WELCOME TO THE GATHERING PLACE

Computer Networks and ISDN Systems 30 (1998) 65–74

COMPUTER NETWORKS and ISDN SYSTEMS

Automatic resource compilation by analyzing hyperlink structure and associated text

Soumen Chakrabarti [a,*], Byron Dom [a], Prabhakar Raghavan [a], Sridhar Rajagopalan [a], David Gibson [b], Jon Kleinberg [c]

[a] IBM Almaden Research Center K53, 650 Harry Road, San Jose, CA 95120, USA
[b] Computer Science Division, Soda Hall, University of California, Berkeley, CA 94720, USA
[c] Department of Computer Science, Upson Hall, Cornell University, Ithaca, NY 14853, USA

Abstract

We describe the design, prototyping and evaluation of ARC, a system for automatically compiling a list of authoritative Web resources on any (sufficiently broad) topic. The goal of ARC is to compile resource lists similar to those provided by Yahoo! or Infoseek. The fundamental difference is that these services construct lists either manually or through a combination of human and automated effort, while ARC operates fully automatically. We describe the evaluation of ARC, Yahoo!, and Infoseek resource lists by a panel of human users. This evaluation suggests that the resources found by ARC frequently fare almost as well as, and sometimes better than, lists of resources that are manually compiled or classified into a topic. We also provide examples of ARC resource lists for the reader to examine. © 1998 Published by Elsevier Science B.V. All rights reserved.

Keywords: Search: Taxonomies; Link analysis; Anchor text; Information retrieval

1. Overview

The subject of this paper is the design and evaluation of an *automatic resource compiler.* An automatic resource compiler is a system which, given a topic that is broad and well-represented on the Web, will seek out and return a list of Web resources that it considers the most authoritative for that topic. Our system is built on an algorithm that performs a local analysis of both text and links to arrive at a "global consensus" of the best resources for the topic. We describe a user-study, comparing our resource compiler with commercial, human-compiled/assisted services. To our knowledge, this is one of the first systematic user-studies comparing the quality of multiple Web resource lists compiled using different methods. Our study suggests that, although our resource lists are compiled wholly automatically (and despite being presented to users without any embellishments in the "look and feel" or the presentation context), they fare relatively well compared to the commercial human-compiled lists.

When Web users seek definitive information on a broad topic, they frequently go to a hierarchical, manually-compiled taxonomy such as **Yahoo!** [2], or

[*] Corresponding author.

[2] http://www.yahoo.com

0169-7552/98/$19.00 © 1998 Published by Elsevier Science B.V. All rights reserved.
PII S0169-7552(98)00087-7

a human-assisted compilation such as **Infoseek** [3]. The role of such a taxonomy is to provide, for any broad topic, such a resource list with high-quality resources on the topic. In this paper we describe ARC (for Automatic Resource Compiler), a part of the CLEVER project on information retrieval at the IBM Almaden Research Center. The goal of ARC is to automatically compile a resource list on any topic that is broad and well-represented on the Web. By using an automated system to compile resource lists, we obtain a faster coverage of the available resources and of the topic space than a human can achieve (or, alternatively, are able to update and maintain more resource lists more frequently). As our studies with human users show, the loss in quality is not significant compared to manually or semi-manually compiled lists.

1.1. Related prior work

The use of links for ranking documents is similar to work on citation analysis in the field of bibliometrics (see e.g. [12]). In the context of the Web, links have been used for enhancing relevance judgments by Rivlin et al. [9] and Weiss et al. [11]. They have been incorporated into query-based frameworks for searching by Arocena et al. [1] and by Spertus [10].

Our work is oriented in a different direction — namely, to use links as a means of harnessing the latent human annotation in hyper-links so as to broaden a user search and focus on a type of "high-quality" page. Similar motivation arises in work of Pirolli et al. [7]; Carriere and Kazman [2]; and Brin and Page [5]. Pirolli et al. discuss a method based on link and text-based information for grouping and categorizing WWW pages. Carriere and Kazman use the number of neighbours (without regard to the directions of links) of a page in the link structure as a method of ranking pages; and Brin and Page view Web searches as random walks to assign a topic-independent "rank" to each page on the WWW, which can then be used to re-order the output of a search engine. (For a more detailed review of search engines and their rank functions, including some based on the number of links pointing to a Web page, see Search Engine Watch [8].) Finally, the link-based algorithm of Kleinberg [4] serves as one of the building blocks of our method here; this connection is described in more detail in Section 2 below, explaining how we enhance it with textual analysis.

