a standardized environment.

In addition to individual research findings, a number of overviews of Web IR issues have been published. Two of the most recent work are the *ARIST* chapter by Rasmussen (2003) on indexing and retrieval from the Web, which reviews research on indexing and ranking functions of search engines on the Web, and another *ARIST* chapter by Bar-Ilan (2004) on the use of Web search engines in information science research, which examines research into the design and the use of search engines from both social and application-centered perspectives.

Research in Web IR

The main focus of early IR research had been on the development of retrieval strategies

for relatively small, static, and homogeneous text corpora. The Web, however, contains massive amounts of dynamic, heterogeneous, and hyperlinked information. Furthermore, information seeking on the Web is much more diverse and unpredictable than in traditional IR contexts. Consequently, determination of the applicability of traditional IR methods to Web IR should begin with a consideration of the contextual differences between traditional IR and Web IR research. The findings of traditional IR are mostly based on experiments conducted with small, stable document collections and sets of relatively descriptive and specific queries. Web IR, on the other hand, must deal with mostly short and unfocused queries posed against a massive collection of heterogeneous, hyperlinked documents that change dynamically. One focus of Web IR research has thus been the characteristics of the Web search environment (notably, document characteristics and searcher behavior).

Characteristics of the Web Search Environment

Studies of the Web search environment are of two main types: those that attempt to characterize the Web via content and structural analysis of a sample set of documents obtained by a Web crawler (Bray, 1996; Broder et al., 2000; Lawrence & Giles, 1998, 1999a, 1999b; Woodruff et al., 1996), and those that attempt to characterize searcher behavior via user surveys (Kehoe et al., 1999) or analysis of search engine logs (Jansen et al., 1998; Jansen, Spink, & Saracevic, 1998, 2000; Silverstein et al., 1998; Spink et al., 2001).

Early studies of Web characteristics were based on relatively small samples and were therefore limited in scope. Moreover, the Web has been growing and changing at a significant rate since its inception, so findings from early studies are outdated and unrepresentative of the current search environment. Tracking Web characterization research over time, however, might

give us insight into the evolution of the Web and thus lead to a better understanding of its underlying nature.

Characteristics of Web Documents

One of the first large scale Web studies was conducted by a Berkeley research project group called Inktomi[4] (Woodruff et al., 1996), who examined various characteristics of Web documents obtained by the Inktomi crawler in 1995. The study reported the usage pattern of HTML tags as well as various statistics on Web documents, such as the average document size after HTML tag extraction (4.4 Kbytes), the average number of tags per document (71), and the average number of unique tags per document (11). According to the study, the most prevalent HTML tag was the title tag (used by 92% of documents), while the most frequent attribute was HREF (used by 88% of documents; on average 14 times per document). The study also observed rapid changes in the Web by comparing the Web crawl from July to October 1995 (1.3 million unique HTML documents) with that of November 1995 (2.6 million unique HTML documents). There was a doubling in size in one month and many of the most popular documents (i.e., documents with highest the indegree[5]) in the first crawl disappeared in the second crawl.

In a related study, Bray (1996) examined the content and structure of 1.5 million Web documents in 1995. In addition to the report of document statistics (i.e., document size, tag count, etc.), which were comparable to those of the Inktomi group, Bray characterized Web documents by their connectivity using measures called visibility (i.e., indegree) and luminosity (i.e., outdegree). The most visible documents in Bray's sample consisted of homepages of well-

[4] Inktomi, which began as a UC Berkeley research project in 1995, is currently a Yahoo! subsidiary that provides Web search products and services. See http://www.inktomi.com for more details about the company.

[5] Indegree of a document *p* denotes the number of inlinks of *p*, where an inlink of *p* denotes a document *q* that points (i.e., links) to *p*. Conversely, outdegree of a document *p* denotes the number of outlinks of *p*, where an outlink of *p* denotes a document *q* that is pointed (i.e., linked) by *p*.

known universities, organizations, and companies, while top luminous sites were dominated by Web indexes such as Yahoo!. Bray also made the observation that the majority (80%) of documents in his sample did not contain any outlinks to external sites and had only "a few" (1 to 10) inlinks from external sites[6]. This implies that the Web is connected together by a relatively few hubs (i.e., documents with high outdegree).

Despite Bray's observation about the irregular connectivity pattern of the Web, the impression of the Web as a network of densely interconnected communities persisted, not least with proponents of link-based search strategies (Kleinberg, 1998; Page et al., 1998). However, findings from a more recent study of Web structure, based on three sets of experiments between May 1999 and October 1999 with two AltaVista crawls of over 200 million pages and 1.5 billion links, seem to validate Bray's picture of the Web (Broder et al., 2000). Broder's study found that only about 28% of the Web pages[7] are "strongly connected" and it takes an average of 16 link traversals to move from one strongly connected site to another, which suggests that the Web may not be as well connected as previously thought (Barabasi, 2003). If such findings turn out to be true for the whole Web, it would pose yet another challenge for Web search engines, which collect most of their data by following hyperlinks. Incidentally, Broder's study also verified an earlier observation regarding the power law phenomenon of indegree (Kumar et al., 1999), which estimated the probability that a page has in-degree k to be roughly $1/k^2$.

Findings from Broder's study, which are based on the graph analysis of the Web link structure, are supplemented by Lawrence and Giles (1998, 1999a, 1999b), who conducted a series of studies analyzing Web content. According to their most recent study (Lawrence &

---

[6] Bray employed a number of ad-hoc rules to define a site, which was based on parsing of URLs with an intent to identify the logical location of documents.

[7] In this paper, the terms "Web documents" and "Web pages" are loosely interchangeable, though the former mostly implies the text content while the latter references both textual and non-textual contents.

Giles, 1999a, 1999b), the estimated size of the publicly indexable Web[8] as of February 1999 was 15 terabytes of data consisting of 800 million pages scattered over 3 million Web servers; this has more than doubled their earlier estimate of 320 million pages in December 1997 (Lawrence & Giles, 1998).

According to the Internet Archive Project (http://www.archive.org/), which has been building a digital library of the Internet by compiling snapshots of publicly accessible Internet sites since 1996 (Kahl, 1997), the Web had reached 1 billion pages as of March 2000 with page content changing at the rate of 15% per month. Given the accelerated growth of the Web and its formidable size of estimated 4 billion indexable pages (Glover et al., 2002), it is not surprising that an up-to-date study of Web document characteristics is lacking.

Information seeking on the Web

Thus far, we have considered the characteristics of Web documents, which is only one aspect of the Web search environment. Information seeking is the other, no less crucial than document collection in its influence on the retrieval process. There is a considerable body of literature on information seeking, but a general review is beyond the scope of this chapter. Instead, we focus on the studies of search patterns and user characteristics on the Web.

According to the most recent user survey by the Graphics, Visualization & Usability (GVU) Center at Georgia Tech (Kehoe et al., 1999), most Web users access the Internet on a daily basis (79%) and use search engines to find information (85%). The survey is now several years old, but the trend appears to be continuing, as evidenced by the 670 million daily searches

[8] "publicly indexable" Web includes only those normally indexed by Web search engines, thus excluding such pages as those specified by the robot exclusion standard, firewall or password-protected pages, and hidden pages (e.g., cgi, dynamic pages).

logged by seven major Web search engines (250 million by Google)[9].

User studies, such as GVU survey, can provide demographic data about the searcher. In order to study patterns of search behavior, however, researchers have examined system transaction logs that record actual search sessions. Finding from one of the first large scale studies analyzing Excite search engine log[10] suggested that Web search behavior patterns are rather different from those assumed in traditional IR contexts (Jansen, Spink, Bateman, & Saracevic, 1998). By examining the content and use of queries (e.g., search terms, use of search features), the study found that Web searchers tend to use very short queries and exert minimum effort in evaluating or refining their searches. About 30% of queries in the study were single term queries, with an average query length of 2.35 terms. The study also reported that most searchers browsed only the first page of search results and did not engage in search refinement processes such as relevance feedback or query reformulation. In a related study using the same data, Jansen, Spink and Saracevic (1998) conducted a failure analysis of queries (i.e., incorrect query construction) and found that Web searchers not only used advanced query features sparingly, but also tended to use them incorrectly.

A much larger study of Web query patterns, analyzing 280 gigabytes of AltaVista query logs collected over a period of 43 days, was conducted by Silverstein et al. (1998). The data used in this study consisted of 1 billion entries in AltaVista transaction logs from August 2, 1998, to September 13, 1998, and thus promised to be less likely to be affected by short-term query trends than might be the case with smaller snapshots of query data. The findings were consistent with earlier studies in that Web searchers use short queries, mostly look at the first few results only,

---

[9] Data as of February, 2003, published in Search Engine Watch (http://www.searchenginewatch.com/reports/article.php/2156461)

[10] The data, which were a random subset of Excite searches on March 10, 1997, consisted of 51,473 queries from 18,113 users.

and seldom modify the query. The study reported that 77% of all search sessions contained only one query, and that 85% of the time only the top 10 results were examined. The searchers' relatively low investment in their search process was further evidenced in the short average query length of 2.35 words, most of which were intended to be phrases. Except for a few highly common queries (mostly sex-related), over two thirds of all queries were unique, implying that information needs on the Web are highly diverse (or at least expressed in diverse ways).

Wolfram et al. (2001) investigated trends in Web search behavior by comparing findings from two follow-up studies of Excite search logs in 1997 (Jansen, Spink, & Saracevic, 2000) and 1999 (Spink, Wolfram, Jansen, & Saracevic, 2001), each of which contained over 1 million queries from over 200,000 Excite users. They found that Web searchers tended to invest little effort in both query formulation and result evaluation, which reconfirmed patterns reported in previous studies.

Large-scale studies of search engine logs describe characteristics of an average searcher on the Web but do not provide detailed information on individual searchers. Though small in scale, and thus more anecdotal than conclusive, studies that examine the characteristics of searchers in context are nevertheless illuminating. For example, Pollock and Hockley (1997) found that Internet-naïve users had a poor grasp of the concept of iterative searching and relevance ranking. They had difficulty formulating good queries, either because they did not understand what were likely to be good quality differentiators or because they did not realize that contextual terms should be included. At the same time, they expected results to be clear and organized, and considered anything less than a perfect match to be an outright failure.

Hölscher and Strube (2000), who investigated the search behavior of expert and novice Web users, found that expert Web users exhibited more complex behaviors than average Web

searchers observed in previous studies. For example, they engaged in various search enhancement strategies, such as query reformulation, exploration of advanced search options, and combination of browsing and querying. In general, they seemed to be much more willing to go the extra mile to satisfy their information needs, as further evidenced by the longer query (average 3.64 words) and evaluation of more search results. Furthermore, Hölscher and Strube found that searchers with high domain knowledge used a variety of terminology in their queries and took less time evaluating documents than their counterparts, who often got stuck in unsuccessful query reformulation cycles resulting from ineffective query modifications.

The profile of the average Web searcher that emerged from the Excite and AltaVista studies (Jansen et al., 1998; Jansen, Spink, & Saracevic, 2000; Silverstein et al., 1998; Spink et al., 2001; Wolfram et al., 2001) seems to be consistent with assumptions about information seeking in electronic environments described by Marchionini (1992). Web searchers in general do not want to engage in an involved retrieval process. Instead, they expect immediate answers while expending minimum effort. Web or domain experts, however, tend to engage in more elaborate efforts to satisfy their information needs. This difference in the degree of involvement with the retrieval process is in all likelihood influenced by the level of search skill and domain knowledge. These help reduce the cognitive load required for engaging in various steps of the retrieval process. The important point here is not how expert and novice searchers differ, but why. If we believe that humans seek the path of least cognitive resistance, then reducing the cognitive load for users should be one of the primary goals of IR system design. Marchionini (1992) describes an appropriate information system as one that combines and integrates the information seeking functions to help users clarify their problems and find solutions.

Information seeking on the Web not only deviates from traditional IR expectations of

searcher behavior, but also encompasses a wider range of information need than previously investigated. In his exploration of Web search taxonomy, Broder (2002) discovered that information sought on the Web was often not informational. In fact, Broder's analysis of AltaVista user surveys and search logs[11], which revealed three broad types of Web search, found informational queries to account for slightly less than 50% of all searches, followed by transactional queries (about 30%), whose objective is to engage in transactional interaction (e.g., shopping, downloading), and navigational queries (about 20%), which are used to reach a particular site (e.g., a homepage).

Detailed examination of the Web highlights some of the potential problems with applying traditional IR approaches. The fact that it is practically impossible for any one search engine to construct a comprehensive and current index due to the Web's massive size and dynamic nature coupled with the fact that there is relatively small overlap in search engines' coverage points to the potential advantage of combining retrieval results from multiple search engines. The diverse nature of both documents and information needs on the Web hints at the need for specialized search systems that cater to specific situations, or flexible systems that can respond appropriately to a variety of situations. The evidence of minimalist approaches to information seeking on the Web suggests the need for support features that can shift the cognitive burden from the user to the system.

We need to go beyond conventional approaches to IR and devise new techniques that fit the Web environment, while adopting established techniques as appropriate. One approach is the use of Web-specific features, such as HTML and link structure (Google) or usage statistics (Direct Hit) to identify relevant documents. Another example is the use of manually constructed

[11] The survey data consisted of 3,190 Web submissions and the search log data consisted of 400 random queries by AltaVista users in 2001.

hierarchical categories like Yahoo!, where the application of age-old cataloging principles allows users to find high quality information by browsing a predefined topical taxonomy without having to explicitly formulate their information need in the language of the search systems. Both of these approaches–the use of topical hierarchies and leveraging of Web-specific information–are important aspects of Web IR.

Web IR approaches

A major focus of IR research is on developing strategies for identifying documents "relevant" to a given query. In traditional IR, evidence of relevance is typically mined from information in the text. Traditional IR is based on ranking documents according to their estimated degree of relevance using such measures as term similarity or term occurrence probability. On the Web, however, information can reside beyond the textual content of documents. For example, it is relatively easy to collect Web document metadata such as usage statistics and file characteristics (e.g., size, date), which can be used to supplement term-based estimation of document relevance. Hyperlinks, being the most prominent source of evidence in Web documents by far, have been the subject of numerous studies exploring retrieval strategies based on link exploitation.

