Automatic Web Resource Discovery

Soumen Chakrabarti*

IBM Almaden Research Center

1 Introduction

Classical information retrieval (IR) is concerned with indexing a collection of documents and answering queries by returning a ranked list of relevant documents [10, 16, 18]. With the growth of the web, the problems of ambiguity, context sensitivity, synonymy and polysemy that are inherent in natural languages, together with the *abundance* of web pages related to prominent topics, have exacerbated the difficulty of narrowing down the user's information need.

Most search sites have added directory-based topic browsing. The web is organized as a tree of topics, similar to the Dewey decimal system, the Library of Congress catalog, or the Patent and Trademarks Office subject codes. Tree nodes are maintained by ontologists, ranging from paid college grads (Yahoo!) to specialist volunteers (The Mining Co.). Directory sites are very popular. Because of human judgement, pages placed in a topic directory tend to be not only relevant, but also exemplary and influential. However, human judgement is slow, subjective and noisy. It is labor-intensive to keep these directories comprehensive. The Yahoo! strategy is adequate for common users and shallow topics. However, the information needs of web-savvy users who are using the web for serious research for extended periods are very hard to address. The Mining Co. strategy may be biased because of sparsity of experts; at any rate it is biased away from the most accomplished and busiest people.

2 Topic distillation

The web is an example of a *social network*. Social networks have been extensively researched [19]. Social networks are formed between academics by co-authoring, advising, serving on committees; by movie personnel by directing and acting; by musicians, football stars, friends and relatives; infection transmitted by contact; papers citing other papers, and web pages hyperlinked to other web pages. Social network theory is in part concerned with finding influential nodes in the graph, representing important papers, core of epidemics, etc.

2.1 Social network analysis in IR

IR literature includes insightful studies of citation, co-citation, and influence of academic publication [14]. Starting in 1996, a series of applications of social network analysis were made to the web graph, with the purpose of identifying the most authoritative sites and pages about a user query.

Google: If one wanders on the web for infinite time, following a random link out of each page, then different pages will be visited at different rates; popular pages with many in-links will tend to be visited more often. PageRank and Google, invented by Brin and Page [3] crawl the web and simulate such a random walk on the web graph in order to estimate the visitation rate, which is used as a score of popularity. Given a keyword query, matching documents are ordered by this score.

*Email: `soumen@cs.berkeley.edu`

HITS: Hyperlink induced topic search (HITS) [12] is slightly different: it does not crawl or pre-process the web, but depends on a search engine. A query to HITS is forwarded to Alta Vista, which retrieves a subgraph of the web whose nodes (pages) match the query. Pages citing or cited by these pages are also included. This expanded graph is analyzed for popular nodes using a procedure similar to Google, the difference being that not one, but two scores emerge: the measure of a page being an authority, and the measure of a page being a *hub* (a compilation of links to authorities).

ARC and CLEVER: HITS's graph expansion sometimes leads to topic *contamination* or *drift.* E.g., the community of *movie awards* pages on the web is closely knit with highly cited (and to some extent relevant) home pages of movie *companies.* Although *movie awards* is a finer topic than *movies*, the top movie companies emerge as the victors upon running HITS. This is partly because in HITS (and Google) all edges in the graph have the same importance. Contamination can be reduced by recognizing that hyperlinks that contain *award* or *awards* near the anchor text are more relevant for this query than other edges. Such heuristic modification of edge weights significantly improve the quality of query results. In user studies, the results compared favorably with lists compiled by humans, such as Yahoo! and Infoseek [6].

Bharat and Henzinger: Another way to integrate textual content to avoid contamination of the graph is to model each page according to the "vector space" model [18], and then prune the graph expansion at nodes whose corresponding vectors are outliers with respect to the set of vectors corresponding to documents directly retrieved from the search engine [2]. Impressive improvements in precision have been observed.

2.2 Discussion

In principle, all the topic distillation systems discussed above can handle ad-hoc queries, because of the dependence on a search engine in the first step. In practice, these systems produce good answers if the query response induces a graph that is dense enough to derive robust popularity scores. Because it uses a precomputed page rank independent of the query, Google is the fastest system. Graph construction at query time is involved in all the other methods. Bharat and Henzinger describe several heuristics that cut down the query time substantially.