1.2. Road map

We begin in Section 2 below with a description of our technique and how some of its parameters are fixed. In Section 3 we describe our experiments using a number of topics, with a diverse set of users from many different backgrounds. In Section 4 we summarize the ratings given by these evaluators, as well as their qualitative comments and suggestions.

2. Algorithm

We now describe our algorithm, and the experiments that we use to set values for the small number of parameters in the algorithm. The algorithm has three phases: a **search-and-growth** phase, a **weighting** phase, and an **iteration-and-reporting** phase.

Given a topic, the algorithm first gathers a collection of pages from among which it will distill ones that it considers the best for the topic. This is the intent of the first phase, which is nearly identical to that in Kleinberg's HITS technique [4]. The topic is sent to a term-based search engine — *AltaVista* in our case — and a *root set* of 200 documents containing the topic term(s) is collected. The particular root set returned by the search engine (among all the Web resources containing the topic as a text string) is determined by its own scoring function. The root set is then augmented through the following *expansion* step: we add to the root set (1) any document that points to a document in the root set, and (2) any document that is pointed to by a document in the root set. We perform this expansion step twice (in Kleinberg's work, this was performed only once), thus including all pages which are link-distance two or less from at least one page in the root set. We will call the set of documents obtained in this way the *augmented set*. In our experience, the augmented set contained between a few hundred and 3000 distinct pages, depending on the topic.

[3] http://www.infoseek.com

We now develop two fundamental ideas. The first idea, due to Kleinberg [4] is that there are two types of useful pages. An **authority** page is one that contains a lot of information about the topic. A **hub** page is one that contains a large number of links to pages containing information about the topic — an example of a hub page is a resource list on some specific topic. The basic principle here is the following mutually reinforcing relationship between hubs and authorities. A good hub page points to many good authority pages. A good authority page is pointed to by many good hub pages. To convert this principle into a method for finding good hubs and authorities, we first describe a local iterative process [4] that "bootstraps" the mutually reinforcing relationship described above to locate good hubs and authorities. We then present the second fundamental notion underlying our algorithm, which sharpens its accuracy when focusing on a topic. Finally, we present our overall algorithm and a description of the experiments that help us fix its parameters.

Kleinberg maintains, for each page p in the augmented set, a *hub score*, $h(p)$ and an *authority score*, $a(p)$. Each iteration consists of two steps: (1) replace each $a(p)$ by the sum of the $h(p)$ values of pages *pointing to* p; (2) replace each $h(p)$ by the sum of the $a(p)$ values of pages *pointed to by* p. Note that this iterative process ignores the text describing the topics; we remedy this by altering these sums to be **weighted** in a fashion described below, so as to maintain focus on the topic. The idea is to iterate this new, text-weighted process for a number of steps, then pick the pages with the top hub and authority scores.

To this end we introduce our second fundamental notion: the text around *href* links to a page p is descriptive of the contents of p; note that these hrefs are not in p, but in pages pointing to p. In particular, if text descriptive of a topic occurs in the text around an href into p from a good hub, it reinforces our belief that p is an authority on the topic. How do we incorporate this textual conferral of authority into the basic iterative process described above? The idea is to assign to each link (from page p to page q of the augmented set) a positive numerical *weight* $w(p,q)$ that increases with the amount of topic-related text in the vicinity of the href from p to q. This assignment is the second, *weighting phase* mentioned above. The precise mechanism we use for computing these weights is described in Section 2.1 below; for now, let us continue to the iteration and reporting phase, assuming that this topic-dependent link weighting has been done.

In the final phase, we compute two vectors $\mathbf{h}$ (for hub) and $\mathbf{a}$ (for authority), with one entry for each page in the augmented set. The entries of the first contain scores for the value of each page as a hub, and the second describes the value of each page as an authority.

We construct a matrix W that contains an entry corresponding to each ordered pair p, q of pages in the augmented set. This entry is $w(p,q)$ (computed as below) when page p points to q, and 0 otherwise. Let Z be the matrix transpose of W. We set the vector $\mathbf{h}$ equal to 1 initially and iteratively execute the following two steps k times:

$$\mathbf{a} = W\mathbf{h},$$

$$\mathbf{h} = Z\mathbf{a}.$$

After k iterations we output the pages with the 15 highest values in h as the hubs, and the 15 highest values in a as the authorities, without further annotation or human filtering. Thus our process is completely automated. Our choice of the quantity 15 here is somewhat arbitrary: it arose from our sense that a good resource list should offer the user a set of pointers that is easy to grasp from a single browser frame.