The advent of link-based approaches predates the Web and can be traced back to citation analysis in the field of bibliometrics and to hypertext research (Borman & Furner, 2002). Two measures of document similarity based on citations were proposed in bibliometrics (White & McCain, 1989): bibliographic coupling (Kessler, 1963), which is the number of documents cited by both document *p* and *q*, and co-citation (Small, 1973), the number of documents that cite both *p* & *q*. One of the successful deployments of these measures was demonstrated by Shaw (1991a,

1991b), who used a combination of text similarity, bibliographic coupling, and co-citation as part of a graph-based clustering algorithm to improve retrieval performance.

In hypertext research, both citations and hyperlinks have been used for clustering and searching. Rivlin, Botafogo and Shneiderman (1992, 1994) used connectivity and compactness measures based on node distance to identify clusters as well as using link analysis to enhance relevance ranking. Weiss et al. (1996) defined similarity measures based on link structure, generalized from co-citation and bibliographic coupling to allow long chains of reference. Using citations or links to cluster documents is closely related to conventional clustering approaches that group together documents with similar content. Instead of identifying related documents (Willett, 1988), passages (Salton & Buckley, 1991), or terms (Sparck Jones, 1971) based on their textual similarity, however, link-based clustering approaches are based on the assumption that documents cited/linked together many times (i.e., co-citation) or documents with many common citations/links (i.e., bibliographic coupling) are likely to be related.

Leveraging Hyperlinks

Using hyperlinks to enhance searching is also based on the notion that hyperlinks connect related documents and thus can provide additional information. Link-based retrieval strategies in hypertext research explore various methods of enriching local document contents with the contents of hyperlinked documents. In the context of hypertext, where the document collection usually consists of homogeneous documents on a single topic that are linked together for the purpose of citation, external content introduced by hyperlinks tends to be of high quality and useful. On the Web, however, where hyperlinks connect documents of varying quality and content for various purposes (Kim, 2000), document enrichments via hyperlink can sometimes

introduce noise and degrade the retrieval performance.

One way to address such problems is to categorize hyperlinks according to their purpose and usefulness so that they may be utilized appropriately. Discussions of hyperlink classification in the literature, however, have not moved beyond explicit link typing (Baron, 1996; Kopak, 1999; Shum, 1996; Trigg & Weiser, 1983) or rudimentary attempts at link type identification based on visualization of text similarity relationships between documents (Allan, 1996; Salton, Buckley, & Allan, 1994). Allan's automatic link typing strategy (1996), for example, involves determination of the density and pattern of similarity between document subparts (i.e., paragraphs), and may not be best suited for Web IR. It seems clear that there are various patterns in hypertext linking (Bernstein, 1998), but whether those patterns can be automatically identified for effective retrieval remains to be seen.

Although hypertext approaches look beyond the boundary of immediate document content, hypertext methods, as the name implies, still mine for evidence in the text of document neighborhoods. Web IR, on the other hand, goes beyond the textual content of the document corpus and leverages various sources of information such as link structure, usage patterns, and manually constructed topic hierarchies. Herein lies one of the fundamental differences between Web IR and traditional IR. While traditional IR depends solely on the text of documents, Web IR goes beyond the textual content and utilizes implicit human judgments about documents, whether embedded in hyperlinks, user statistics or Web directories.

One of the earliest attempts to adapt the traditional IR model was by Croft (1993), who demonstrated the successful incorporation of hypertext links into the Inference Network model. The Inference Network model is a probabilistic retrieval model based on a Bayesian inference network, where nodes represent documents and queries and directed edges represent dependence

relations between nodes. In a text-based implementation of the Inference Network model, the middle layer of nodes consists of terms whose dependencies to outer layer nodes (i.e., documents, query) are computed based on term occurrence probabilities. In a link-based implementation, where the existence of a hyperlink is taken as evidence of a dependency between linked documents, the hyperlink evidence is incorporated into the term-based network structure by adding to each document the dependence relations of new terms introduced by the linked documents and strengthening the dependence relations of existing terms shared by the linked documents. Croft's use of hyperlinks results in increased importance of terms contained in linked documents, which can be problematic if the quality of hyperlinks is poor.

A more discriminatory method of leveraging hyperlinks was proposed by Frei and Stieger (1995). Instead of blindly following all hyperlinks to obtain additional information about a document, Frei and Stieger annotated each link with a content-specific link description comprising of common terms between the source and the destination of a link. They followed only those links whose query-link description similarities were above some threshold. Each time a link is traversed, Frei and Stieger's algorithm updates the Retrieval Status Value (RSV), the sum of query-document similarity scores weighted by link distance with which retrieval results are ranked. Computation of RSV, as described in the equation below, is not only similar in essence to the link-based document enrichment strategy proposed by Marchiori (1997), but also resembles in form subsequent link-based algorithms that propagate information through hyperlinks:

$$RSV_{d+1} = RSV_d + w_d * RSV_{current}, \quad (1)$$

where $RSV_{current}$ is the sum of similarity between query and documents at distance d+1, and w_d is a propagation factor dependent on the navigation distance.

Marchiori describes a mathematical model, where the information propagated through hyperlinks is scored with a recursive formula based on exponentially fading textual content of the linked documents. Marchiori coins the term "hyper information" to denote the information provided by hyperlinks and suggests that hyper information could work on top of "local" textual information to produce a more "global" score for a Web document. Marchiori's *HyperSearch* algorithm scores a document's information content by adding to the textual information of a document *p* its hyper information, which is computed by summing up the textual information of documents reachable from *p* by recursively following outlinks, diminished by a damping factor that decays exponentially with link distance from *p*. If we denote hyper information of *p* with *HI*(*p*), textual information of *p* by *TI*(*p*), and the damping factor by *F* ($0<F<1$), HI(*p*) can be expressed as follows:

$$HI(p) = F*TI(p_1) + F^2*TI(p_2) + \ldots + F^k*TI(p_k) \qquad (2)$$

At the heart of this approach is the notion that a document can be enriched with the textual contents of linked documents. Since linked documents are often linked in turn, the computation becomes recursive, with the provision for fading information propagation based on the link distance. The informative content of a Web document should ideally involve all the linked documents, but Marchiori fixes an arbitrary link distance limit *k* in his model for reasons of practicality and uses the *HyperSearch* algorithm in practice to compute the relevance of a document to a given query by propagating the relevance scores of documents within link

distance *k*.

The idea of information propagation via links is extended by Page et al. (1998), who developed a method for assigning a universal rank to Web pages based on a weight-propagation algorithm called PageRank. Instead of propagating textual information backwards through outlinks of a fixed size neighborhood as in HyperSearch, PageRank propagates the PageRank scores forward through inlinks of the entire Web. This recursive definition of PageRank differs sharply from other link-based methods in that it arrives at a global measure of a Web page without taking into account any textual information. In other words, the PageRank score of a Web page is influenced by neither the page itself nor any potential query, but is based solely on the aggregate measure of human-judged importance implied in each hyperlink.

Page et al. start with the notion of counting backlinks (i.e., indegree) to assess the importance of a Web page, but point out that simple indegree does not always correspond to importance; thus they arrive at propagation of importance through links, where a page is important if the sum of the importance of its backlinks is high. This idea is captured in the PageRank formula as follows:

$$R(p) = d \cdot \frac{1}{T} + (1-d) \cdot \sum_{i=1}^{k} \frac{R(p_i)}{C(p_i)}, \qquad (3)$$

where T is the total number of pages on the Web, d is a damping factor, $C(p)$ is the outdegree of p, and p_i denotes the inlinks of p. $R(p)$ can be calculated iteratively, starting with all $R(p_i)$ values equal to 1 and repeating computations until the values converge. This calculation corresponds to

computing the principal eigenvector of the link matrix of the Web[12]. When PageRank is scaled so that $\sum R(p)=1$, it can be thought of as a probability distribution over Web pages. In that light, $R(p)$ can be interpreted as a weighting function that estimates the probability that a Web surfer will arrive at page p from some starting point by a series of forward link traversals as well as occasional jumps to random pages. By incorporating into the formula the damping factor d, which represents the probability that p will be arrived at randomly instead of via link traversal, PageRank models the behavior of the "random" Web surfer who walks the Web by following the hyperlinks for a finite amount of time before going on to something unrelated.

The underlying assumption of PageRank is the notion that a link from page p_i to page p signifies the recommendation of p by the author of p_i. By aggregating all such recommendations recursively over the entirety of the Web, where each recommendation is weighted by its importance and normalized by its outdegree, PageRank arrives at an objective measure of importance from subjective determinations of importance scattered over the Web. By the same token, PageRank can be said to measure a collective notion of importance, otherwise known as "authority" in Web IR. One may challenge the "conferred authority" assumption about links and argue that a linked page is popular rather than important or authoritative. Indeed, even the academic journal citations sometimes reflect an author's deference or preferential treatment towards the cited document rather than careful judgment (Cronin & Snyder, 1997). The debate surrounding popularity and authority is important, and it has been argued that the recursive and exhaustive nature of PageRank tends to favor authority over popularity since it is more likely that important pages will link to other important pages than popular pages to other popular pages.

There are several ways PageRank can be used in Web IR. It can be used as a part of a

[12] The $(i,j)^{th}$ entry of the link matrix corresponds to the link from page i to page j.

search engine's ranking mechanism to improve term-based retrieval results by giving preference to more important and central pages (Brin & Page, 1998). It can also be used to guide a Web crawler (Cho, Garcia-Molina, & Page, 1998), or help find representative pages for page clusters. When employed by a search engine, PageRank works well with broad queries that return large numbers of documents by identifying a few high quality (i.e., important, popular) documents. Since PageRank is a global measure based on collective opinions, its effectiveness depends largely on the coverage on which the computations are made. Google, for example, computes PageRank based on link analysis of almost 600 million pages (1 billion URLs), which is a large chunk of the publicly indexable Web.

Kleinberg's (1998) HITS (Hyperlink Induced Topic Search) algorithm considers both inlinks and outlinks to identify mutually reinforcing communities of "authority" and "hub" pages. HITS defines "authority" as a page that is pointed to by many good hubs and defines "hub" as a page that points to many good authorities. Mathematically, these circular definitions can be expressed as follows:

$$a(p) = \sum_{q \to p} h(q), \qquad (4)$$

$$h(p) = \sum_{p \to q} a(q). \qquad (5)$$

These equations define the authority weight *a(p)* and the hub weight *h(p)* for each page *p*, where $p \to q$ denotes "page *p* has a hyperlink to page *q*".

Though HITS embraces the link analysis assumption that equates a hyperlink with a human judgment conferring authority on the pages pointed to, it differs from other link-based

approaches in several regards. Instead of simply counting the number of links, HITS calculates the value of page *p* based on the aggregate values of pages that point to *p* or are pointed to by *p*, in a similar fashion as PageRank. HITS, however, differs from PageRank in three major regards. First, it takes into account the contributions from both inlinks and outlinks to compute two separate measures of a page's value, namely authority and hub scores, instead of a single measure of importance like PageRank. Second, HITS measures pages' values dynamically for each query, rather than assigning global scores regardless of the query. Third, HITS scores are computed from a relatively small subset rather than the totality of the Web.

Unique to HITS is the premise that the Web contains mutually reinforcing communities (i.e., hubs and authorities) on sufficiently broad topics. To identify these communities, HITS starts with a root set *S* of text-based search engine results in response to a query about some topic, expands *S* to a base set *T* with the inlinks and outlinks of *S*, eliminates links between pages with the same domain name in *T* to define the graph *G*, runs the iterative algorithm (equations 4 and 5) on *G* until convergence, and returns a set of documents with high *h(p)* weights (i.e., hubs) and another set with high *a(p)* weights (i.e., authorities). The iterative algorithm works as follows: Starting with all weights initialized to 1, each step of the iterative algorithm computes *h(p)* and *a(p)* for every page *p* in *T*, normalizes each of them so that the sum of the squares adds up to 1, and repeats until the weights stabilize. In fact it can be shown that the authority weights at convergence correspond to the principal eigenvalues of $\mathbf{A^T A}$ and hub weights correspond to those of $\mathbf{A A^T}$, where **A** is the link matrix of the base set *T*[13]. Typically, convergence occurs in 10 to 50 iterations for *T* consisting of about 5,000 Web pages, expanded from the root set *S* of 200

[13] The $(i,j)^{th}$ entry of A is 1 if there exists a link from page *i* to page *j*, and is 0 otherwise. In A^T, the transpose of the link matrix A, the $(i,j)^{th}$ entry of A corresponds to the link from page *j* to page *i*. The $(i,j)^{th}$ entry of AA^T gives the number of pages pointed to by both page *i* and page *j* (bibliometric coupling), while the $(i,j)^{th}$ entry of A^TA gives the number of pages that point to both page *i* and page *j* (co-citation).

pages while being constrained by the expansion limit of 50 inlinks per page.

The base set *T* often contains multiple distinct communities (i.e., sets of hubs and authorities), which turn out to be document clusters of sorts with different meanings (e.g., jaguar), different contexts (e.g., recall), standpoints (e.g., abortion), or simply varying degrees of "relevance" to the query topic. The most densely linked community, which is returned by the iterative algorithm of HITS, is called the principal community, while others are called non-principal communities. Non-principal communities, which can be identified by finding the non-principal eigenvectors of $\mathbf{A}^T\mathbf{A}$ and $\mathbf{A}\mathbf{A}^T$, can not only contain relevant documents when the principal community misses the mark, but also reveal interesting information about the fine structure of a Web community (Kumar et al., 1999).

In addition to finding a set of high quality pages on a topic as well as hubs pointing to many such pages, HITS can be used to find similar pages to a given page *p* by starting with a root set *S* of pages that point to *p* (as opposed to obtaining *S* by querying a search engine) and returning authority pages in the community to which *p* belongs. Of course, the target page *p* should have enough inlinks with which to construct *S* of sufficient size. The advantage of the HITS method of finding similar pages over text-based methods is that it can locate pages classified together by Web authors (i.e., hub creators) whether they have any text overlap or not.

Though HITS is query-dependent in a sense that it begins with a root set of documents returned by a search engine in response to a query, the textual contents of pages are only considered in the initial step of obtaining that root set, after which the algorithm simply propagates weight over links without regard to the relevance of pages to the original topic. In other words, once HITS locates a topic neighborhood, it is guided by the link structure alone. Consequently, HITS can drift away from relevant documents if there exists in the neighborhood

T a community of documents with a higher link density. This phenomenon is called "diffusion", and has been observed to occur most frequently in response to a specific query with a large, generalized topic presence, for which the algorithm converges to the community on the generalized topic instead of focusing on the original topic. The diffusion effect, also known as "topic drift" (Bharat & Henzinger, 1998) or the "Tightly-Knit Community (TKC)" effect (Lempel & Moran, 2000), is perhaps the most serious weakness of HITS. As we will see in the next section, several researchers have investigated ways to compensate for the diffusion effect by tempering link analysis with content analysis.