3 Portholes and focused crawling

All the distillation systems depend on large, comprehensive web crawls and indices. The best crawlers running on heavy-duty servers can cover 35–40% of the web, currently over 340 million pages [1]. Googol has crawled about 60 million pages to date. There is a sobering maturity that size is not everything [11, 17, 15]; seasoned users need deep, specific *portholes*, not shallow, generic *portals.* There is a great need for topic-specific crawlers that build focused libraries of web resources. We will first discuss why this cannot be done using topic distillation and then discuss a new approach called *focused crawling.*

3.1 Exploiting topic distillation

Topic distillation systems work well for well-connected communities concerning broad topics. It is thus tempting to use a topic distillation system to generate a web taxonomy in the following trivial way: with each node in the taxonomy, associate a keyword query that describes the topic, and run a distillation program. While the simplicity is appealing, it is not easy to make this succeed.

First, constructing the query involves trial and error, and a fair amount of thought. E.g., we needed the query `+"power suppl*" "switch* mode" smps -multiprocessor* "uninterrupt* power suppl*" ups -parcel` to do a good job for `/Business&Economy/Companies/Electronics/PowerSupplies`. (`SMPS` stands for Switch Mode Power Supply, but also matches pages containing `SMPs`, or Symmetric Multi-Processors! Similarly with UPS and parcel.) In a study with 966 nodes from Yahoo!, in order to match (in the opinion of blind testers) the quality of Yahoo!, queries had to be tuned by hand until the average query had 7.03 terms and 4.34 operators (`"+-*`), in sharp contrast to the average Alta Vista query having 2.35 words and only 0.41 operators. These queries are not a one-time effort, because inclusion of additional topic vocabulary, which may not be known a priori, improved the results. E.g., good results were obtained for “European Airlines” using the query `+lufthansa +iberia +klm` (the fourth response from Alta Vista was itself a hub).

The second issue arises from the aforesaid susceptibility to contamination from popular but irrelevant nodes. The contamination problem can be addressed in a few ad-hoc ways: stop-sites, term weighting and edge weighting. Stop-sites are nodes forcibly removed from the graph before the iterations. Query terms can be assigned weights by humans to be used for better ranking. The weights of links incident with example pages, or lexically close to links to example pages, can be increased artificially. These fixes are ad-hoc; there are no principles for setting edge weights guided by a precise model of hyperlinkage.

3.2 Learning from example

The most successful way to combat contamination has been the use of examples. In 86% of the test cases, specifying an example page improved the results.

An example page offers more than just a forced node in the web graph (as it was used above). It has textual content, and is linked to neighbors with more textual content. In fact, extensive literature in relevance feedback, automatic feature selection and classification suggests that given examples, it is not even necessary to provide a keyword query—the learning process will implicitly recognize the important terms and compute the decision boundaries that can be used to determine whether a given web page is relevant to a topic.

These decision boundaries can be derived implicitly and without any human effort, given the example documents. Most types of classifiers, such as nearest neighbor (NN) [4], bayesian [5, 13], support vector [9], are capable of more reliable discrimination than keyword search, even if boolean constructs were used. (Boolean search induces hard and brittle rules that tend to overfit, whereas well-designed learning algorithms protect against overfitting.)

In the hypertext domain, it is extremely important to build models and classifiers that take link-based features into account. The topic of a page influences its text and the topics of pages in its neighborhood. The latter influence induces circularity, but this can be resolved by an *iterative relaxation* algorithm such as HyperClass [7]. Classification error reduced from 36% to 21% in an experiment with US Patents. With Yahoo!, a more dramatic reduction from 69% to 20% was observed.

3.3 Guiding a focused crawler

Provided a crawler is started off from connected examples of topics, it can be guided and scheduled by HyperClass. The goal of a focused crawler is to selectively seek out pages that are relevant to a pre-defined set of *topics*. The topics are specified not using keywords, but using exemplary documents that are analyzed by HyperClass. The focused crawler analyzes its crawl boundary to find the links that are likely to be most relevant for the crawl. It avoids irrelevant regions of the web. This leads to significant savings in hardware and network resources, and helps keep the crawl more up-to-date.

Extensive focused-crawling experiments have been performed using several topics at different levels of specificity [8]. The system acquires relevant pages steadily while standard crawling quickly loses its way, even though they are started from the same root set. Focused crawling is robust against large perturbations in the starting set of URLs. It discovers largely overlapping sets of resources in spite of these perturbations. In contrast with topic distillation systems, it is also capable of exploring out and discovering valuable resources that are dozens of links away from the start set, while carefully pruning the millions of pages that may lie within this same radius. These results suggest that focused crawling is very effective for building high-quality collections of web documents on specific topics, using modest desktop hardware.