Intuitively, the first step in each iteration reflects the notion that good authority pages are pointed to by hub pages and are described in them as being relevant to the topic text. The second step in each iteration reflects the notion that good hub pages point to good authority pages and describe them as being relevant to the topic text. What do we set k to? It follows from the theory of eigenvectors [3] that, as k increases, the relative values of the components of $\mathbf{h}$ and $\mathbf{a}$ converge to a unique steady state, given that the entries of W are non-negative real numbers. In our case, a very small value of k is sufficient — and hence the computation can be performed extremely efficiently — for two reasons. First, we have empirically observed that convergence is quite rapid for the matrices that we are dealing with. Second, we need something considerably weaker than convergence: we only require that the identities of the top 15 hubs/authorities become stable, since this is all that goes into the final resource list. This is an important respect in which our goals differ from those of classical matrix eigenvector computations: whereas that literature focuses on the time required for the values of the eigenvector components to reach a stable state, our emphasis is only on identifying the 15 largest entries without regard to the actual values of these entries. We found on a wide range of tests that this type of s type of "near-convergence" occurs around five iterations. We therefore decided to fix k to be 5.

2.1. Computing the weights $w(p, q)$

Recall that the weight $w(p, q)$ is a measure of the authority on the topic invested by p in q. If the text in the vicinity of the href from p to q contains text descriptive of the topic at hand, we want to increase $w(p, q)$; this idea of anchor-text arose first in the work of McBryan [6]. The immediate questions, then, are (1) what precisely is "vicinity"? and (2) how do we map the occurrences of descriptive text into a real-valued weight? Our idea is to look on either side of the href for a window of B bytes, where B is a parameter determined through an experiment described below; we call this the *anchor window*. Note that this includes the text between the *<a href="...">* and *</a>* tags. Let $n(t)$ denote the number of matches between terms in the topic description in this anchor window. For this purpose, a term may be specified as a contiguous string of words. We set

$$w(p, q) = 1 + n(t).$$

Since many entries of **W** are larger than one, the entries of **h** and **a** may grow as we iterate; however, since we only need their relative values, we normalize after each iteration to keep the entries small.

Finally, we describe the determination of B, the parameter governing the width of the anchor window. Postulating that the string <a href="http://www.yahoo.com"> would typically co-occur with the text *Yahoo* in close proximity, we studied on a test set of over 5000 Web pages drawn from the Web — the distance to the nearest occurrence of *Yahoo* around all hrefs to http://www.yahoo.com in these pages. The results are shown below in Table 1; the first row indicates distance from the string <a href="http://www.yahoo.com">, while the second row indicates the number of occurrences of the string *Yahoo* at that distance. Here a distance of zero corresponds to occurrences between <a href="http://www.yahoo.com"> and </a>. A negative distance connotes occurrences before the href, and positive distances after.

Table 1
Anchor text position versus distance

Distance	−100	−75	−50	−25	0	25	50	75	100
Density	1	6	11	31	880	73	112	21	7

The table suggests that most occurrences are within 50 bytes of the href. Qualitatively similar experiments with hrefs other than Yahoo (where the text associated with a URL is likely to be as clear-cut) suggested similar values of B. We therefore set B to be 50.

2.2. Implementation

Our experimental system consists of a computation kernel written in C, and a control layer and GUI written in Tcl/Tk. An 80GB disk-based Web-cache hosted on a PC enables us to store augmented sets for various topics locally, allowing us to repeat text and link analysis for various parameter settings. The emphasis in our current implementation has not been heavy-duty performance (in that we do not envision our system fielding thousands of queries per second and producing answers in real time); instead, we focused on the quality of our resource lists. The iterative computation at the core of the analysis takes about a second for a single resource list, on a variety of modern platforms. We expect that, in a full fledged taxonomy generation, the principal bottleneck will be the time cost of crawling the Web and extracting all the root and augmented sets.

3. Experiments

In this section we describe the setup by which a panel of human users evaluated our resource lists in comparison with Yahoo! and Infoseek. The parameters for this experiment are: (1) the choice of topics; (2) well-known sources to compare with the output of ARC; (3) the metrics for evaluation; (4) and the choice of volunteers to test the output of our system.

3.1. Topics and baselines for comparison

Our test topics had to be chosen so that with some reasonable browsing effort our volunteers could make judgments about the quality of our output, even if they are not experts on the topic. One way the volunteers could do this relatively easily is through comparison with similar resource pages in well-known Web directories. Several Web directories, such as Yahoo!, Infoseek, etc. are regarded as "super-hubs"; therefore it is natural to pick such directories for comparison. This in part dictated the possible topics we could experiment on.