Enhancing Link Analysis with Content Analysis

Chakrabarti et al. (1998b) extended HITS in the ARC (Automatic Resource Compiler) of the CLEVER project by incorporating the text around links into the computation of hub and authority weights. The idea that the text surrounding the links pointing to *p* is descriptive of the content of *p* had been recognized for some time (McBryan, 1994), but fusing textual content with the link-based framework of HITS was an innovative step. The ARC algorithm is largely identical to HITS except for the root expansion by 2 links (i.e., expand *S* by all pages that are 2-link distance away from *S*), and the use of anchor text similarity weights $w(p,q)$ in hub and authority weight computations[14]. $w(p,q)$ is defined as $1 + n(t)$, where $n(t)$ is the number of query terms in the window of 50 bytes around a link (i.e., 50 bytes before <a href> and after) from *p* to *q*. Anchor window size was determined by examining the occurrence distribution of the term "Yahoo!" around http://www.yahoo.com anchor in 5000 pages, where most occurrences fell

[14] Incorporation of anchor weights into HITS can be better demonstrated by using matrix notations. If equation (2) and (3) are rewritten as $\mathbf{a}=A^T\mathbf{h}$ and $\mathbf{h}=A\mathbf{a}$, where **a** and **h** are authority and hub vectors and A is a link matrix whose $(i,j)^{th}$ entry is 1 if a link exists from p_i to p_j and 0 otherwise, then ARC formulas can be written as $\mathbf{a}=Z^T\mathbf{h}$ and $\mathbf{h}=Z\mathbf{a}$, where Z is a weighted link matrix whose $(i,j)^{th}$ entry is the anchor text similarity weight $w(i,j)$.

within the 50 byte window.

As the name indicates, the main goal of ARC was to devise a mechanism that could automatically compile and maintain resource lists similar to those provided by Yahoo!. To test the effectiveness of ARC in that regard, Chakrabarti et al. compared ARC with both Yahoo! and Infoseek in an experiment, where volunteers evaluated the retrieval results of 27 broad topic queries. The queries, constructed by choosing words or short phrases representative of pages in both Yahoo! and Infoseek directories, were submitted to AltaVista to obtain the root sets for ARC, and submitted to Yahoo! and Infoseek to find matching category pages. Volunteers were first asked to use the resulting list from each search engine for 15 to 30 minutes as a starting point to learn about the topic in any way they choose, and then asked to assign three scores to each search engine. Volunteers used a 10-point scale to rate the accuracy (i.e., how topically focused the list was), comprehensiveness (i.e., how broadly the list covered the topic), and overall value (i.e., how helpful the list was in locating valuable pages). Though the results of the experiment were not statistically significant, the average score ratios[15] showed ARC to be comparable with Infoseek and only marginally worse than Yahoo!.

Chakrabarti et al. (1998a) continued the development of the HITS-based method in the CLEVER project by making improvements to ARC. While still focusing on the main objective of topic distillation, where the objective is to derive a small number of high-quality Web pages most representative of the topic specified by a query, CLEVER incorporates more in-depth information about links between pages. Based on the assumption that pages on the same logical Web site were authored by the same organization or individual, CLEVER varies the weight of links according to the location of their endpoints, so that intra-site links confer less authority than

[15] Average score ratio is the score ratio of SE a to SE b averaged over topic.

inter-site links. Though Kleinberg (1998) had suggested eliminating intra-host links in his original paper on HITS, it is not clear whether it was implemented in ARC. A host or Website, determined by the root portion of a URL as suggested by Kleinberg, may contain within it many distinct authors (e.g., AOL), so weighting links based on location may be a better strategy than eliminating intra-host links altogether.

Another significant approach of CLEVER is based on the observation that some good hub pages contain within them sections of sub-topical link clusters. Such hub pages could cause problems for queries focusing on the subtopics by blurring topic boundaries of authority pages. For example, a good hub page on Web IR, containing links to pages on text-based techniques and link-based techniques may affect the return of Kleinberg's paper on HITS along with Salton's paper on term weighting in response to a query on link analysis. Unfortunately, the exact detail of how CLEVER addresses this problem is not provided in the paper, which mentions only that CLEVER uses "interesting (physically contiguous) sections of Web pages to determine good hubs or authorities" (Chakrabarti et al., 1998a, p.15). It might be that CLEVER partitions hub pages into physically contiguous sections based on some heuristic using HTML tags and/or anchor texts. CLEVER also reduces the scores of near-duplicate hubs in order to keep them from unduly dominating the computation.

In an experiment conducted to test the performance of the "new and improved" CLEVER, Chakrabarti et al. (1998a) compared the system's precision with that of Yahoo! and AltaVista on 27 benchmark topics used in the ARC experiment. The precision measure was based on relevance judgments made by 37 users, who evaluated 30 documents per topic (10 each from Yahoo! and AltaVista, 5 hubs and 5 authorities from CLEVER, merged and sorted alphabetically) as bad, fair, good, or fantastic, based on how useful they were in learning about

the topic. Unlike in the ARC experiment, the source of documents was hidden from the user and there was no time limit imposed. Comparison of precision results, where "good", and "fantastic" documents were considered relevant, showed that Yahoo! and CLEVER tied in 31% of the topics, while CLEVER did better in 50% of topics.

Bharat and Henzinger (1998) explored methods to augment HITS with full content analysis of documents that go beyond the use of anchor text. Their research was motivated by the findings from failure analysis of HITS, where the results of query topics with poor performance were analyzed to better understand HITS' weaknesses. In addition to the obvious finding that the neighborhood graph induced by a base set *T* must be densely connected for HITS to be effective, examinations of the problem neighborhood graphs revealed three additional instances where HITS could fail.

The first failure situation, called "mutually reinforcing relationships between hosts" (Bharat & Henzinger, 1998, p.104), was caused by certain linkage patterns that unduly influenced HITS computation. A set of documents *A* on one host (*host1*) all pointing to a single document *b* on another host (*host2*) will drive up the hub scores of *A* and the authority score of *b*. Conversely, a single document *a* on *host1* pointing to a set of documents *B* on *host2* will drive up the hub score of *a* and authority scores of *B*. When documents on the same host are authored by the same person, which is a common occurrence, such linkage patterns gives undue weight to the opinion of one person, thus violating the notion of collaborative judgment inherent in HITS. The second problem situation was caused by automatically generated links in documents created with authoring tools, which often insert commercial or proprietary links that have nothing to do with perceived authority. The third problem observed in the neighborhood graph of failed results is probably the most common and serious failing of HITS. Bharat and Henzinger often found that

hubs and authorities returned by HITS were not relevant to the query topic because the computation drifted away from the query topic toward the topic areas that were more densely linked. Kleinberg, who discussed HITS's tendency to focus on the generalized topic of a specific query, described this as "diffusion".

To overcome the HITS problems they observed, Bharat and Henzinger combined the connectivity analysis of HITS with content analysis and also strengthened the connectivity analysis by weighting links based on linkage pattern. The purpose of link weights, called "edge weights" by Bharat and Henzinger, is to combat mutually reinforcing relationships by giving fractional weights to edges (i.e., links) connecting multiple nodes (i.e., documents) on one host to a single node on another host. CLEVER's use of link weights arises from the same underlying assumption that documents on the same host implies single authorship, but the weighting is based on the location of link endpoints rather than the pattern of linkage[16]. The edge weights essentially normalize the contribution of authorship by dividing the contribution of each page by the number of pages created by the same author. To achieve this effect, HITS formulae are modified as follows:

$$a(p) = \sum_{q \to p} h(q) \times auth\_wt(q,p), \qquad (6)$$

$$h(p) = \sum_{p \to q} a(q) \times hub\_wt(p,q). \qquad (7)$$

In above equations, *auth_wt(q,p)* is 1/*m* for page *q*, whose host has *m* documents pointing to *p*,

[16] Since I could not find the details of the link weighting strategy by CLEVER, this statement about CLEVER is my inference only and may not be correct. Chakrabarti et al. (1998a) do not mention mutually reinforcing relationships, but simply say that CLEVER "varies the weight of links between pages based on the domain of their endpoints" (p.15) to prevent pages by the same author from conferring authority upon one another.

and *hub_wt(p,q)* is 1/*n* for page *q*, which is pointed by *n* documents from the host of *p*. According to Bharat and Henzinger, application of this modified algorithm successfully eliminated all the instances of mutually reinforcing relationships they had observed.

Since the topic drift problem is caused by HITS's attraction to densely linked documents that are not relevant to the query, Bharat and Henzinger reasoned that the solution lay in eliminating non-relevant documents from consideration or regulating the influence of documents based on their relevance to the query. One of the key tasks of such an approach, of course, is measuring the relevance of documents with respect to a given query, for which most link-based methods are ill equipped. Consequently, Bharat and Henzinger adapted a fusion strategy (BHITS[17]), where text-based methods of computing the relevance scores of documents were combined with link-based methods of HITS, in order to temper the contribution of links with documents' relevance to a query. It turned out that Bharat and Henzinger's method of combining connectivity and content analysis was an effective remedy for the problem caused by auto-generated links as well as the topic drift problem, since both instances involved influences of non-relevant documents that were modulated by the method.

The first step of BHITS is the determination of a document's relevance to the query topic, where the relevance is estimated by text similarity between the document and the query. In keeping with Bharat and Henzinger's contention that the query topic is often broader than the query itself in the context of topic distillation, where the aim is to find quality documents related to the topic of the query rather than to find documents that match the query precisely, BHITS defines a broad query by concatenating the first 1000 words from each document in the root set *S*

[17] Henceforth, BHITS will denote Bharat and Henzinger's fusion method of content and connectivity analysis.

(i.e., top 200 search results from a search engine). In BHITS, *tf*idf* weights[18] are used for terms in both documents and the query, and the similarity score is computed by the cosine vector similarity formula to compensate for length variations in documents (Salton & Buckley, 1988).

Once the relevance weights of all documents in the base set *T* have been computed, they can be used either to eliminate non-relevant documents entirely or to regulate the link influence based on the relevance of the document in which the link occurs. Since there is no precise point that separates relevant documents from non-relevant documents, eliminating non-relevant documents, or pruning non-relevant nodes in BHITS terminology, involves setting a relevance threshold. BHITS implements three different methods of determining the relevance threshold: one where the threshold is the median relevance weight of the base set *T*, the other where the threshold is the median relevance weight of the root set *S*, and another where the threshold is one tenth of the maximum relevance weight.

Regulating the node influence, as BHITS calls it, is an attempt to reduce the influence of less relevant nodes on the scores of their neighbors by introducing the relevance weight, *rel_wt(q)*, into the computation of hub and authority scores in the following manner:

$$a(p) = \sum_{q \to p} h(q) \times rel\_wt(q), \qquad (8)$$

$$h(p) = \sum_{p \to q} a(q) \times rel\_wt(q). \qquad (9)$$

The above formulae, modulating the node influence based on its relevance weight, can easily be

[18] Since the inverse document frequency of term is impossible to obtain for the whole Web, *idf* is based on term frequencies measured in a crawl of 400,000 Yahoo! documents in January 1997.

combined with the edge weights (equations 6 and 7)[19].

Bharat and Henzinger also explored partial content analysis (PCA) approaches in order to reduce the content analysis cost of downloading thousands of pages. Based on the assumption that not all nodes are equally influential in deciding the outcome of HITS, PCA methods attempt to prune the most highly connected, non-relevant nodes that dominate the computation. As a first step, relevance weights are computed for the 30 most highly connected documents in the root set against a reduced query vector consisting of the first 1,000 words of those 30 documents, and the pruning threshold is set at the 25th percentile. Bharat and Henzinger suggest two pruning strategies, namely degree-based pruning that selects influential nodes based on their indegree and outdegree (4*indegree + outdegree), and iterative pruning that selects nodes to prune based on HITS computations. In degree-based pruning, relevance weights of the top 100 influential nodes are computed, and nodes with relevance weights below the threshold are pruned, after which 10 iterations of edge-weighted HITS are run on the reduced graph. In iterative pruning, pruning occurs in each of 10 iterations of edge-weighted HITS by eliminating documents below the threshold from the top-5 ranking documents. The rationale for using the top-5 documents to prune is as follows: pruning top-ranking documents is sufficient to combat topic-drift because other documents supported in the ranking by high-ranking, pruned documents will be affected as well by their mutually reinforcing relationships.

To evaluate the effectiveness of the methods outlined so far, Bharat and Henzinger devised an experiment comparing 10 combinations of their methods with the baseline HITS

---

[19] $a(p) = \sum_{q \to p} h(q) \times auth\_wt(q,p) \times rel\_wt(q)$,

$h(p) = \sum_{p \to q} a(q) \times hub\_wt(p,q) \times rel\_wt(q)$.

using 28 queries from the ARC experiment (Chakrabarti et al., 1998b)[20]. The evaluation metrics used were precision and relative recall at 5 and 10 documents, where relative recall was computed by dividing the number of relevant documents retrieved by the total number of relevant documents in the pool of top-10 documents retrieved by all systems. Three volunteers, who independently evaluated a pool of the top 14 documents from all systems, made the relevance judgments.

The results of the experiment showed that the edge weights alone improved precision over HITS (26% for authorities and 23% for hubs); adding regulation and pruning each improved precision further (10% each for authorities, 10% by pruning and under 1% by regulation for hubs), but combining regulation and pruning did not give any more improvement. Recall exhibited similar patterns. The performances of partial-content analysis systems were comparable with those of full-content analysis systems. In fact, the precision of the iterative pruning system was highest overall. Bharat and Henzinger suggest that the good performance of partial-content analysis systems is due to their ability to avoid pruning non-influential but useful (i.e., connected to good hubs and authorities) documents with low relevance weights. It is also possible that good hubs with little content other than links may be pruned when matched against a fully expanded query, but manage to remain in the computation when matched against a shorter query.

