CRANAI: A New Search Model Reinforced by Combining a Ranking Algorithm with Author Inputs

Jun Lai and Ben Soh
Department of Computer Science and Computer Engineering
La Trobe University Bundoora, VIC, Australia 3083
jun@cs.latrobe.edu.au *ben@cs.latrobe.edu.au*

Abstract

The explosion of information on the Internet has made search engine become the main method for users to find information on the web. In this paper, we propose a new search model by combining a hyperlink-based page ranking algorithm with author inputs. The page ranking algorithm measures page importance by calculating the page weight based on incoming and outgoing hyperlinks in the page. The author input takes into account the weight of relevance in terms of keywords or phrases of a page specified by the page creators. Our evaluation shows that the proposed search method performs better than that using only the page ranking algorithm.

1. Introduction

With the tremendous growth of information on the Internet, more and more people are using search engine as one of the main searching tools. However, the amount of information on the Web is increasing far more quickly than our ability to process it. As of December 2002, the largest search engine claims over 3 billion pages in its index. This figure is exponentially increasing every year. How to precisely return the search results to satisfy a user's interest from such a great deal of information on the Web has become a challenge.

One way to alleviate this issue is to utilize hyperlink-based page ranking algorithm to calculate page importance by the number of other pages that link to it. Page ranking algorithm has made Google so successful [1][2]. The hyperlink analysis has also been well adopted in Web community [3][4][5][6].

However, considering only the importance of page is not enough to bring the best search result to the end users. The page relevance to what the users' searching for in terms of the search query should also be taken into consideration. Good information filtering can successfully indicate the relevance of pages and protect the user from irrelevant information and without missing relevant information [7][8]. Through decade, information filtering methods based on users' behaviors are developed to improve the relevance of pages [9][10]. A link-based method for enhancing relevant judgments in tree-structured hypertext is developed in [11]. All these methods consider the relevance from the users' point of view and push the information that they believe relevant to users based on users' behavior and the analysis of page content they made.

In this paper, we propose a new search model called "combination of hyperlink-based rank algorithm and author input" (CRANAI). Hyperlink-based page ranking algorithm measures page importance by calculating the page weight based on incoming and outgoing hyperlinks in the page. The author input is to take the weight of relevance in terms of keywords or phrases of a page specified by the page creators into consideration.

This paper is organized as follows: in section 2, we give the hyperlink-based page weight measurement. Next, the weighted author input tree (WAIT) is proposed in section 3. Then the combination of hyperlink-based page ranking algorithm and author input is discussed in section 4. Followed by section 5 is the evaluation based upon the consideration of both page importance and relevance using CRANAI. Finally, the conclusions are drawn and future work is discussed in section 6.

2. Hyperlink-based Rank Algorithm

We can view the web as a directed graph which is composed of nodes and paths [1][3][12]. Where nodes are web objects and paths are links between web objects. Figure 1 shows a directed web graph. We define this graph as G = (N, P), where N

Proceedings of the 2005 IEEE International Conference on e-Business Engineering (ICEBE'05)
0-7695-2430-3/05 $20.00 © 2005 IEEE

represents n nodes in the graph, a path $(i,j) \in P$ indicates the path between node i and node j. $Pi \rightarrow j$ is a directed path from node i to node j.

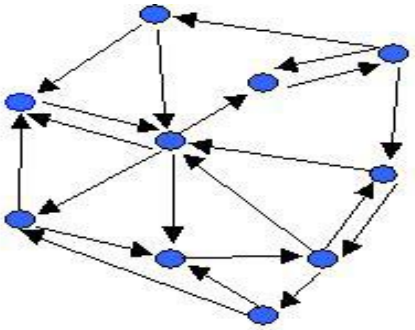

Figure 1. Directed web graph

2.1 Path structure of web graph

The web graph consists of three types of nodes: *hub*, *head* and *tail* nodes. *Hub* nodes are those pages which have many incoming and outgoing links and playing an important multi-junction role in the traffic of web as shown in figure 2 (a). *Head* nodes only contain outgoing links, but do not have incoming links shown in figure 2 (b). *Tail* nodes have incoming links and do not have outgoing links shown in figure 2 (c).

Generally speaking, highly linked pages are more important than pages with few links. A page with a link off a very important page should be ranked higher than those pages with links from obscure places [1]. Simple citation counting has been used to speculate on the future winners of the Nobel Prize [13].

2.2 Page weight measurement

Based on the theory above, the incoming and outgoing links define the importance of a page. Here, we adopt web page weight to measure the importance of a page (node) in the directed web graph. By doing so, we define that each node N_i has two weight vectors. One is N_i^{in} which measures the page weight in term of incoming links, the other one N_i^{out} which indicates the page weight associating with outgoing links.

Figure 3 shows that the directed path $P_{A \rightarrow B}$ is the outgoing link of page A, but the incoming link of page B.

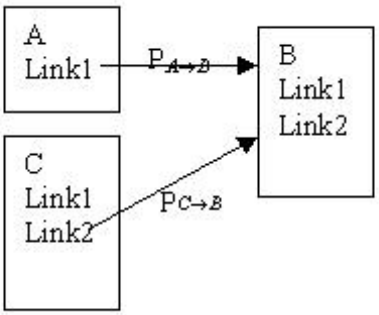

Figure 3. Incoming and outgoing links

Each page has an initial weighting value of 1. Then N_i^{in} and N_i^{out} are defined in the following iterative operations [3].

$$N_i^{in} \leftarrow \sum_{P_{j \rightarrow i}} N_j^{out}$$

$$N_i^{out} \leftarrow \sum_{P_{i \rightarrow j}} N_j^{in}$$

The iteration continues until a fixed point reaches, where the weighting value of each page becomes stable. The incoming weight and outgoing weight vectors are normalized after each iteration.