We started by picking a set of topics, each described by a word or a short phrase (2–3 words). Most topics were picked so that there were representative "resource" pages in both Yahoo! and Infoseek. We tried to touch topics in arts, sciences, health, entertainment, and social issues. We picked 28 topics for our study: *affirmative action, alcoholism, amusement parks, architecture, bicycling, blues, classical guitar, cheese, cruises, computer vision, field hockey, gardening, graphic design, Gulf war, HIV, lyme disease, mutual funds, parallel architecture, rock climbing, recycling cans, stamp collecting, Shakespeare, sushi, telecommuting, Thailand tourism, table tennis, vintage cars* and *zen buddhism*. We therefore believe that our system was tested on fairly representative topics for which typical Web users are likely to seek authoritative resource lists.

3.2. Volunteers and test setup

The participants range in age from early 20s to 50s, and were spread around North America and Asia. All have some experience with Web browsing and search. The set included computer science students and professionals, and also a significant number of professionals from other areas (including service organizations, psychology, and linguistics). Each participant was assigned one or two topics to rate, and could optionally rate any others they wished to. This was done to ensure that all our topics got adequately covered.

The participants were directed to a URL where they were presented with a tabular evaluation form (see the Appendix). Each row of the table represented one of the topics. There were three columns to be compared, corresponding to Yahoo!, Infoseek, and the resource page compiled by our system. In many cases, the topic was directly present as a predefined resource page in the Yahoo! or Infoseek directory, and our table entry pointed to this page. In a few cases, no single resource page appeared satisfactory; in these cases, we listed the response of Yahoo! or Infoseek to the topic posed as a *query*. The resource page from our system consisted of two side-by-side lists: one containing a list of the 15 top-rated hubs, the other showed the top 15 authorities.

We decided not to try to present the three sources of resource-lists as a *blind test*: i.e., a test in which evaluators were given 3 lists with their sources masked. There were several basic reasons for this. First, we felt that a blind test, with a uniform "look" to each resource list, would put Yahoo! and Infoseek (with their customized, human-annotated lists) at a disadvantage. Indeed, subsequent comments from the evaluators highlighted such human-generated "look-and-feel" as important in their experience. Second, the resource lists from the various sources had different lengths; and there was no easy way to give a representative list of (say) 15 good resources from each source. (Yahoo! for instance lists its resources alphabetically, rather than by their judgment of importance).

The participants were asked to use these lists as "starting points" from which to use the Web to learn about the topic as effectively as possible. They were asked to spend 15–30 minutes interacting with the Web, beginning from these lists, in any fashion they chose.

We picked qualitative measures on whose basis participants rated the three resources. Our measures were influenced by the usual concerns Web users express while searching for resources, which are related to the notions of *recall* and *precision* in the Information Retrieval literature. Participants were asked to assign three scores to each of the three resource pages.

Accuracy or precision:

To what extent were the pages in the resource list focused on the query topic at hand? To what extent were they off-topic? On a scale of 1–10, the participants were asked to rate each resource list on its "accuracy" with respect to the given topic (10 = very accurate.)

Comprehensiveness or recall:

With respect to their view of the given topic after 15–30 minutes of searching, the participants were asked to rate how broadly the resource list covered the topic, and to what extent it failed to cover certain broad aspects of the topic. On a scale of 1–10, participants rated each resource list on its "comprehensiveness" with respect to the given topic (10 = very comprehensive.)

Overall value:

On a scale of 1–10, participants rated each resource list on the extent to which it helped lead them to the pages they found most "valuable" in their effort to learn about the topic (10 = very helpful in locating valuable pages.)

Optionally, participants were also encouraged to provide any further comments on their experience using the three resource lists.

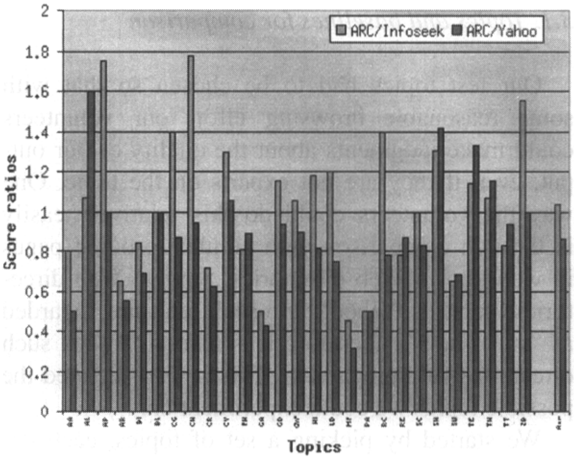


Fig. 1. Ratio of overall ARC scores to Infoseek and Yahoo scores for each of the topics. The blue/light bars represent the ratio of ARC to Infoseek, and the purple/dark bars represent the ratio of ARC to Yahoo. The last pair of bars show the average ratio of scores.