Lempel and Moran (2000) take another approach to combat what they call the TKC effect (i.e., diffusion effect, topic drift) of HITS. Their SALSA algorithm (Stochastic Approach for Link-Structure Analysis) is similar in principle to PageRank in that it considers "random" walks on graphs derived from link structure, but in its implementation it follows the HITS approach of

[20] In the ARC experiment, 28 queries were constructed originally, but only 27 were used in the analysis.

identifying topic-driven neighborhood graphs while replacing the iterative algorithm of mutual reinforcement approach with a non-iterative stochastic approach to identify hubs and authorities. The authority and hub scores computed by the SALSA algorithm turn out to be equivalent to the normalized sum of weights of inlinks and outlinks, or simple count of inlinks and outlinks when link contributions are not differentiated (i.e., the link is unweighted).

Since SALSA is a non-iterative algorithm, the effect of link density on hub and authority computation is less profound than that of HITS. SALSA in essence uses the indegree as the measure of authority, which is in direct contrast to Kleinberg's contention that says the contribution of links should be differentiated by iterative link propagation. SALSA does differentiate the link contributions in the case of weighted links, but not with link propagation. Instead, Lempel and Moral suggests that links may be weighted by such factors as query-anchor text similarity or link location in a page.

Perhaps the most important reason for the claimed effectiveness of SALSA lies in its careful filtering of links in the link graph formulation stage. Lempel and Moran, proposing that filtering out of "non-informative" links is one of the most crucial steps in link analysis, eliminate 38% of links to arrive at a high-quality link graph by ignoring related-domain links (e.g., www.yahoo.com and shopping.yahoo.com), cgi scripts, and advertisement links[21] in addition to intra-domain links, which is the only type of link excluded from link analysis in HITS. Lempel and Moran suggest that link differentiation by link propagation is not as important when the link graph is of high quality. In other words, the "noise" in the link graph, which is suppressed by iterative mutual reinforcement computation of HITS, is eliminated in the link graph identification phase, thus making the end results comparable. The overpowering effect of the

[21] Advertisement links are identified by certain characters in URL, such as =, ?.

high-quality link graph is also reported in the study by Amento, Terveen and Hill (2000), who compared the effectiveness of indegree, PageRank, and authority scores on Yahoo! documents and found no significant performance differences.

Li, Shang and Zang (2002) extend HITS even further to address what they call the "small-in-large-out" problem, which is a tendency of a hub page with a large in-to-outdegree ratio to dominate the HITS computation. In their investigation of HITS-based algorithms, they observed that a hub page with a small indegree and a large outdegree in a tightly-knit link community would achieve a substantive hub score and thus affect the high authority scores of its outlinks, regardless of their relevance to the query. The authors' solution to this problem is to assign high inlink weights when a small-in-large-out page exists in the root set so that such pages would not unduly dominate the HITS computation. Their experiment, which compared the HITS, BHITS, and BHITS with small-in-large-out weighting (WHITS) showed significant performance improvement by WHITS over both HITS and BHITS.

Leveraging Implicit Links

Link analysis approaches like HITS and PageRank aim to measure the relative value of a Web page by mining the human thought involved in creating hyperlinks. These algorithms capture collective judgments reflected in hyperlink structures, and use these to rank Web pages. Hyperlinks, however, are not the only type of links that can be leveraged in Web IR. Implicit links, such as those found in a bibliographic citation index, a hierarchy of URLs, or the structure of documents can be used in various ways to help find information on the Web. Lawrence, Giles and Bollacker (1999), for example, specifically mined citation links in Web pages to build an effective search engine called CiteSeer that specializes in finding scientific research papers on

the Web.

CiteSeer uses a method called "autonomous citation indexing" to automatically compile a citation index by extracting citations and context of citations into a database. The first step for any search engine is obtaining documents of interest. Even general purpose search engines conduct selective crawls to keep their indices current, since the entire Web is simply too large to crawl in a reasonable amount of time. For a special purpose search engine that targets documents of a specific type, efficient and effective identification of such documents is crucial for its successful deployment.

Instead of using a selective crawler guided by machine learning heuristics (Chakrabarti, van der Berg, & Dom, 1999; Cho, et al., 1998), CiteSeer starts by querying mutiple search engines with terms such as postscript, PDF, technical report, conference, proceedings, publications, and papers. Then CiteSeer converts postscript and PDF documents to text using PreScript from the New Zealand Digital Library project [22] or pstotext from the DEC Virtual Paper research project [23] , and filters them by eliminating documents without reference/bibliography sections. Having thus found the seed set of scientific research papers, CiteSeer can build its database by selectively crawling the Web to locate the cited documents. CiteSeer also monitors mailing lists and newsgroups, keeps in contact with publishers, and accepts notifications by authors to supplement its collection of scientific literature, whose full text and citation indexes are continually updated using the incremental indexing method (Brown, Callan, & Croft, 1994). Once the documents have been collected, CiteSeer applies a series of heuristics to extract the citations as well as the context in which they occur.

In addition to providing full Boolean search with proximity support, CiteSeer finds

---

[22] http://www.nzdl.org/html/prescript.html
[23] http://www.research.compaq.com/SRC/virtualpaper/pstotext.html

related documents by several methods. It uses similarity scoring based on *tf*idf* weights to find documents with similar textual content by string distance comparison of article headers[24] to find similar headers, and the CCIDF measure to find articles with similar citations. CCIDF stands for Common Citation × Inverse Document Frequency, which is computed by summing up common citations weighted by their inverse frequency of citation. The IDF component of CCIDF works in similar fashion as *idf* in the *tf*idf* measure by downplaying the importance of citations that occur frequently in the collection (i.e., citations to highly cited documents).

An experimental system called ParaSite by Spertus (1997) employs "implicit" link analysis methods that examine URLs and document structure. URLs are leveraged by ParaSite in several ways. ParaSite uses the file hierarchy of URLs to determine relationships (e.g., parent, child, sibling) between pages, examines domain names of URLs to group together related pages (e.g., www.ai.mit.edu and www.mit.edu), and considers stereotypical names in URLs to infer page types (e.g., hometown.aol.com). ParaSite also examines the structure within a document in order to mine information about links. For example, it uses link proximity with respect to document hierarchy to gauge the strength of link relationships. In other words, ParaSite considers links within a same list item (i.e., <LI> tag) to be most closely related, and those within a same list (i.e., between <OL> and </OL>, or <UL> and </UL> tags) to be more closely related than ones located outside list boundaries. Consideration of document structure also comes into play in locating information about links. For example, general information about a group of links in a list can be found in the headers and text preceding a list, and specific information about an individual link can be found in the text surrounding and anchoring a link.

Spertus describes three potential applications–finding moved pages, person finder, and

[24] Headers referred to here are bibliographic in character, usually consisting of author's name, title, and publication year.

finding related pages–that use the information inferred by implicit link analysis. Pages whose URLs have changed may be found by link-based heuristics. One method is to check inlinks of moved pages, which may reveal updated URLs. Another heuristic is to remove portions of URLs until a valid URL is found and then search downwards in the URL hierarchy. Finding a person's homepage by querying a search engine with his or her name is not always successful. The page of interest may not yet be indexed by search engines, it may not contain the actual name, or the correct name may not be known. For example, finding someone named Albert who teaches Physics at Princeton University by searching within the pages of the Princeton physics department would be a lot more efficient than brute force searching of the entire Web. When the full name is known, searching the anchor text may prove quite effective (e.g., "anchor: Bill Clinton"). Related pages can be identified by finding pages pointed to by links in close proximity to one another.

Finding Related Pages

Finding related pages by link proximity as proposed by Spertus (1997) can be thought of as an extension of co-citation analysis and collaborative filtering (Shardnand & Maes, 1995), which matches users with other "like-minded" users, where "like-mindedness" is indicated by correlations among user ratings of items.

A more sophisticated approach to finding related pages was proposed by Dean and Henzinger (1999), who described two link-based algorithms that identify related Web pages, one of which is based on HITS and the other on co-citation analysis. Their objective was to quickly find a set of high quality Web pages that addresses the same topic as the page specified by a query URL. Their *Companion* algorithm extends HITS in several ways. First, it uses the same

edge weight devised by Bharat and Henzinger (equations 6 and 7) to reduce the influence of pages that all reside on one host. Second, it compresses duplicate and near-duplicate pages as well as eliminating a stoplist of URLs with very high indegree (e.g., Microsoft, Yahoo!) to keep them from dominating the hub and authority computation. Third, the base set *T* includes not only the parents (i.e., inlinks) and children (i.e., outlinks) of the query page, but also its siblings (i.e., pages that share a child with the query page). Fourth, it considers link location in a page to determine which pages to include.

To build the "vicinity graph" induced by the base set *T*, the *Companion* algorithm starts by including *m* randomly chosen parents and the first *n* children of the query page *u*, and then includes the siblings of *u* by getting *k* children of each parent of *u* immediately before and after its link to *u* as well as *l* parents of each child of *u* (excluding *u*) with highest indegree. Dean and Henzinger found that large *m* (2000) and small *k* (8) values work better than moderate values, because the large *m* prevents undue influence being exercised by a single parent page and small *k* is enough to capture links on a similar topic, which tend to be clustered together on a page. Near-duplicate pages are defined as pages with more than 10 links, 95% of which are common. When duplicate pages are detected, they are replaced with a page consisting of the union of links in duplicate pages.

The *Cocitation* algorithm, another method of finding related pages by Dean and Henzinger, finds siblings of the query page with the highest degree of co-citation (i.e., pages most frequently co-cited with the query page). Siblings, as seen above, are pages with a common parent, and the degree of co-citation is the number of common parents for a given pair of pages. The *Cocitation* algorithm builds a vicinity graph that includes only the siblings of the query page *u* by following *m* randomly chosen parents of *u* and including *k* children of each of them

immediately before and after its link to *u*. The algorithm then determines the degree of co-citation with *u* for every sibling and returns 10 pages with highest scores.

The effectiveness of the *Companion* algorithm (CM) and *Cocitation* algorithm (CC) were compared in a user study with Netscape's (NS) "What's Related[25]", which combines link, usage and content information to determine relationships between Web pages. The evaluation metrics for comparison were precision at a fixed rank, average precision, which is the sum of precisions at ranks with relevant documents divided by the number of relevant documents retrieved, and overall average precision, which is the average precision averaged over queries. Eighteen volunteers made binary[26] relevance judgments on randomly ordered pools of 30 documents retrieved by the three systems (10 by each system) for self-supplied query URLs, with an instruction that a page had to be both relevant and high quality to be scored as 1. There were a total of 59 URLs, 37 of which had results returned by all three systems.

The results showed that all three algorithms did well with highly linked query URLs, though answers by NS were about broader topics than those produced by the other systems. CM had the highest precision at rank 10 (73% better than NS for 59 queries, 40% better than NS for 37 queries), followed by CC (51% better than NS for 59 queries, 22% better than NS for 37 queries). In general, CM and CC substantially outperformed NS at all ranks, with CM performing better than CC on average. Sign Tests and Wilcoxson Sums of Ranks Tests for each pair of algorithms showed that differences between CM and NS, and between CC and NS, were statistically significant. Also, there was a large overlap in the answers of CM and CC due to the similarity in vicinity graph construction methods, but relatively little overlap between NS and

---

[25] http://home.netscape.com/escapes/related/faq.html

[26] Actually, the scoring scale was 0, 1, and '-' for inaccessible, but '-' was considered the same as 0 in precision computations.

CM or CC, which is consistent with previous research findings regarding the overlap of results among different search systems (Harman, 1994; Katzer et al., 1982; Lawrence & Giles, 1998).

Other Link-based Approaches

The link-based approaches use link structure to measure the value of documents with which to rank the retrieval results. There are other ways to leverage hyperlinks, such as visualizing the link structure (Carriere & Kazman, 1997; Mukherjea & Foley, 1995) or querying the Web as a relational database (Arocena, Mendelzon & Mihaila, 1997; Mendelzon, Mihaila, & Milo, 1996). Although these are interesting Web IR approaches in their own right, only a cursory review will be included here since they are peripheral to the main interests of this review.

Carriere and Kazman (1997) examined a link-based method for visualizing as well as ranking search engine query results. They constructed the WebQuery system, which queries a search engine and expands the returned results by inlinks and outlinks in a manner similar to HITS. WebQuery's resemblance to HITS, however, ends there. It ignores both the link directionality and the link importance, and simply ranks pages in the expanded result set by the sum of their indegree and outdegree. Although WebQuery's search method is designed to augment text-based retrieval by finding relevant documents that do not contain query terms and to filter out uninteresting documents by ordering the search results by the degree of connectivity, it falls short of more sophisticated link analysis approaches such as HITS or PageRank. WebQuery's visualization approach, on the other hand, is a significant attempt to capitalize on humans' innate pattern recognition abilities by allowing the searchers to interact with visual representations of retrieved documents and their interconnectivity, so that they may find

documents of interest more efficiently.

WebSQL by Arocena et al. (1997) demonstrates another approach to exploiting link structure between pages by using structured queries and a relational database representation of the Web. WebSQL queries multiple search engines with a uniform query interface integrating text-based retrieval with link-topology based queries. In the relational model of the Web proposed by WebSQL, Web pages are represented in a relational database containing document attributes (e.g., URL, title, text) as well as link attributes (e.g., base, href, label) so that queries specifying both content and link related features can be satisfied by combinations of attribute matching and selective link traversal. For more detailed examination of the database approach to Web IR, see Florescu, Levy, and Mendelzon (1998).

Mining the Usage Data

In addition to hyperlinks, there are two other major sources of information that provide human judgements on the value of Web documents. One source is usage data collected by Web servers and search engines that monitor surfer and searcher actions, and the other is Web directories that contain a considerable body of human knowledge and judgement in the form of topic hierarchies. Despite the fact that commercial search engines (e.g., Google, DirectHit) have readily embraced usage data mining[27], there has been scant research on exploiting Web user data until recently.