References

[1] K. Bharat and A. Broder. A technique for measuring the relative size and overlap of public web search engines. In *7th World-Wide Web Conference (WWW7)*, 1998. Online at http://www7.scu.edu.au/programme/fullpapers/1937/com1937.htm; also see an update at http://www.research.digital.com/SRC/whatsnew/sem.html.

[2] K. Bharat and M. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 104–111, Aug. 1998. Online at ftp://ftp.digital.com/pub/DEC/SRC/publications/monika/sigir98.pdf.

[3] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the 7th World-Wide Web Conference (WWW7)*, 1998. Online at http://decweb.ethz.ch/WWW7/1921/com1921.htm.

[4] E. Brown. Execution performance issues in full-text information retrieval. Technical Report TR95-81, University of Massachusetts, Amherst, 1995. Online at ftp://ftp.cs.umass.edu/pub/techrept/techreport/1995/UM-CS-1995-081.ps.

[5] S. Chakrabarti, B. Dom, R. Agrawal, and P. Raghavan. Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. *VLDB Journal*, Aug. 1998. Invited paper, online at http://www.cs.berkeley.edu/~soumen/VLDB54_3.PDF.

[6] S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, P. Raghavan, and S. Rajagopalan. Automatic resource compilation by analyzing hyperlink structure and associated text. In *7th World-wide web conference (WWW7)*, 1998. Online at http://www7.scu.edu.au/programme/fullpapers/1898/com1898.html.

[7] S. Chakrabarti, B. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In *SIGMOD Conference*. ACM, 1998. Online at http://www.cs.berkeley.edu/~soumen/sigmod98.ps.

[8] S. Chakrabarti, M. van den Berg, and B. Dom. Focused crawling: a new approach to topic-specific resource discovery. Submitted to the World Wide Web Conference, Jan. 1999. Online at http://www.cs.berkeley.edu/~soumen/www8focus.pdf.

[9] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. In *7th Conference on Information and Knowledge Management*, 1998. Online at http://www.research.microsoft.com/~jplatt/cikm98.pdf.

[10] W. B. Frakes and R. Baeza-Yates. *Information retrieval: Data structures and algorithms.* Prentice-Hall, 1992.

[11] D. Gillmor. Small portals prove that size matters. San Jose Mercury News, Dec. 1998. Online at http://www.sjmercury.com/columnists/gillmor/docs/dg120698.htm and http://www.cs.berkeley.edu/~soumen/focus/DanGillmor19981206.htm.

[12] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *ACM-SIAM Symposium on Discrete Algorithms*, 1998. Online at http://www.cs.cornell.edu/home/kleinber/auth.ps.

[13] D. Koller and M. Sahami. Hierarchically classifying documents using very few words. In *International Conference on Machine Learning*, volume 14. Morgan-Kaufmann, July 1997. Online at http://robotics.stanford.edu/users/sahami/papers-dir/ml97-hier.ps.

[14] R. Larson. Bibliometrics of the world wide web: An exploratory analysis of the intellectual structure of cyberspace. In *Annual Meeting of the American Society for Information Science*, 1996. Online at http://sherlock.berkeley.edu/asis96/asis96.html.

[15] D. Lidsky and N. Sirapyan. Find it on the web. ZDNet, Jan. 1999. Online at http://www.zdnet.com/products/stories/reviews/0,4161,367982,00.html and http://www.cs.berkeley.edu/~soumen/focus/Lidsky_0_4161_367982_00.html.

[16] G. Miller, R. Beckwith, C. FellBaum, D. Gross, K. Miller, and R. Tengi. Five papers on WordNet. Online at ftp://ftp.cogsci.princeton.edu/pub/wordnet/5papers.pdf, Princeton University, Aug. 1993.

[17] M. Mirapaul. Well-read on the web. The New York Times, Dec. 1998. Online at http://www.nytimes.com/library/tech/98/12/circuits/articles/24port.html and http://www.cs.berkeley.edu/~soumen/focus/MatthewMirapaul19981224.html.

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

[19] S. Wasserman and K. Faust. *Social Network Analysis.* Cambridge University Press, 1994.