4. Results

4.1. Summary of experimental data

In this section we summarize the information that was thus gathered from the survey. Of the 28 topics, 27 were chosen by at least one volunteer. Overall, 54 records were received, each having nine numerical scores, corresponding to the three sources (ARC, Yahoo, and Infoseek) and the three measures (Accuracy, Coverage, and Overall). There was thus an average of two reviews per topic.

We first study the perceived overall quality of ARC relative to Infoseek and Yahoo. Figure 1 shows the ratio of ARC's score to that of Infoseek and Yahoo. The y-axis value for "indifference" is equal to one; values exceeding one mean ARC is adjudged better, and less than one mean ARC is worse.

From the average score ratios, ARC is adjudged comparable to Infoseek and reasonably close to Yahoo. There is a large variance across topics. We observed that ARC's scores w.r.t. Yahoo and Infoseek are often both favourable or both unfavourable. Some of the topics for which ARC scores relatively poorly are *affirmative action* and *mutual funds*, topics with large Web presence and plenty of carefully hand-crafted resource pages. ARC's best scores were in topics like *cheese*, *alcoholism*, and *Zen Buddhism*, topics that have relatively less Web presence because they are not as connected with political or economic issues.

We emphasize that owing to the small number of evaluators on any single topic, it is hard to infer statistical significance of these ratios on individual topics. However, the two bars marked "Ave" on the right, being an aggregate over a large number of evaluators and topics, may be considered more reliable than individual topics. Next we present in more detail the accuracy and coverage scores. The

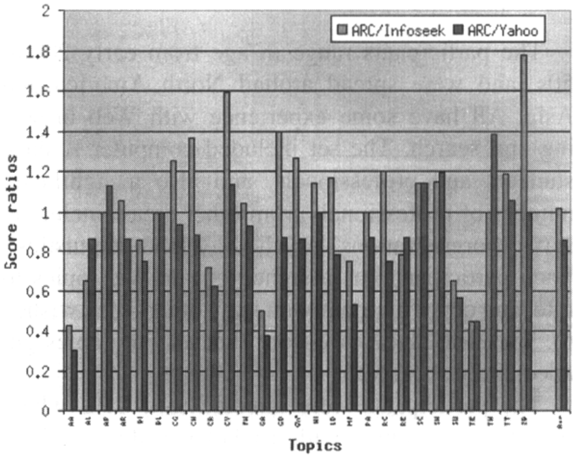


Fig. 2. Ratio of ARC accuracy scores to Infoseek and Yahoo scores.

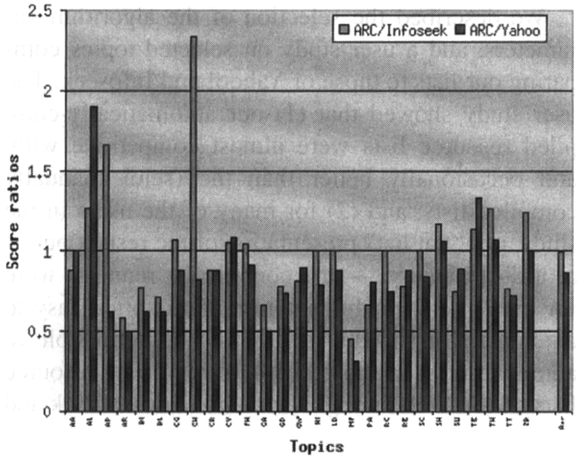


Fig. 3. Ratio of ARC coverage scores to Infoseek and Yahoo scores.

ratio of accuracy scores is shown in Fig. 2 and the ratio of coverage scores is shown in Fig. 3.

Roughly speaking, the accuracy score ratios track the overall ratios (the correlation is over 0.6), whereas ARC found it really challenging to match the perceived coverage of Yahoo and Infoseek. As we discuss later, this seems at least in part a function of annotation and presentation of the links in a proper format. We also show in Fig. 4 a scatter plot of relative accuracy to relative coverage over all volunteers and all topics.

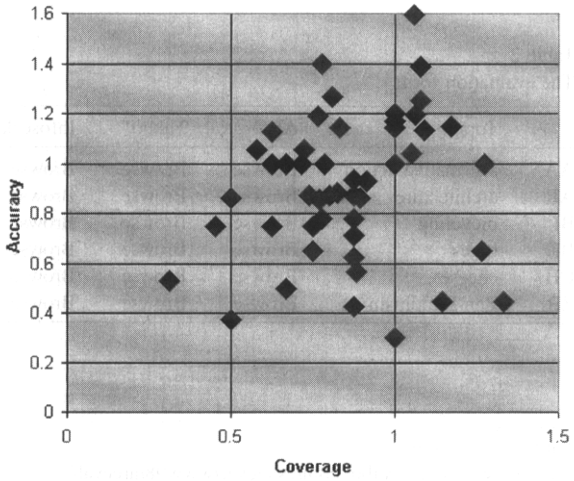


Fig. 4. Scatter plot of relative accuracy to relative coverage over all volunteers and all topics.