Schapira (1999) describes an experimental system called Pluribus, which reranks SE results by "user popularity" scores based on accumlated user selection frequencies of search

[27] See the Search Engine Review section of Search Engine Watch for articles on Web search engine technology (http://www.searchenginewatch.com/resources/article.php/2156581). A summary of search engines based on Search Engine Watch data in 2000 can be seen in http://ella.slis.indiana.edu./~kiyang/wse/search/WSE.pdf (Appendix B).

results. Pluribus was designed to test the hypothesis that retrieval performance could be improved by the implicit "relevant feedback" in users' collective past actions on search results. The basic idea of Pluribus is encapsulated in its ranking formula, which increments the scores of documents frequently selected by searchers and decreases the scores of documents infrequently selected by searchers. The user popularity score U(p), as seen in the equation below, consists of the selection count S(p), which is the number of times searchers selected a page p, and the expected selection count E(p), which is the rank-based number of times p should have been selected (0.6 for the top 25%, 0.3 for the middle 50%, 0.1 for the bottom 25% of the SE results), and the base relevance score R(p) returned by a SE.

$$U(p) = R(p) + 100*S(p) - 100*E(p) \quad (10)$$

With this equation, low ranking documents will rise rapidly to the top as they are selected by users, but their rate of ascent will slow as they near the top, and conversely, high ranking documents not selected by users will fall rapidly towards the bottom. One of the potential problems with this formula is its tendency to overpower the original text-base ranking with exaggerated contributions from selection counts, as the number of selections increases.

The effectiveness of Pluribus hinges on the validity of its assumptions, which are as follows. First, user selection/non-selection of a document indicates relevance/irrelevance. Second, users consistently select relevant documents and ignore irrelevant documents. Third, the same query represents the same information need. Similar assumptions underpin collaborative filtering (Shardnand & Maes, 1995), a successful example of which is a recommender system (Resnik & Varian, 1997) that suggests music, films, books, and other products and services to

users based on examples of their likes and dislikes. In the context of collaborative filtering, however, users' evaluation of items are explicitly assigned in the form of scores, ranks, or opinions, rather than implied in the form of link clicks.

Estimating relevance by counting link clicks can be problematic. Users can select a document only to decide it is not relevant, or skip relevant documents they have seen before. Consideration of page browsing time and link traversal count (i.e., the number of links traversed for a page) may be needed for a more accurate measure of user-based relevance. Another weakness of Pluribus is its dependence on query overlap (i.e., queries must be repeated), which is necessary for learning to occur. Pluribus also does nothing to improve the recall of the original SE results since all it does is improve precision by reranking documents based on user popularity.

Cui et al. (2002) also explored the idea of link click counts as implicit relevance judgments, but they leveraged the usage data for probabilistic query expansion instead of direct manipulation of document scores. After computing correlations between query terms and document terms based on 4.8 million query sessions and 42,000 documents from the Encarta website (http://encarta.msn.com), they expanded the query with highly correlated terms to improve the retrieval performance. They compared their approach with Local Context Analysis query expansion (Xu & Croft, 1996), which leverages both corpus-wide term co-occurrence statistics and pseudo-relevance feedback, and found that their method improved the retrieval performance substantially. Cui et al.'s query expansion by usage data mining improves on the Pluribus approach on several levels; the massive quantity of usage data compensates for potential false positives (i.e., user clicks that are not indication of relevance), and the implicit relevance judgments embedded in usage data are further distilled into the form of term correlation

probabilities, which enable high-quality query expansion that can boost the overall retrieval performance rather than simple reranking of the initial retrieval results.

In a related study, Zhou et al. (2001) analyzed web logs to compute the "Associate Degree" (AD), which is essentially a conditional probability of users visiting page *q* from page *p*. Although their application of AD was limited to increasing the link connectivity of Web sites by adding implicit links described by high AD, it is not difficult to envision a scenario where AD is used to derive a user-based popularity measure of a Web page. One simple approach, given a large quantity of usage data, would be to apply a link analysis formula, such as PageRank, to the weighted directed graph of user visiting patterns. Such a measure could be either query-dependent (if AD is computed from search engine logs) or query-independent (if AD is computed from web server logs), and computed solely from usage data or in combination with hyperlink data. Once such approach is proposed by Miller et al. (2001), who describe a modification of the HITS algorithm based on the idea that more frequently followed links should play a larger role in determining authority scores by employing a usage-weighted link matrix[28].

Although Davison's (2002) strategy for pre-fetching Web pages is designed to speed up page access by improved Web caching, it deserves mentioning here because he approaches the idea of estimating the degree of association between pages based on user visiting patterns with a new twist. Davison's approach differs from that of Zhou et al. in that he uses the content similarity of page *q* to a set of pages that lead to *q* as an association measure between page *p* and *q*. Such set consists of page *p*, which explicitly links *q*, and pages the user visited prior to *p* during the same user session.

Last but certainly not least, Xue et al.'s (2003) study of implicit links analysis for small

---

[28] Miller et al.'s approach is identical to the ARC formula except that the $(i,j)^{th}$ entry of the weighted link matrix is the frequency of link traversal from page *i* to *j* in the usage data instead of the anchor text similarity weight w(*i,j*).

Web searches raises some interesting notions. Observing the frequent failure of link analysis methods in TREC experiments (Hawking, 2001; Hawking & Craswell, 2002), Xue et al. contend that link analysis applied to small Web collections is ineffective due to the truncated link structure, and suggest that link analysis should be applied to an implicit link structure constructed from user access patterns instead of the incomplete hyperlink structure of a small Web collection. Their approach to small Web search extends previous work on Web usage data mining by incorporating both the probabilistic approach to implicit link construction (Zhou et al., 2001) and usage-based modification of link analysis (Miller et al., 2001). Xue et al.'s implicit link construction method is based on conditional probabilities associated with pairs of pages visited in user sessions and produces implicit links with associated weights, which are then used to modify the link matrix of PageRank formula.

To test their approach to small Web search, Xue et al. mined 336,000 implicit links from UC Berkeley Web server log collected over a 4-month period and compared their modified PageRank approach to fulltext, PageRank, modified HITS, and DirectHit searches. Since the effectiveness of the modified PageRank method they proposed is directly affected by the quality of implicit link construction, Xue et al, evaluated random subsets of 375 implicit and 290 explicit links by 7 human subjects to find that 67% of implicit and 39% of explicit links were recommendation links. This finding suggests that implicit link construction can not only help complete the link structure but also raise the quality of links by reducing the link noise (e.g., navigational links). The search results, evaluated using precision at rank 30 and degree of authority (proportion of authority pages at rank 10), showed the proposed method outperforming all other methods by significant margins.

The basic idea of mining usage data for user-based relevance seems to hold promise for

Web IR, especially when combined with other sources of evidence. The 670 million searches per day logged by major search engines suggest that usage data is a potential source of evidence rich with possibilities.

Web IR experiments in TREC

TREC has been a fertile ground for IR experimentation using large-scale test collections (Harman, 1994; Voorhees, 2003). In addition to demonstrating the continuing viability of statistical IR approaches, TREC has also shown that fine tuning of IR systems to the data and the task at hand can improve retrieval performance. Over the years, TREC has been expanding its scope beyond the realm of text retrieval to areas such as cross-language retrieval, interactive retrieval, video retrieval and Web retrieval, and is actively investigating various non-traditional IR issues such as user-system interaction and link analysis, as well as testing the applicability of traditional methods to non-traditional settings.

The Web IR experiment of TREC, called the Web track, initially investigated the same ad-hoc retrieval task undertaken with plain text documents. Although many TREC participants explored methods of leveraging non-textual sources of information, such as hyperlinks and document structure, the general consensus among the early Web track participants was that link analysis and other non-textual methods did not perform as well as content-based retrieval methods (Hawking et al., 1999; Hawking et al., 2000; Gurrin & Smeaton, 2001; Savoy & Rasolofo, 2001).

There has been much speculation as to why link analysis, which showed much promise in previous research and has been so readily embraced by commercial search engines, did not prove more useful in Web track experiments. Most speculation pointed to potential problems with the

Web track's test collection, from the inadequate link structure of truncated Web data (Savoy & Picard, 1998; Singhal & Kazkiel, 2001), and relevance judgments that penalize the link analysis by not counting the hub pages as relevant (Voorhees & Harman, 2000) and boost the content analysis by counting multiple relevant pages from the same site as relevant (Singhal & Kazkiel, 2001), to queries that are too detailed and specific to be representative of real-world Web searches (Singhal & Kaszkiel, 2001).

In a case study of TREC algorithms, Singhal and Kazkiel (2001) suggest that the Web track task is not representative of real-world Web searches, which range from locating a specific site to finding high quality sites on a given topic. In an experiment that compared retrieval performances of TREC's content-based, ad-hoc algorithms to the search results of several Web search engines in a more "realistic" environment–designed specifically to address the problems with the Web track experiments[29]–Singhal and Kazkiel found that TREC content-based systems performed much worse than Web search engines utilizing link analysis and suggested that content-based methods may retrieve documents about a given topic, but not necessarily the documents users are looking for.

In a related study, Craswell, Hawking and Robertson (2001) used link anchor text to find the main entry point of a specific Web site (i.e., homepage). In a carefully controlled experiment, where all the retrieval system parameters were identical except for the document content that consisted of either textual content or inlink anchor texts of a Web page, they found that using anchor text was twice as effective as using document content in what they called the site finding task.

---

[29] Singhal and Kazkiel used 100 queries that sought a certain page/site in 17.8 million Web pages (217.5 gigabyte crawl in October, 2000) and counted the number of queries that retrieved the correct page in top ranks to compare the retrieval performances.

In an effort to address the criticisms and problems associated with the early Web track experiments, TREC abandoned the ad-hoc Web retrieval task in 2002 in favor of topic distillation and named-page finding tasks and replaced its earlier Web test collection of randomly selected Web pages with a larger and potentially higher quality, domain-specific collection[30]. The topic distillation task in TREC-2002 entailed finding a short, comprehensive list of pages that are good information resources, and the named-page finding tasks involved finding a specific page whose name is described by the query (Hawking & Craswell, 2002; Craswell & Hawking, 2003).

Adjustment of the Web track environment brought forth renewed interest in retrieval approaches that leverage Web-specific sources of evidences, such as link structure and document structure. For the homepage finding task, where the objective is to find the entry page of a specific site described by the query, the Web page's URL characteristics, such as type and length, as well as the anchor text of the Web page's inlinks, proved to be useful sources of information (Hawking & Craswell, 2002). In the named-page finding task, which is similar to the homepage finding task except that the target page described by the query is not necessarily the entry point of a Web site but any specific page on the Web, the use of anchor text still proved to be an effective strategy though the use of URL characteristics did not work well as it did in the homepage finding task (Craswell & Hawking, 2003).

In the topic distillation task, anchor text still seemed to be a useful resource, especially as a way to boost the performance of content-based methods via fusion (i.e., result merging), although its utility level fell much below that achieved in named-page finding tasks (Hawking &

[30] The current test collection of the Web track consists of 1.25 million Web pages (19 gigabytes) from .gov domain, which is larger, less diverse and likely to be of higher quality than the previous collection, which was a 10 gigabyte subset of the Web crawl from Internet Archive.

Craswell, 2002; Craswell & Hawking, 2003). Site compression strategies, which attempt to select the "best" pages of a given site, were another common theme in the topic distillation task, once again demonstrating the importance of fine-tuning the retrieval system according to the task at hand (Amitay et al., 2003b; Zhang et al., 2003b). It is interesting to note that link analysis (e.g., PageRank, HITS variations) has not yet proven itself to be an effective strategy and the content-based method still seems to be the most dominant factor in the Web track. In fact, the two best results in the TREC-2002 topic distillation task were achieved by the baseline systems that used only content-based methods (MacFarlane, 2003; Zhang et al., 2003b).

The two tasks in the TREC-2003 Web track were slight modifications of tasks in TREC-2002. The topic distillation task in TREC-2003 is described as finding relevant "homepages" given a broad query, which introduces bias in favor of homepage finding approaches. For the second Web track task, the named-page finding task was combined with homepage finding task in a blind mix of 150 homepage queries and 150 named-page queries (Craswell et al., 2003b). There were only a few relevant documents for topic distillation queries in 2003 (average 10.32 relevant documents per topic), so R-precision, which is the precision at rank *n*, where *n* is the number of relevant documents for a given topic, was used to evaluate the topic distillation results instead of precision at rank 10, as was done in the previous year. The evaluation metrics for home/named-page finding results were mean reciprocal rank, which is reciprocal ranks of the first correct answer averaged over queries, and the success rate, which is the proportion of queries with correct answers at top 10 ranks.

The homepage bias of topic distillation task in TREC-2003 prompted many participants to combine their topic distillation approaches with strategies found to be effective in homepage finding task (Kraaij, Westerveld, & Hiemstra, 2002). Tomlinson (2003), for example, used URL

information[31] to identify potential homepages and boost their retrieval scores, Jijkoun et al. (2003) combined the results of body text, title, and inlink anchor text retrieval runs, and Zhang et al. (2003a) employed an entry page location algorithm based on URL characteristics to boost the score of entry pages as well as utilizing site compression techniques and the anchor text index.

System tuning based on query classification was one of the new approaches that emerged in the home/named-page finding tasks, where participants recognized the disadvantage of the "one size fits all" approach for the mixed search types. Using past Web track results as training data, Craswell et al. (2003a) tuned anchor text weight and Okapi BM25 parameters to produce system optimizations for each of homepage and named-page finding task and used a simple query classifier based on keyword occurrence to select an appropriate system to deploy for each given query. Amitay et al.'s (2003a) "Query-Sensitive Tuner" is another example of retrieval strategy adjustment by query typing. Query-Sensitive Tuner, which considers query length as well as the expected number of documents containing all query terms to classify queries, tunes the weighting of content, anchor text, and indegree contributions to the combined retrieval score based on query classification. Plachouras et al. (2003) determined which combinations of evidence to use in a fusion formula based on the concept of "query scope", which is computed from statistical evidence obtained from the set of retrieved documents.

It is not clear which query classification approach is most effective, nor is it obvious whether a high-powered query classifier alone is sufficient to deal with a mixed search environment such as home/named-page task. What is apparent, however, is the need to explore a diverse set of retrieval settings that reflects the true conditions of Web search environment. By rigorously engaging in such endeavors, TREC has played an indispensable role in Web IR

[31] Tomlinson classified Web pages into categories of root, subroot, path and file, based on URL endings (e.g. index.htm, home.htm) and the number of slashes in the URL.

research. In addition to producing high quality test collections for Web IR and much associated experimental data from numerous researchers both in and outside the TREC community, TREC has led the way in establishing novel performance measures designed specifically for Web IR tasks. Furthermore, TREC has been instrumental in validating traditional IR approaches in the Web setting, as well as nurturing and fine-tuning new ideas. For instance, TREC research validated term weighting and query expansion methods of traditional IR for the Web, established anchor text, URL and document structure as important sources of Web evidence, and suggested such techniques as site compression, query classification, and fusion as promising areas of investigation for future Web IR research.

