DiscoWeb: Applying Link Analysis to Web Search

Brian D. Davison, Apostolos Gerasoulis, Konstantinos Kleisouris,
Yingfang Lu, Hyun-ju Seo, Wei Wang, and Baohua Wu
Department of Computer Science, Rutgers University
{davison,gerasoul,kkonst,luyf,hseo,ww,baowu}@cs.rutgers.edu

How often does the search engine of your choice produce results that are less than satisfying, generating endless links to irrelevant pages even though those pages may contain the query keywords? How often are you given pages that tell you things you already know? While the search engines and related tools continue to make improvements in their information retrieval algorithms, for the most part they continue to ignore an essential part of the web – the links. We have found that link analysis can have significant contributions to web page retrieval from search engines, to web community discovery, and to the measurement of web page influence. It can help to rank results and find high-quality *index/hub/link* pages that contain links to the best sites on the topic of interest.

Our work is based on research from IBM's CLEVER project [7, 4, 6], Stanford's Google [3], and the Web Archaeology research [2, 1] at Compaq's Systems Research Center. These research teams have demonstrated some of the contributions that link analysis can make in the web. In our work, we have attempted to generalize and improve upon these approaches. Just as in citation analysis of published works, the most influential documents on the web will have many other documents recommending (pointing to) them. This idea underlies all link analysis efforts, from the straightforward technique of *counting* the number of incoming edges to a page, to the deeper eigenvector analysis used in our work and in those projects mentioned above.

It turns out that the identification of "high-quality" web pages reduces to a sparse eigenvalue of the adjacency matrix of the linked graph [7, 3]. In 1998, Kleinberg [7] provided the analysis and substantial evidence that each eigenvector creates a web clustering which he called "web communities". The most important web community corresponds to the principal eigenvector and the component values within each eigenvector represent a ranking of web pages. Determining the eigenvectors is computationally intensive since the linked graph can be quite large, e.g. each keyword search could result in millions of page hits. For this reason, the approach taken in most implementations is to determine only the principal eigenvector [7, 3], using the well-known power iterative method [5] for eigenvectors. If the initial approximation is the unit vector then the first iteration in the power method corresponds to the *counting* of the incoming and/or outgoing edges for each page of the web graph. There are certain advantages and disadvantages of this approach that we discuss below:

- In Google the power iterative method is applied off-line over the whole web graph. The ranking determined by the first eigenvector is then stored in the database. The major advantage of this approach is that there is no additional run-time link analysis penalty during a query search process. However, there is an open problem with this approach. The rankings will be dominated by "strong" web pages that are irrelevant to the specific search query. For example, pages such as `excite.com` will have a much higher ranking because they have a larger number of incoming edges as compared to e.g. `city.net/countries/greece`. So if `excite.com` is a node in the linked graph it will be ranked much higher than `city.net/countries/greece` when the query is `greece`. Obviously the node `excite.com` can be removed afterwards via text analysis, as it is done in the current version of Google, but the impact on its page ranking needs to be studied further. For example, it is unclear if the the ranking will remain the same if the power method is applied to the subgraph corresponding to the specific query, such as `greece`, in which all irrelevant pages have been removed first.

- Compaq's Connectivity Server and the CLEVER project also compute only the principal eigenvector, but on the linked graph of the neighborhood of pages resulting from a specific query. This substantially reduces the size of the linked graph and as a result it will converge much faster. However, unless the linked graph is weighted correctly the principal eigenvector will not represent the best ranking, as described by Bharat and Henzinger [1]. Even if the linked graph is correctly weighted, the resulting adjacency matrices could be reducible, e.g. the linked

graph consists of disconnected subgraphs, and so their systems might miss highly ranked nodes that are clustered in eigenvectors other than the principal eigenvector.

At Rutgers we have built a prototype called DiscoWeb that consists of a generalization of the method proposed by Kleinberg [7], a page retriever and graph generator, with a Java interface. DiscoWeb determines up to the top 1/4 eigenvalues and eigenvectors associated with the graph. We use an eigenvalue solver that converges quickly, less than a minute for 100 eigenvectors on graphs on the order of 10000 nodes. Preliminary results show a very promising method. By looking at a larger set of eigenvectors we can find clusterings of web pages that are more interesting than the ones extracted by the principal eigenvector (similar to the conclusions derived in the original Kleinberg paper). We can then use heuristics to extract a *global ranking* as well as *local rankings* given by each eigenvector community.

For example, Chakrabarti et. al. [4] concluded that the principal eigenvector for the ARC method, a weighted extension of Kleinberg's power method, scores relatively poorly for the query `affirmative action` as compared to human-edited directories provided by Infoseek and Yahoo. When our prototype was applied to this query, using AltaVista as the source of the pages, the ranking produced many pages that were highly ranked in the Infoseek and Yahoo directories. This result was generated from a rather small sample (only 2211 pages in the neighborhood of the query results). How to select an appropriate eigenvector is still an open research question. We are currently using a heuristic that looks at the highest ranked pages in each eigenvector. For example, what is often considered the "best" page for the `affirmative action` query (`www.auaa.org`) is highly rated in eigenvectors other than the first (2nd, 3rd, etc. depending on DiscoWeb parameters). This page might have been missed if the principal eigenvector were the only one considered because its ranking value is close to zero in that eigenvector.

We believe that there is much promise in the use of link analysis for improving search engine retrievals as well as other tasks. Some of the results that DiscoWeb has produced are quite striking. For example for the query `cancer`, DiscoWeb has found all the three star sites that have been selected by the editors of the Infoseek directory, plus many other important pages. On the other hand AltaVista found only few of the three star Infoseek sites. There are, however, many problems that need to be addressed before *run-time link analysis* can be incorporated into interactive search engine queries. The most important one is the reduction of the computational and retrieval time of the method. An interactive search engine requires response time in seconds. The current DiscoWeb implementation takes few minutes, for graphs of sizes of the order of 10000, which is beyond the interactive search engine constraints. We are currently investigating the reduction of computational and retrieval time via parallelization. However, even in its current form, DiscoWeb can be extremely useful in building high quality web directories automatically, where time constraints are not as strict as those of interactive search engines [4].

References

[1] Krishna Bharat, Andrei Broder, Monika Henzinger, Puneet Kumar, and Sureesh Venkatasubramanian. The connectivity server: Fast access to linkage information on the web. In *Proceedings of the Seventh International World Wide Web Conference*, Brisbane, Australia, April 1998.

[2] Krishna Bharat and Monika R. Henzinger. Improved Algorithms for Topic Distillation in Hyperlinked Environments. In *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 104–111, August 1998.

[3] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the Seventh International World Wide Web Conference*, Brisbane, Australia, April 1998.

[4] Soumen Chakrabarti, Byron Dom, Prabhakar Raghavan, Sridhar Rajagopalan, David Gibson, and Jon M. Kleinberg. Automatic resource compilation by analyzing hyperlink structure and associated text. In *Proceedings of the Seventh International World Wide Web Conference*, Brisbane, Australia, April 1998.

[5] S. D. Conte and Carl de Boor. *Elementary Numerical Analysis: An Algorithmic Approach*. McGraw-Hill, New York, third edition, 1980.

[6] David Gibson, Jon M. Kleinberg, and Prabhakar Raghavan. Inferring web communities from link topology. In *Proceedings of the 9th ACM Conference on Hypertext and Hypermedia (Hypertext'98)*, 1998. Expanded version at http://www.cs.cornell.edu/home/kleinber/.

[7] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA-98)*, pages 668–677, San Francisco, CA, January 1998. Expanded version at http://www.cs.cornell.edu/home/kleinber/.