There is a mild *positive* correlation (coefficient = 0.26) between these measurements. Both the coverage and the accuracy of ARC's output thus seem to be helped or hurt by topic-specific properties (such as link density, anchor-text coherence, etc.), and less influenced by trade-off parameters like the number of hubs or authorities returned.

4.2. *Summary of comments by evaluators*

In addition to the scores, a number of evaluators provided comments supporting their evaluations and providing suggestions for improving our resource compiler. We now outline the principal themes that emerged in these comments.

On the whole, respondents liked the explicit distinction between hubs and authorities, although some respondents found pages among the authorities that they clearly regarded as hubs. Several found the hub pages more valuable than the authorities as "starting points" from which to begin browsing on one's own.

Virtually all of our evaluators said they used the three resource lists not as ends in themselves, but rather as starting points for further information discovery.

There were cases where evaluators sought resources at levels more narrow or broad than those offered by our system. Controlling the granularity of focus is perhaps the the most useful feature of a hierarchical taxonomy. One wrote, "My goal was to learn about alcoholism as a disease, not to learn about prevention or treatment," whereas for the topic *Gulf war*, one volunteer complained that most links were to Gulf war *syndrome* pages.

By far the most common suggestion for improving our resource lists was their presentation. These suggestions listed one or more of the following features (that we did not provide) as useful:

(1) The position of the topic in a taxonomy (as in Yahoo! and in Infoseek), which provides a context — a visual cue — within which to view the resources listed under that topic; for instance, Yahoo! lists Computer Vision under Science: Computer Science:Computer Vision as the primary taxonomy position.

(2) Yahoo! and Infoseek provide a brief (1–2 lines) summary of each resource page they list. This gives users a quick visual cue of where to con-

tinue their search for further information. These summaries are often the first line or two of the document, but occasionally manually generated after inspection.

(3) Presenting hubs posed a further challenge because, unlike authorities, they may have little character or identity of their own to help the user decide whether to visit them or not.

The first issue is relatively easy to address: if one begins with a taxonomy of topics, it is straightforward to include the position of the topic in this taxonomy when presenting the results. (Some hubs may be broader than, or different from, the topic in the taxonomy; it will also be necessary to eliminate such cases.)

The second issue — that of providing a good summary for a document — remains a fundamental problem in information retrieval; we see no easy way of addressing this gap in a fully-automated resource compiler. If we had the perfect tool to handle the second issue, we might apply it to all the authorities pointed by a hub to generate a presentation for the hub.

In general, we spent little or no effort to compete with the presentation of Yahoo! or Infoseek; this is an interesting area of future work in user interfaces. Indeed, there were documents gathered by our resource compiler for which we did not even extract the appropriate title since it was hidden in an in-line image (rather than in the *title* tag). Our experimental results should therefore be regarded as fairly stringent on our system.

5. Conclusions

Searching for authoritative Web resources on broad topics is a common task for users of the WWW. While there exist manual and semi-manual compilations of authoritative resources by services such as Yahoo! and Infoseek, there has been relatively little work on the full-scale automation of this task. In this paper, we began from this underlying observation, and presented a method for the automated compilation of resource lists, based on a combination of text and link analysis for distilling authoritative Web resources. To our knowledge, this is the first time link and text analysis have been combined for this purpose.

We described the selection of the algorithm parameters, and a user study on selected topics comparing our lists to those of Yahoo! and Infoseek. The user study showed that (1) our automatically-compiled resource lists were almost competitive with, and occasionally better than the (semi-)manually compiled lists; and (2) for many of the users in the study, our "flat list" presentation of the results put us at a disadvantage — the commonest requests were for minor additional annotation that is often easy to automate. Our results suggest that it is possible to automate most of the process of compiling resource lists on the Web through the combination of link and text analysis.

Appendix: ARC evaluation form

Table 2 is a sample from the form sent out to evaluators. We include a **relative link** [4] and an **absolute link** [5] to a larger collection of resources; these may be modified or deleted in future. A sample ARC resource output page is shown in Table 3.