Research in Information Organization

Knowledge organization, the process of organizing concepts into an ordered group of categories (i.e., taxonomy), is the way humans understand the world. When we encounter a phenomenon for the first time, we try to understand it by comparing it with what we already know and identifying it with a similar pattern in our existing frame of reference, thus "transforming isolated and incoherent sense impressions into recognizable objects and recurring patterns" (Langridge, 1992, p.3). We learn about the world by looking for differences and similarities in things and finding relationships between them. The process of grouping together similar things according to some common quality or characteristics is an integral part of daily life. As infants, we begin by partitioning the world into those elements that give us comfort and those that cause discomfort. As we grow and accumulate more information, we refine our understanding of the world by updating our knowledge organization system with more complex categories and relationships between them.

According to John Dewey, "knowledge is classification." Langridge (1992) states that without classification there could be no human thought, action and organization. Classification, in a broad sense, is a mechanism for both organizing and utilizing knowledge. At a fundamental level, we make sense of the world by organizing perceptions and experiences. In other words, information organization (IO) is essential to human learning and knowledge. Not surprisingly, researchers in the field of artificial intelligence, machine learning in particular, have been studying IO as a way to emulate human intelligence. IO has also been explored in information and library science as a mechanism with which to bridge the user's information need and the information collection. Librarians for centuries have been organizing library collections in various ways (e.g., Library of Congress Classification, Dewey Decimal Classification) to help library patrons find information. Web directories, though more ad-hoc and less standardized than traditional library systems, are another application of IO that guides users through the information discovery process.

IO approaches in IR reflect ideas from both machine learning and library science. Based on the notion that IO facilitates learning and is one of the basic building blocks of knowledge, some researchers have experimented with IO as a tool to develop intelligent IR systems that go beyond term matching, while others have investigated how it could be used to help users better grasp the content of a collection. Though system- and user-oriented IO approaches differ in their focus, both hope to harness intelligence and knowledge to enhance retrieval by first organizing information in some manner.

Two types of information organization are studied in IR: classification[32] and clustering. Classification organizes entities by pigeonholing them into predefined categories, whereas

[32] In this paper, "classification" is used in the narrow sense of the term, mostly in the context of text categorization.

clustering organizes information by grouping similar or related entities together. In classification, categories are first determined and objects are assigned to them according to characteristics of interest. In clustering, categories are revealed as a result of grouping objects based on some common characteristics. Although both clustering and classification have been active areas of research for some time, studies that investigate their application on the Web have been scarce. Yet, information organization is a concept reflected across the Web landscape, from Web directories and metadata initiatives to the Semantic Web and the digital library movement. Research that explores adaptation of clustering and classification approaches for the Web may not only prove useful for ongoing Web IO activities, but also provide valuable insights into leveraging knowledge for information discovery on the Web.

Organizing Web Documents

Traditional text-based IR research uses homogeneous corpora with coherent vocabulary, high quality content, and congruous authorship. The Web corpus, however, introduces the challenges of diverse authorship, vocabulary, and quality. Furthermore, some Web documents are intentionally fragmented to facilitate navigation and hyperlinking, making it difficult to determine their topics from local content alone. Leveraging hyperlinks in the Web is also more problematic than in traditional hypertext IR research due to the number of link types that are hard to classify automatically.

Information organization on the Web inherits all the problems and challenges generally associated with Web IR. For instance, it is difficult to cluster or classify the whole Web due to its massive size and diversity. Even if such a feat were possible, most clustering approaches, which are not incremental, and text categorization approaches, which are based on a static classification

scheme, would not be able to deal with the dynamic nature of the Web. Consequently, methods of organizing Web documents need to be efficient, flexible and dynamic. Moreover, post-retrieval organization of retrieved documents may be both a more desirable and realistic approach than trying to organize the entire Web.

Clustering the Web

There are several approaches to clustering Web documents, such as co-citation analysis to identify topic clusters (Larson, 1996), the "trawling" process to find emerging Web communities (Kumar et al., 1999), and topic management approaches that collect and organize Web pages related to a particular topic (Modha & Spangler, 2000; Mukherjea, S., 2000a, 2000b). The post-retrieval clustering approach of Scatter/Gather (Cutting et al., 1992; Hearst & Pederson, 1996) has also been applied to the Web to dynamically produce topic-coherent clusters of retrieved documents (Sahami, Yusufali, & Baldonaldo, 1998)

One of the few studies that tailored a clustering algorithm specifically for the Web was conducted by Zamir and Etzioni (1998). To satisfy the rigorous requirements of the Web environment, which demands fast, effective, and dynamic algorithms that produce concise and accurate descriptions, they proposed an incremental, linear-time clustering algorithm called *Suffix Tree Clustering* (STC), which clusters documents based on shared phrases. In a related study, Zamir and Etzioni (1999) used STC to dynamically group metasearch results into clusters by using snippets returned from search engines rather than the full-text of documents in order to avoid fetching result pages, usually the major bottleneck in Web IR[33].

Site clustering, sometimes called site compression, is the grouping of Web pages by

[33] A prior study comparing the clusters using snippets to those using full-texts has shown the snippet approach to have only a moderate degradation of 15% average precision loss (Zamire & Etzioni, 1998).

authorship rather than by content similarity. Site compression in its simplest form, which uses the domain name portion of URLs to cluster pages by common authorship, was first used in link analysis to differentiate self-citational links from true recommendation links (Kleinberg, 1998; Bharat & Henzinger, 1998). Largely motivated by topic distillation and entry page finding tasks in TREC, researchers explored more sophisticated site compression methods, which combine multiple sources of Web evidence, such as page content, hyperlinks, and URL, to not only cluster pages by site but also identify the site entry page (Amitay et al., 20003a, 2003b; Wen et al., 2003; Zhang et al., 2003a, 2003b).

Clustering is also used to aggregate multiple Web pages that make up a coherent body of material on a single topic. While link analysis approaches that produce topical clusters (Chakrabarti et al., 1998a, 1998b; Kleinberg, 1988; Kumar et al., 1999) leverage link structure to achieve document clustering by topic, Eiron and McCurley (2003) proposes an evidence fusion approach that combines analysis of anchor text similarity and hierarchical structure of hyperlinks to aggregate compound Web documents into a single information unit.

Classifying the Web

Though post-retrieval clustering of Web documents offers a viable and effective alternative to traditional ranked list retrieval, it lacks the clarity and purpose of a structured organizational hierarchy. Classification of Web documents not only produces a useful organization of information that can be browsed or searched, but also offers a standardized way–like a thesaurus–to describe or refer to the content of Web pages, which can be leveraged to enhance the retrieval performance. Chakrabarti, Dom and Indyk (1998) explored the application of a term- and link-based classifier to hypertext documents and proposed a method that uses link

analysis for better classification.

Geffner et al. (1999) suggest a method of exploiting the classification hierarchy. By representing a classification hierarchy as a data cube, where leaves become singleton ranges and nodes become the union of all ranges of its children, the data cube approach summarizes and encapsulates organizational structure in a multidimensional database of attributes. Yahoo!'s display of its categories with associated sizes and subcategories, for example, can be thought of as an application of the data cube idea to present a compact summary of the categories.

The study by Jenkins et al. (1998) describes a method for using the Dewey Decimal Classification (DDC) to classify Web pages for WWLib[34]. WWLib is organized by Automated Classification Engine (ACE), which use class representatives constructed from DDC to hierarchically classify documents. The class representatives, consisting of DDC classmark and accompanying header text, as well as a manually selected set of keywords and synonyms, are compared to documents using Dices similarity coefficient. ACE is similar to TAPER (Chakrabarti et al., 1997) in that it employs the hierarchical classification approach using customized class representatives at each node of the DDC hierarchy. The ACE approach has the advantage of using a universal classification scheme that covers all subject areas at high levels of granularity, but requires that of class representatives be constructed manually.

Chekuri et al. (1996) describe a method of leveraging Yahoo! categories with the objective of increasing the precision of the Web search. Their main idea is to use a Yahoo!-trained classifier in conjunction with the traditional keyword search in an integrated search interface. In their proposed system, where users can specify both keywords and category terms, the classifier categorizes the keyword search results and presents filtered and organized

[34] WWLib (http://www.scit.wlv.ac.uk/wwlib/), a virtual library at the University of Wolverhampton, is a searchable, classified catalogue of Web pages in the United Kingdom.

documents to the user. In keeping with Larson's (1992) idea of automatic classification as a semi-automatic classification tool, Chekuri et al. suggest presenting the top *k* matching categories to the user and contend that a reasonably accurate classifier is sufficient in such a setting. To validate this contention, they trained a Rocchio classifier (Rocchio, 1971) using a sample of 2,000 Web pages from 20 high-level Yahoo! categories and classified 500 random documents from Yahoo!. More than 50% of test documents were classified correctly at the top ranking category, more than 80% at top 3, and more than 90% at top 5, from which they concluded that displaying the top 10 categories for each of the search results would be quite sufficient for an interactive search interface.

A slightly different method of leveraging Yahoo! categories is described by Grobelnik and Mladenic (1998). Instead of using the actual Web documents associated with Yahoo! categories, they use Yahoo!'s descriptions of the categorized pages to train a hierarchical Naive Bayes classifier (Koller & Sahami, 1997). Since the feature space induced by such descriptions can be sparse, Grobelnik and Mladenic propagate the description terms upwards in the hierarchy with weights proportional to the node size where they appear. In classifying documents, they use a pruning strategy where categories with less than the required number of features in common with the target document are pruned using an inverted index of features by categories. They tested their method by training the classifier on 3 sets of training documents selected from the top 14 Yahoo! categories and classifying 100 to 300 randomly selected Web pages categorized by Yahoo!. The results showed that the correct category assignment probability was over 0.99 using only a small number of features[35]. They also found that pruning over half the categories resulted in misclassification of only 10 to 15% of the documents.

---

[35] Using n-grams instead of single words as features, they found the category feature vector size of 3 to 4 to be effective in most cases.

Labrou and Finin (1999) also leveraged Yahoo! terms to train a classifier called Teltale. Their approach, which combined category labels, summaries and link titles in a Yahoo! page and content of the Web pages referenced by the links to describe categories, revealed that best results would occur when using very brief descriptions of category entries (i.e., Yahoo! summaries and titles).

The research reviewed so far leverages existing taxonomies to organize Web documents. A recent study by Glover et al. (2002) examined the use of anchor text for automatic classification and labeling of Web pages. They found that the fusion approach, which combines text-based classification with link-based classification by considering features in inlink anchor text as well as content text to classify a Web page, worked better than either approach by itself.

Based on the assumption that a site's functionality creates site signature, Amitay et al. (2003c) examined the structural and connectivity-based properties of 8 Web site[36] categories by function (e.g., Web hierarchies/directories, virtual hosting service, corporate site) to identify patterns for site-classifier feature determination. They then selected 16 features based on internal and external linkage patterns to and from the site (e.g., outdegree, outdegree per leaf page) as well as page distribution pattern in the site hierarchy (e.g., average page level, percentage of pages in most populated level). A decision-rule classifier with one or more of these features was then constructed using training data. The classifier operates with a series of decision rules, one of which may be of the kind "if feature #5 (outdegree) is greater than 5,000 AND feature #8 (outdegree per leaf page) is greater than 10, then assign 0.7 to category class #3 (Web Directory)". Amitay et al.'s approach to site classification addresses the principal weakness of link-based classifiers, the fact that they can be unduly influenced by the absolute link density.

[36] A site refers to the set of Web pages belonging to the same virtual entity.

Effective classification of site by function could be useful for filtering or clustering search results as well as modifying link-weight propagation in link analysis.

The study by Liu, Chin, and Ng (2003) takes topic distillation a step further by infusing data mining and hierarchical classification into their "topic mining" algorithm. They argue that current term matching and link analysis approaches are inadequate for satisfying the information need associated with a new topic learning, which requires the retrieval of the topic hierarchy populated by pages that include definitions and descriptions of topic and subtopics. Their proposed topic mining algorithm, which is designed to find and organize pages containing definitions and descriptions of topic and subtopics, constructs and populates a topic hierarchy by recursively applying the following process until the topic hierarchy is complete (i.e., there are no more subtopics to be identified). First, obtain a set of relevant pages about a given topic by retrieving 100 top-ranked pages from a search engine with a topical query. Next, apply the "concept discovery" method, which is based on frequency analysis of HTML tag-emphasized phrases (e.g., <b>, <h>) to mine subtopic concepts from those pages. Then apply the "definition finding" method, which is based on analysis of HTML structure (e.g., header tag frequency, anchor text similarity) and pattern matching of definition cues (e.g., "known as", "defined as"), to identify pages containing definitions of the topic and subtopics. If another level of topic hierarchy is desired, repeat the process by querying a search engine with subtopic concepts. In an experiment where the results of Liu et al.'s topic mining search were compared with the top 10 results of Google and AskJeeves, the topic mining method showed higher precision at rank 10 than other systems in 26 out of 28 topics tested.

The application of classification methods to Web IR is not limited to content, nor is its purpose limited to information organization. Query or task classification, as commonly practiced

in Web IR research (Amitay et al., 2003a; Craswell et al., 2003a; Plachouras et al., 2003), aims to categorize queries by task type and is typically a precursor to dynamic system tuning that will optimize system parameters for the identified query type. Recently, a couple of new approaches that apply classification for the purpose of retrieval have been proposed. One approach exploits existing categories of Web pages to directly influence PageRank computations (Jeh & Widom, 2003; Haveliwala, 2002), while the other automatically identifies effective query modifications from iterations of the classification and feature selection process (Flake et al., 2002).

Haveliwala's research extends PageRank capability with a set of "topic sensitive" PageRank vectors, each of which is used to compute a Web page's importance score with respect to a given topic. A PageRank vector can be thought of as a vector of preference weights for all the pages, where a weight reflects the degree of preference for a given page. If we express PageRank vector as a function, $PV(p)$ that returns the preference weight of page p, we can rewrite the PageRank formula in the following way:

$$R(p) = c \cdot PV(p) + (1-c) \cdot \sum_{i=1}^{k} \frac{R(p_i)}{C(p_i)} \qquad (11)$$

where $C(p)$ is the outdegree of p, and p_i denotes the inlinks of p and c is a "teleportation" probability that a Web surfer breaks the normal traversal of hyperlinks and "teleports" to another page. In the case of global PageRank where page preference is not considered, $PV(p)$ returns a uniform weight of $1/T$ (T is the number of pages in the Web). In the case of "topic sensitive" PageRank, a PageRank vector biased towards a topic is constructed by assigning preference weights to pages about that topic. Such a PageRank vector introduces a computational bias towards its topic since the "teleportation" to a preferred page (e.g., topics pages with preference

weights, pages linking to those pages, and so on) will produce a larger PageRank score than jumping to a non-topical page.