These were the instructions to the evaluators: *The table below lists one topic per row. For each topic, there are links to resource pages in the* **Yahoo!** [6] *and* **Infoseek** [7] *directories, together with the results compiled by our system, ARC. Use the links below to browse the responses and then rate the three systems on a 10-point scale as in the instructions. Please fill out* ***all*** *columns of your chosen rows.*

Table 2
The evaluation form

	Topic	ARC	Yahoo!	Infoseek
AA	affirmative action	Browse	Browse	Browse
AR	architecture	Browse	Browse	Browse
BI	bicycling	Browse	Browse	Browse
BL	blues	Browse	Browse	Browse
CH	cheese	Browse	Browse	Browse
ZB	Zen Buddhism	Browse	Browse	Browse

[4] arceval/index.html

[5] http://www.almaden.ibm.com/cs/clever/www98/arceval/

[6] http://www.yahoo.com

[7] http://www.infoseek.com

Table 3
A sample of ARC output from where the evaluators started exploring

Hubs	Authorities
Hudson Valley Sports Bicycling	VooDoo Cycles
United Bicycle Institute – Industry...	Cannondale.com
Cycling-Related Businesses	Welcome to Syncros
Cavin's Mountain Bike Links	GT Bicycles
Cycling Links	Ibis Cycles
BICYCLING 97	ritcheylogic: home
ALL CYCLING LINKS	ZIPP Speed Weaponry
Alex's Bike Page	Welcome to MARIN BIKES «1998...
BICYCLING 97	Trek Bikes
Bicycling Information Sources	Welcome to the Mavic Home Page
The Ralston Interface – Bicycling	Co-Motion Cycles
BR's Eng. Resource List: Bicycling/...	GIRO Home Page!
Bicycling Info Sources	Reynolds
Bicycling	White Industries Home Page

Acknowledgements

We thank the following people for evaluating the ARC output: Pankaj Agarwal, Andris Ambainis, Arnon Amir, Ari Arjavalingam, Adnan Aziz, Anjay Bajaj, Mukund Balasubramanian, Chandramouli Banerjee, Saugata Basu, Harish Belur, Dan Brown, Gerry Buena, Kanad Chakraborty, Satish Chandra, William Evans, Michael Farrow, Susan Flynn Hummel, Ruchir Garg, Dimitrios Gunopulos, Anshul Gupta, Naveen Gupta, Prosenjit Gupta, Monika Henzinger, Sandy Irani, Haresh Kheskani, P. Krishnan, Lionel Kuhlmann, Ramachandran Manjeri, Carlos Morimoto, Praveen Murthy, Vinod Nagarajan, Rafail Ostrovsky, Sanjay Rajagopalan, Dana Randall, S. Ravi Kumar, Sujit Saraf, Harpreet Sawhney, Leonard Schulman, Raymond Snead, Aravind Srinivasan, Harini Srinivasan, Savitha Srinivasan, David Steele, P. Subbarao, K. Suresh, Hisao Tamaki, and Takeshi Tokuyama.

References

[1] G.O. Arocena, A.O. Mendelzon, and G.A. Mihaila, Applications of a Web query language, in: *Proc. 6th International World Wide Web Conference*, 1997.

[2] J. Carriere, and R. Kazman, WebQuery: Searching and visualizing the Web through connectivity, in: *Proc. 6th International World Wide Web Conference*, 1997.

[3] G.H. Golub and C.F. van Loan, *Matrix Computations*, 2nd edition. The Johns Hopkins University Press, Baltimore, MD, ISBN 0-8018-3739-1, 1989.

[4] J. Kleinberg, Authoritative sources in a hyperlinked environment, in: *Proc. ACM-SIAM Symposium on Discrete Algorithms*, 1998, to appear; also appears as IBM Research Report RJ 10076(91892) May 1997, (http://www.watson.ibm.com:8080/main-cgi-bin/search_paper.pl/entry_ids=8690) and as http://www.cs.cornell.edu/home/kleinber/auth.ps

[5] S. Brin and L. Page, The anatomy of a large-scale hypertextual Web-search engine, in: *Proc. 7th Int. World Wide Web Conference*, Brisbane, Australia, 1998; also *Comput Networks ISDN Systems*, 30(1–7): 107–117, 1998 (this volume).

[6] O.A. McBryan, GENVL and WWWW: Tools for taming the Web, in: *Proc. of the 1st International Conference on the World Wide Web*, 1994.

[7] P. Pirolli, J. Pitkow and R. Rao. Silk from a sow's ear: Extracting usable structures from the Web, in: *Proceedings of ACM SIGCHI Conference on Human Factors in Computing*, 1996.

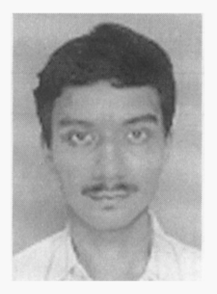
[8] A discussion of search-engine Web-page ranking schemes, at the Search Engine Watch Web site, http://searchenginewatch.com/rank.htm

[9] E. Rivlin, R. Botafogo and B. Shneiderman, Navigating in hyperspace: designing a structure-based toolbox, *Communications of the ACM*, 37(2): 87–96, 1994.

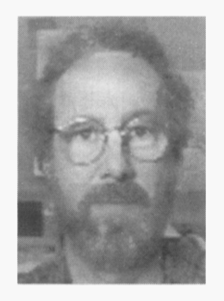
[10] E. Spertus, ParaSite: mining structural information on the Web, in: *Proc. 6th International World Wide Web Conference*, 1997. http://atlanta.cs.nchu.edu.tw/www/PAPER206.html

[11] R. Weiss, B. Velez, M. Sheldon, C. Nemprempre, P. Szilagyi and D.K. Giffor, HyPursuit: a hierarchical network search engine that exploits content-link hypertext clustering, in: *Proc. of the 7th ACM Conference on Hypertext*, 1996.

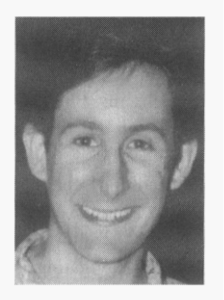
[12] H.D. White and K.W. McCain, Bibliometrics, in: *Ann. Rev. Info. Sci. and Technology*, Elsevier, 1989, pp. 119-186.



Soumen Chakrabarti received his B.Tech in Computer Science from the Indian Institute of Technology, Kharagpur, in 1991 and his M.S. and Ph.D. in Computer Science from UC Berkeley in 1992 and 1996. His research interests include information retrieval and on-line scheduling of parallel information servers. Home page: http://http.cs.berkeley.edu/~soumen/

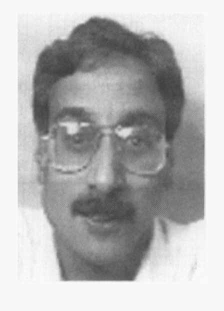


Byron Dom received a Ph.D. in Applied Physics from The Catholic University of America in 1977. He is currently manager of Information Management Principles in the Computer Science Department at the IBM Almaden Research Center in San Jose, CA, where he has been a Research Staff Member since 1987. Prior to that he was a Research Staff Member at the Thomas J. Watson Research Center in Yorktown Heights, NY. His research interests are in information retrieval, machine learning, computer vision, and information theory. He has led several successful projects to develop automatic inspection systems for aspects of computer manufacturing. He is the recipient of several IBM awards for technical excellence; has served as conference chair, session chair and on the program committees of several computer vision conferences; and has served as Associate Editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI).



David Gibson is a Ph.D. student at the University of California, Berkeley, where he pursues fundamental aspects of computer science, and experimental computation. He is a part-time researcher at IBM's Almaden Research Center. Home page: http://www.cs.berkeley.edu/~dag/

Jon Kleinberg received an A.B. from Cornell University in 1993 and a Ph.D. in computer science from MIT in 1996. He subsequently spent a year as a Visiting Scientist at the IBM Almaden Research Center, and is now an Assistant Professor in the Department of Computer Science at Cornell University. His research interests include algorithms, discrete and combinatorial optimization, and geometric methods for clustering and indexing. He is supported in part by an Alfred P. Sloan Research Fellowship and by NSF Faculty Early Career Development Award CCR-9701399. Home page: http://www.cs.cornell.edu/faculty/annual_reports/kleinberg.htm



Prabhakar Raghavan received his Ph.D. in Computer Science from the University of California, Berkeley in 1986. Since then he has been a Research Staff Member at IBM Research, first at the Thomas J. Watson Research Center in Yorktown Heights, NY, and since 1995 at the Almaden Research Center in San Jose, California. He is also a Consulting Professor at the Computer Science Department, Stanford University. His research interests include algorithms, randomization, information retrieval and optimization. He is the co-author of a book **Randomized Algorithms**[8] and numerous technical papers. He has served on the editorial boards and program committees of a number of journals and conferences. Home page: http://www.almaden.ibm.com/cs/people/pragh/pragh.html

Sridhar Rajagopalan has received a B.Tech. from the Indian Institute of Technology, Delhi, in 1989 and a Ph.D. from the University of California, Berkeley in 1994. He spent two years as a DIMACS postdoctoral fellow at Princeton University. He is now a Research Staff Member at the IBM Almaden Research Center. His research interests include algorithms and optimization, randomization, information and coding theory and information retrieval.

[8] http://www.cup.org/Reviews&blurbs/RanAlg/RanAlg.html