Haveliwala constructed a set of 16 PageRank vectors from pages classified in 16 Open Directory (http://dmoz.org/) categories by assigning a uniform preference weight of 1/*N* to pages in the Open Directory category and zero to all other pages, where *N* is the total number of pages in a given category. He then computed 16 sets of topic-sensitive PageRank scores, which are combined at retrieval time to generate a final "query-sensitive" PageRank score. Topic-sensitive PageRank scores are combined by weighted sum formula, where the weight of each topic-sensitive PageRank score is the class probability of the query belonging to a given class, computed by a probabilistic classifier.

Jeh and Widom (2003) generalize the notion of topic-sensitive PageRank vectors as Personalized PageRank Vectors (PPV), which are PageRank vectors that can reflect any number of personal preferences. Haveliwala's approach to computing preference-biased PageRank scores is resource-intensive and can accommodate only a limited number of PageRank vectors. To address this problem, Jeh and Widom propose a solution that represents PPV as a linear combination of "basis vectors", which are encoded as shared components that can derive PPV at query time in order to allow scalable PageRank computation. PPV is an innovative approach that "fuses" automatic classification and link analysis at a method level to dynamically optimize the retrieval strategy according to user preferences.

Flake et al.'s (2002) research can be viewed as tackling a similar problem in a quite different fashion. Flake et al. address the problem of "personalized" retrieval by modifying a user-specified query with automatically generated query modifications that will effectively filter retrieval results by category. Query modifications (QM) are extracted from nonlinear SVMs

using training data of positive and negative category examples in an iterative process. Both QM and PPV aim to accommodate a facet of information need not explicitly stated in a query, but approach the problem space from different directions. Although both approaches make use of prior evidence (e.g., personal preference data) and employ classification as an integral component of the overall strategy, QM solution focuses on query optimization while PPV solution focuses on dynamic tuning of the scoring function.

The idea of tackling the implicit query facet warrants further consideration. The facet in question, which might be a topic category, document type, or simply a set of preference features not explicitly stated in a query, could be mined from an existing taxonomy, link topology, or usage data among other things. Harnessing multiple sources of evidence and combining different approaches to satisfy multi-faceted information needs may be the type of approach required to bring Web IR research to the next level.

Concluding Remarks

In this chapter, we examined the complexities of the Web search environment and reviewed strategies for finding information on the Web from two different perspectives. In both the retrieval and organizational approaches to information discovery on the Web, leveraging rich sources of evidence on the Web has been a consistent theme. Utilizing link information is a central approach in Web IR. The assumptions underlying the link-based methods (Craswell, Hawking, & Robertson, 2001)–the recommendation assumption (page author is recommending the page that he/she is linking to), topic locality assumption (pages connected by links are more likely to be about the same topic), and anchor description assumption (anchor text of a link describe its target)–not only make intuitive sense but also have been shown to be valid in

numerous studies (Kleinbeg, 1997; Page et al., 1997; Dean & Henzinger, 1999; Davison, 2000; Craswell & Hawking, 2003).

It is therefore rather curious to note that sophisticated link analysis techniques such as PageRank and HITS have yet to prove their effectiveness in the TREC arena, where other Web IR techniques such as anchor text use and URL-based scoring have been validated repeatedly. Whether this phenomenon is due to the small Web effect (Xue et al., 2003) or an artifact of some other characteristics of TREC environment, such as query, relevance judgments, or retrieval task, remains to be seen.

The nature of the Web search environment is such that retrieval approaches based on single sources of evidence can suffer from weaknesses that can hurt the retrieval performance in certain situations. For example, content-based IR approaches have difficulty dealing with the variability in vocabulary and quality of web documents, while link-based approaches can suffer from incomplete or noisy link topology. The inadequacies of singular Web IR approaches coupled with the fusion hypothesis (i.e., "fusion is good for IR") make a strong argument for combining multiple sources of evidence as a potentially advantageous retrieval strategy for Web IR.

The fusion strategies in early Web IR experiments achieved only moderate success, mostly by combining content-based results with anchor-text results (Hawking & Craswell, 2002). Fusion, however, has been widely adopted by today's Web IR researchers. Combining multiple sources of Web evidence such as document content, structure, hyperlinks, and URL information has become de facto standard practice in TREC at the time of writing (Craswell et al., 2003b). Several innovative approaches that integrate both retrieval and information organization approaches have also been explored in recent studies (Chin & Ng, 2003; Flake et

al., 2002; Haveliwala, 2002; Jeh & Widom, 2003).

Finding information on the Web is a complex and challenging task that requires innovative solutions. Research in Web IR has produced some approaches that effectively leverage the characteristics of the Web search environment and suggested the potential of fusion that combine both methods and sources of evidences. As for the future of Web IR, we may see a move towards information rich areas such as user data mining and knowledge-based retrieval that leverage stored human knowledge (e.g., Web directories), where fusion and dynamic tuning are standard approaches to bring together multiple sources of evidence and multiple methods for personalized information discovery on the Web.

References

Allan, J. (1996). Automatic hypertext link typing. *Proceedings of the 7th ACM Conference on Hypertext*, 42-52.

Amitay, E., Carmel, D., Darlow, A., Herscovici, M., Lempel, R., Soffter, A., Kraft, R., & Zien, J. (2003a). Juru at TREC 2003: Topic distillation using query-sensitive tuning and cohesiveness filtering. *The 12th Text Retrieval Conference (TREC 2003) Notebook*, 255-261.

Amitay, E., Carmel, D., Darlow, A., Lempel, R., & Soffer, A. (2003b). Topic distillation with knowledge agents. *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, 263-272.

Amitay, E., Carmel, D., Darlow, A., Lempel, R., & Soffer, A. (2003c). The connectivity sonar: detecting site functionality by structural patterns. *Proceedings of the 14th ACM Conference on Hypertext and Hypermedia*, 38-47.

Amento, B, Terveen, L., & Hill, W. (2000). Does authority mean quality? *Proceedings of the 23rd ACM SIGIR Conference on Research and Development in Information Retrieval*, 296-303.

Arocena, G.O., Mendelzon, A.O., & Mihaila, G.A. (1997). Applications of a Web query language. *Proceedings of the 6th International WWW Conference*, 587–595.

Bar-Ilan, J. (2004). The use of Web search engines in information science research. *Annual Review of Information Science and Technology*, 38, 231-288.

Barabasi, A. (2003). *Linked: How everything is connected to everything else and what it means*. New York: Plume.

Baron, L., Tague-Sutcliffe, J., & Kinnucan, M. T. (1996). Labeled, typed links as cues when reading hypertext documents. *Journal of the American Society for Information Science*, 47(12), 896-908.

Bernstein, M. (1998). Patterns of hypertext. *Proceedings of the 9th ACM Conference on Hypertext*, 21-29.

Bharat, K., & Henzinger, M. R. (1998). Improved algorithms for topic distillation in hyperlinked environments. *Proceedings of the 21st ACM SIGIR Conference on Research and Development in Information Retrieval*, 104-111.

Borgman, C. & Furner, J. (2002). Scholarly Communication and Bibliometrics. *Annual Review of Information Science and Technology*, 36, 3-72.

Botafogo, R. A., Rivlin, E., & Shneiderman, B. (1992). Structural analysis of hypertexts: Identifying hierarchies and useful metrics. *ACM Transactions on Information Systems*, 10(2), 142-180.

Bray, T. (1996). Measuring the Web. *Proceedings of the 5th International WWW Conference*, 994-1005.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Proceedings of the 7th International WWW Conference*, 107-117.

Broder, A. (2002). A taxonomy of web search. *ACM SIGIR Forum*, 36(2), 3-10.

Broder, A. Z., Kumar, S. R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A., & Wiener, J. (2000). Graph structure in the Web: experiments and models. *Proceedings of the 9th WWW Conference*, 309-320

Brown, E. W., Callan, J. P., & Croft, W. B. (1994). Fast incremental indexing of for full-text information retrieval. *Proceedings of the 20th VLDB Conference*, 192-202.

Carriere, J., & Kazman, R. (1997). WebQuery: searching and visualizing the Web through connectivity. *Proceedings of the 6th WWW Conference*, 701–711.

Chakrabarti, S., Dom, B., Agrawal, R., & Raghavan, P. (1997) Using taxonomy, discriminants, and signatures for navigating in text databases. *Proceedings of the 23rd VLDB Conference*, 446-455.

Chakrabarti, S., Dom, B., Gibson, D., Kumar, S.R., Raghavan, P., Rajagopalan, S., & Tomkins., A. (1998a). Experiments in topic distillation. *ACM SIGIR workshop on Hypertext Information Retrieval on the Web*, 13-21.

Chakrabarti, S., Dom, B., Raghavan, P., Rajagopalan, S., Gibson, D., & Kleinberg, J. (1998b). Automatic resource list compilation by analyzing hyperlink structure and associated text. *Proceedings of the 7th International WWW Conference*, 65-74.

Chakrabarti, S., Dom, B., & Indyk, P. (1998). Enhanced hypertext categorization using hyperlinks. *Proceedings of ACM SIGMOD Conference on Management of Data*, 307-318.

Chakrabarti, S., van der Berg, M., & Dom, B. (1999). Focused crawling: a new approach to topic-specific Web resource discovery. *Proceedings of the 8th WWW Conference*, 1623-1640.

Chekuri, C., Goldwasser, M., Raghavan, P., & Upfal, E. (1996). Web search using automatic classification. *Proceedings of the 6th WWW Conference*.

Cho, J., Garcia-Molina, H., & Page, L. (1998). Efficient crawling through URL ordering. *Proceedings of the 7th WWW Conference*, 161-172.

Craswell,N., & Hawking, D. (2003). Overview of the TREC-2002 Web track. *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, 86-95.

Craswell, N., Hawking, D., & Robertson, S. (2001). Effective site finding using link anchor information. *Proceedings of the 24th ACM SIGIR Conference on Research and Development in Information Retrieval*, 250-257.

Craswell, N., Hawking, D., McLean, A., Wilkinson, R., & Wu, M. (2003a). TREC 12 Web Track at CSIRO. *The 12th Text Retrieval Conference (TREC 2003) Notebook*, 237-247.

Craswell, N., Hawking, D., Wilkinson, R., & Wu, M. (2003b). Overview of the TREC 2003 Web Track. *The 12th Text Retrieval Conference (TREC 2003) Notebook*, 220-236.

Croft, W. B. (1993). Retrieval strategies for hypertext. *Information Processing and Management*, 29, 313-324.

Cronin, B., Snyder, H., & Atkins, H. (1997). Comparative citation ranking of authors in monographic and journal literature: a study of sociology. *Journal of Documentation*, 53(3), 263-273.

Cui, H., Wen, J., Nie, J., & Ma, W. (2002). Probabilistic query expansion using query logs. *Proceedings of the 11th International WWW Conference*, 325-332.

Cutting, D.R., Karger, D.R., Pedersen, J.O., Tukey, J.W. (1992). Scatter/Gather: A cluster-based approach to browsing large document collections. *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 318-329.

Davison, B. (2000). Topical locality in the Web. *Proceedings of the 23rd ACM SIGIR Conference on Research and Development in Information Retrieval*, 272-279.

Davison, B. (2002). Predicting web actions from HTML content. *Proceedings of the 13th ACM Conference on Hypertext and Hypermedia*, 159-168.

Dean, J., & Henzinger, M. R., (1999). Finding related pages in the World Wide Web. *Proceedings of the 8th International WWW Conference*, 389-401.

Eiron, N. & McCurley, K. (2003). Untangling compound documents on the web. *Proceedings of the 14th ACM Conference on Hypertext and Hypermedia*, 85-94.

Flake, G.,W., Glover, E. J., Lawrence, S., Giles, C. L. (2002). Extracting query modifications from nonlinear SVMs. *Proceedings of the 11th International WWW Conference*, 317-324.

Florescu, D., Levy, A., & Mendelzon, A. (1998). Database techniques for the World-Wide Web: A Survey. *SIGMOD Record*, 27(3), 59-74.

Frei, H.P., & Stieger, D. (1995). The Use of semantic links in hypertext information retrieval. *Information Processing and Management*, 31(1), 1-13.

Geffner, S., Agrawal, D., Abbadi, A. E., & Smith, T. (1999). Browsing large digital library collections using classification hierarchies. *Proceedings of the 8th ACM International Conference on Information and Knowledge Management*, 195-201.

Glover, E. J., Tsioutsiouliklis, K., Lawrence, S., Pennock, D. M., & Flake, G. W. (2002). Using web structure for classifying and describing web pages. *Proceedings of the 11th International WWW Conference*, 562-569

Grobelnik, M., Mladenic, D. (1998) Efficient text categorization. *Proceedings of Text Mining Workshop on ECML-98*, 1-10.

Gurrin, C., & Smeaton, A.F. (2001). Dublin City University experiments in connectivity analysis for TREC-9. *Proceedings of the 9th Text Retrieval Conference (TREC-9)*, 179-188.

Harman, D. (1994). Overview of the second Text Retrieval Conference. *Proceedings of the 2nd Text Retrieval Conference (TREC-2)*, 1-20.

Haveliwala, T. (2002). Topic-sensitive PageRank. *Proceedings of the 11th WWW Conference*, 517-526.

Hawking, D. (2001). Overview of the TREC-9 Web track. *Proceedings of the 9th Text Retrieval Conference (TREC-9)*, 87-102.

Hawking, D., & Craswell, N. (2002). Overview of the TREC-2001 Web track. *Proceedings of the 10th Text Retrieval Conference (TREC 2001)*, 25-31

Hawking, D., & Craswell, N., Thistlewaite, P., & Harman, D. (1999). Results and challenges in web search evaluation. *Proceedings of the 8th WWW Conference*, 243-252.

Hawking, D, Voorhees, E, Craswell, N., & Bailey, P. (2000). Overview of the TREC-8 Web track. *Proceedings of the 8th Text Retrieval Conference (TREC-8)*, 131-148.

Hearst, M., & Pedersen, J.O. (1996). Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. *Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval*, 76-84.

Hölscher, C., & Strube, G. (2000). Web search behavior of internet experts and newbies. *Proceedings of the 9th International WWW Conference*.

Jansen, B. J., Spink, A., Bateman, J., & Saracevic, T. (1998). Real life information retrieval: a study of user queries on the Web. *SIGIR Forum*, 32(1), 5-17.

Jansen, B. J., Spink, A., & Saracevic, T. (1998). Failure analysis in query construction: data and analysis from a large sample of Web queries. *Proceedings of the 3rd ACM International Conference on Digital Libraries*, 289-290.

Jansen, B. J., Spink, A., & Saracevic, T. (2000). Real life, real users and real needs: A study and analysis of users queries on the Web. *Information Processing and Management*, 36(2): 207-227.

Jeh, G. & Widom, J. (2003). Scaling personalized web search. *Proceedings of the 12th International WWW Conference*, 271 – 279.

Jenkins, C., Jackson, M., Burden, P., & Wallis, J. (1998). Automatic classification of Web resources using Java and Dewey Decimal Classification. *Proceedings of the 7th International WWW Conference*.

Jijkoun, V., Kamps, J., Mishne, G., Monz, C., de Rijke, M., Schlobach, S., & Tsur, O.(2003). The University of Amsterdam at TREC 2003. *The 12th Text REtrieval Conference (TREC 2003) Notebook*, 560-573.

Kahle, B. (1997). Archiving the Internet. *Scientific American*, March, 1997

Katzer, J., McGill, M. J., Tessier, J. A., Frakes, W., & DasGupta, P. (1982). A study of the overlap among document representations. *Information Technology: Research and Development*, 1, 261-274.

Kehoe, C., Pitkow, J., Sutton, K., Aggarwal, G., & Rogers, J. D. (1999). Results of GVU's tenth WWW user survey. Retrieved November 11, 2003, from http://www.gvu.gatech.edu/user_surveys/survey-1998-10/tenthreport.html

Kessler, M. M. (1963). Bibliographic coupling between scientific papers. *American Documentation*, 14, 10-25.

Kim, H. J. (2000). Motivation for hyperlinking in scholarly electronic articles: a qualitative study. *Journal of the American Society for Information Science*, 51(10), 887-899.

Kleinberg, J. (1998). Authoritative sources in a hyperlinked environment. *Proceeding of the 9th ACM-SIAM Symposium on Discrete Algorithms*, 668-677.

Koller, D., & Sahami, M. (1997). Hierarchically classifying documents using very few words. *Proceedings of the 14th International Conference on Machine Learning*, 170-178.

Kopak, R. W. (1999). Functional link typing in hypertext. *ACM Computing Surveys (CSUR)*, 31(4es).

Kraaij, W., Westerveld, T., & Hiemstra, D. (2002). The importance of prior probabilities for entry page search. *Proceedings of the 25th ACM SIGIR Conference on Research and Development in Information Retrieval*, 27-34.

Kumar, S. R., Raghavan, P., Rajagopalan, S., & Tomkins, A. (1999). Trawling the Web for emerging cyber-communities. *Proceedings of the 8th WWW Conference*, 403-415.

Labrou, Y., & Finin, T. (1999). Yahoo! as an ontology: using Yahoo! categories to describe documents. *Proceedings of the 8th ACM International Conference on Information and Knowledge Management*, 180-187.

Langridge, D.W. (1992). *Classification: Its kinds, elements, systems, and applications*. London: Bowker Saur.

Larson, R. (1992). Experiment in automatic Library of Congress Classification. *Journal of the American Society for Information Science*, 43(2), 130-148.

Larson R.R. (1996). Bibliometrics of the World Wide Web: An exploratory analysis of the intellectual structure of cyberspace. *Proceedings of Annual American Society for Information Science Meeting*, 71-78.

Lawrence, S., & Giles, C. L. (1998). Searching the World Wide Web. *Science*, 280, 98-100.

Lawrence, S., & Giles, C. L. (1999a). Searching the Web: general and scientific information access. *IEEE Communications*, 37(1), 116-122.

Lawrence, S., & Giles, C. L. (1999b). Accessibility of information on the Web. *Nature*, 400 (6740), 107-110.

Lawrence, S., Giles, C. L., & Bollacker, K (1999). Digital libraries and autonomous citation indexing. *IEEE Computer*, 32(6), 67-71.

Lempel, R., & Moran, S. (2001). SALSA: The stochastic approach for link-structure analysis. *ACM Transactions on Information Systems*, 19(2), 131-160.

Liu, B., Chin, C., Ng, H. (2003). Mining topic-specific concepts and definitions on the web. *Proceedings of the 12th International WWW Conference*, 251-260.

MacFarlane, A. (2003). Pliers at TREC 2002. *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, 152-155.

Marchionini, G. (1992). Interfaces for end-user information seeking. *Journal of the American Society for Information Science*, 43(2), 156-163.

Marchiori, M. (1997). The quest for correct information on the Web: Hyper search engines. *Proceedings of the 6th International WWW Conference*, 265-274.

McBryan, O. A. (1994). GENVL and WWWW: Tools for taming the Web. *Proceedings of the 1st International WWW Conference*, 58-67.

Mendelzon, A., Mihaila, G., & Milo, T. (1996). Querying the World Wide Web. *Proceedings of the 1st International Conference on Parallel and Distributed Information Systems (PDIS'96)*, 80-91.

Modha, D., & Spangler, W. S. (2000). Clustering hypertext with applications to Web searching. *Proceedings of the 11th ACM Conference on Hypertext*, 143-152.

Miller, J. C., Rae, G., Schaefer, F., Ward, L. A., LoFaro, T., & Farahat, A. (2001). Modifications of Kleinberg's HITS algorithm using matrix exponentiation and web log records. *Proceedings of the 24th ACM SIGIR Conference on Research and Development in Information Retrieval*, 444-445.

Mukherjea, S. (2000a). Organizing topic-specific Web information. *Proceedings of the 11th ACM Conference on Hypertext*, 133-141.

Mukherjea, S. (2000b). WTMS: a system for collecting and analyzing topic-specific Web information. *Proceedings of the 9th International WWW Conference*, 457-471.

Mukherjea, S., & Foley, J. (1995). Visualizing the World-Wide Web with the navigational view builder. *Proceedings of the 3rd WWW Conference*, 1075-1087.

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). The PageRank citation ranking: Bringing order to the Web. Retrieved November 30, 2003, from http://dbpubs.stanford.edu/pub/showDoc.Fulltext?lang=en&doc=1999-66&format=pdf

Pirolli P., Pitkow J., Rao R. (1996). Silk from a sow's ear: Extracting usable structures from the Web. *Proceedings of ACM SIGCHI Conference on Human Factors in Computing Systems*, 118-125.

Plachouras, V., Ounis, I., van Rijsberger, C. J., & Cacheda, F. (2003). University of Glasgow at the Web Track: Dynamic application of hyperlink analysis using the query scope. *The 12th Text REtrieval Conference (TREC 2003) Notebook*, 248-254.

Pollock, A., & Hockley, A. (1997). What's wrong with Internet searching? *D-Lib Magazine*. Retrieved November 30, 2003, from http://www.dlib.org/dlib/march97/bt/03pollock.html

Rasmussen, E. (2003). Indexing and retrieval from the Web. *Annual Review of Information Science and Technology*, 37, 91-124.

Resnik, P., & Varian, H. R. (1997). Introduction (to the special section on recommender systems). *Communications of the ACM*, 40(3), 56-59

Rivlin, E., Botafogo, R., & Shneiderman, B. (1994). Navigating in hyperspace: Designing a structure-based toolbox. *Communications of the ACM*, 37(2), 87-96.

Rocchio, J. J., Jr. (1971). Relevance feedback in information retrieval. In G. Salton (Ed.), *The SMART Retrieval System - Experments in Automatic Document Processing*, 313-323. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Sahami, M., Yusufali, S., & Baldonaldo, M. (1998). SONIA: a service for organizing networked information autonomously. *Proceedings of the 3rd ACM International Conference on Digital Libraries*, 200-209.

Salton, G., & Buckley, C. (1988). Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24, 513-523.

Salton, G., & Buckley, C. (1991). Global text matching for information retrieval. *Science*, 253, 1012-1015.

Salton, G., Buckley, C., Allan, J. (1994). Automatic structuring and retrieval of large text files. *Communications of the ACM*, 37(11), 97-108.

Savoy, J., & Picard, J. (1998). Report on the TREC-8 Experiment: Searching on the Web and in distributed collections. *Proceedings of the 8th Text Retrieval Conference (TREC-8)*, 229-240.

Savoy, J., & Rasolofo, Y. (2001). Report on the TREC-9 experiment: Link-based retrieval and distributed collections. *Proceedings of the 9th Text Retrieval Conference (TREC-9)*, 579-516.

Schapira, A. (1999). Collaboratively searching the Web – An initial study. Retrieved November 30, 2003, from http://none.cs.umass.edu/~schapira/thesis/report/

Li, L., Shang, Y., & Zhang, W. (2002). Improvement of HITS-based algorithms on Web documents. *Proceedings of the 11th International WWW Conference*, 527-535.

Shardanand, U., & Maes, P. (1995). Social information filtering: Algorithms for automating "word of mouth". *Proceedings of the ACM Conference on Human Factors in Computing Systems: Mosaic of Creativity*, 210-217.

Shaw, W. M., Jr. (1991a). Subject and citation indexing. Part I: The clustering structure of composite representations in the cystic fibrosis document collection. *Journal of the American Society for Information Science*, 42, 669-675.

Shaw, W. M., Jr. (1991b). Subject and citation indexing. Part II: The optimal, cluster-based retrieval performance of composite representations. *Journal of the American Society for Information Science*, 42, 676-684.

Shum, S. B. (1996). The missing link: Hypermedia usability research & the Web. *ACM SIGCHI Bulletin*, 28 (4), 68-75.

Silverstein, C., Henzinger, M., Marais, H., & Moricz, M. (1998). Analysis of a very large AltaVista query log. *Technical Report 1998-014*, COMPAQ System Research Center.

Singhal, A., & Kaszkiel, M. (2001). A case study in Web search using TREC algorithms. *Proceedings of the 11th International WWW Conference*, 708-716.

Small, H. (1973). Co-citation in the scientific literature: A new measure of the relationship between two documents. *Journal of the American Society for Information Science*, 24(4), 265-269.

Sparck Jones, K. (1971). *Automatic keyword classification for information retrieval*. London: Butterworth.

Spertus, E. (1997). ParaSite: Mining structural information on the Web. *Proceedings of the 6th International WWW Conference*, 587-595.

Spink, A., Wolfram, D., Jansen, B. J., & Saracevic, T. (2001). Searching the web: The public and their queries. *Journal of the American Society for Information Science*, 53(2), 226-234.

Tomlinson, S. (2003). Robust, Web and Genomic retrieval with Hummingbird SearchServer at TREC 2003. *The 12th Text REtrieval Conference (TREC 2003) Notebook*, 372-385.

Trigg, R., & Weiser, M. (1983). TEXTNET: A network-based approach to text handling. *ACM Transactions on Office Information Systems*, 4(1).

Voorhees, E. (2003). Overview of TREC 2002. *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, 1-16.

Voorhees, E., & Harman, D. (2000). Overview of the eighth Text Retrieval Conference. *Proceedings of the 8th Text Retrieval Conference (TREC-8)*, 1-24.

Wen, J. R., Song, R., Cai, D., Zhu, K. Yu, S., Ye, S., & Ma, W.Y. (2003). Microsoft Research Asia at the Web Track of TREC 2003. *The 12th Text Retrieval Conference (TREC 2003) Notebook*, 262-272.

Weiss, R., Velez, B., Sheldon, M. A., Nemprempre, C., Szilagyi, P., Duda, A., & Gifford, D. K. (1996). Hypursuit: A hierarchical network search engine that exploits content-link hypertext clustering. *Proceedings of the 7th ACM Conference on Hypertext*, 180-193.

White, H.D., & McCain, K.W. (1989). Bibliometrics. *Annual Review of Information Science and Technology*, 24, 119-165.

Willett, P. (1988). Recent trends in hierarchic document clustering: A critical review. *Information Processing and Management*, 24, 577-597.

Wolfram, D., Spink, A., Jansen, B. J., & Saracevic, T. (2001). Vox populi: The public searching of the web. *Journal of the American Society for Information Science*, 53(12), 1073-1074.

Woodruff, A., Aoki, P. M., Brewer, E., Gauthier, P., and Rowe, L. A. (1996). An investigation of documents from the World Wide Web. *Proceedings of the 5th International WWW Conference*.

Xu, J., & Croft, W. B. (1996). Query expansion using local and global analysis. *Proceedings of the 20th ACM SIGIR Conference on Research and Development in Information Retrieval*, 4-11.

Xu, H., Yang, Z., Wang, B., Liu, B., Cheng, J., Liu, Y., Yang, Z., Cheng, X., & Bai, S. (2003). TREC 11 Experiments at CAS-ICT: Filtering and Web. *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, 141-151.

Xue, G., Zeng, H., Chen, Z., Ma, W., Zhang, H, & Lu, C. (2003). Implicit link analysis for small web search. *Proceedings of the 26th ACM SIGIR Conference on Research and Development in Information Retrieval*, 56-63.

Zamir, O., & Etzioni, O. (1998). Web document clustering: a feasibility demonstration. *Proceedings of the 22nd ACM SIGIR Conference on Research and Development in Information Retrieval*, 46-54.

Zamir, O., & Etzioni, O. (1999). Grouper: a dynamic clustering interface to Web search results. *Proceedings of the 8th International WWW Conference*.

Zhang, M., Lin, C., Liu, Y., Zhao, L., Ma, L., & Ma, S. (2003a). THUIR at TREC 2003: Novelty, Robust, Web and HARD. *The 12th Text Retrieval Conference (TREC 2003) Notebook*, 137-148.

Zhang, M., Song, R., Lin, C., Ma, S., Jiang, Z., Jin, Y., Liu, Y., & Zhao, L. (2003b). THU TREC 2002: Web track experiments. *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, 591-594.

Zhou, B., Chen, J., Shi, J., Zhang, H., & Wu, Q. (2001). Website link structure evaluation and improvement based on user visiting patterns. *Proceedings of the 12th ACM Conference on Hypertext and Hypermedia*, 241-244.