This page ranking algorithm is not simply counting the number of links in each page, it shows that hyperlink connectivity between pages affects the importance of pages.

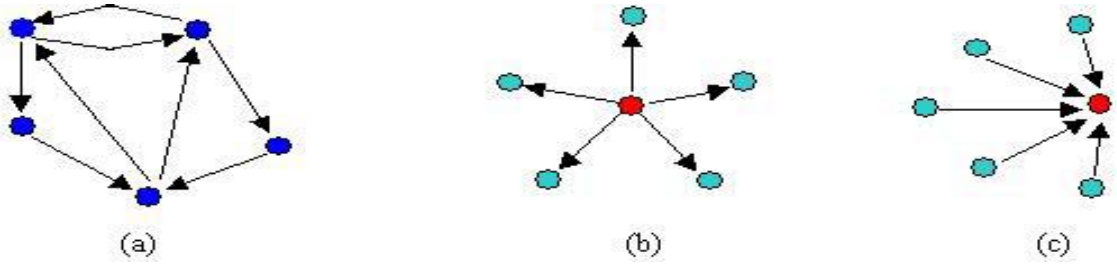

Figure 2. Three typical kinds of nodes in web graph

In order to compute the weighting value of a page, we define the average of N_i^{in} and N_i^{out} as $\overline{N^{in}}$ and $\overline{N^{out}}$ respectively in the following formulas:

$$\overline{N^{in}} = \sum_{i=1}^{n} N_i^{in} / n$$

$$\overline{N^{out}} = \sum_{i=1}^{n} N_i^{out} / n$$

Where n is the total number of nodes in the web graph. Based on N_i^{in} and N_i^{out}, we further define the weight of page N_i as follows:

$$N_i = \frac{N_i^{in} - \overline{N^{in}}}{\sigma_{N^{in}}} \times \frac{N_i^{out} - \overline{N^{out}}}{\sigma_{N^{out}}} \quad (1)$$

Where $\sigma_{N^{in}}$ and $\sigma_{N^{out}}$ are standard deviations and defined as:

$$\sigma_{N^{in}} = \sqrt{\sum_{i=1}^{n} (N_i^{in} - \overline{N^{in}})^2 / n}$$

$$\sigma_{N^{out}} = \sqrt{\sum_{i=1}^{n} (N_i^{out} - \overline{N^{out}})^2 / n}$$

To calculate the weight of a page, we use non-linear mapping through Error function with coefficient β, which is widely used in statistics to map the value of N_i in the formula 1 to the range between 0 and 1 [14].

$$\varphi_{(N_i)} = \mathrm{Erf}(N_i / \beta)$$

Where

$$\mathrm{Erf}(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-u^2/2} \cdot du \quad (2)$$

3. Weighted Author Input Tree

The hyperlink-based rank is also called connectivity-based ranking schemes which can be classified into two categories: query independent and query dependent. The former scheme ranks a page independent of a given query and the latter scheme assigns a score to a page in the context of a given query [21]. We have discussed query-independent ranking in section 2. Carriere and Kazman [22] propose a query-dependent ranking with a neighborhood graph, which only contains pages on the query topic. Other query-dependent ranking schemes are proposed in [3][4][5], in which web pages are ranked by the search engine according to the relevance of the content of the page and the query.

However, no search engines can determine what a page is exactly about better than the author of the page. To date, there have been no approaches of taking author's "say" into consideration. To this end, we propose a weighted author input tree (WAIT). In this approach, author can submit a header file as a part of HTML code which indicates the weight of pages in terms of different keywords or phrases and is defined as follows:

$$\lambda_{(d)} = \{ W_{K_i} \mid K_i \in C,\ 0 <= W_{K_i} <= 1 \} \quad (3)$$

where:

- d denotes a document ID.
- C is a set of all keywords to which the document can be related.
- K_i denotes a keyword.
- W_{K_i} is the pre-defined weight of the keyword K_i.

In WAIT, we represent the author's input by a hierarchical tree with weights associated with each branch of the tree. The author of the page denotes a weight W_{K_i} associating with a keyword or phrase for the page that indicates the relevance of the page and a phrase. Figure 5 shows an example of WAIT in a domain of tourism. The range of weight is denoted between 0 and 1. In this example, it shows the page of activities has weights 0.8, 0.62, 0.8 and 0.6 in terms of keyword skiing, dinner, game room and spa.

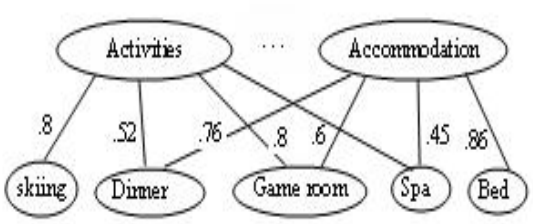

Figure 5. An example of WAIT

By weighting the keywords, the author has prioritized the relevant keywords of their documents. In this approach, each page has more than one weight in terms of the relevance of different keyword or phrase. In other words, W_{K_i} reflects the relevant degree of a page with the phrase of K_i. The bigger

W_{K_i} is, the more relevant the phrase is to the page. For instance, the weight of "skiing" is 0.8, "dinner" is 0.52 and "spa" is 0.6, we can conclude that the phrase skiing is more relevant to the page of activities than the phrase spa. On the other hand, the phrase spa is more relevant to the page than dinner.

How much a phrase is relevant to a page has made the value of λ in formula 3 become crucial. If an author gives equal value to all phrases, basically, this author is not prioritizing any phrases. For example, if an author gives all phrases a weight value of 0.99 in order to get high rank in the search result, this leads to misusing the approach. To tackle this issue, we normalize the weight value as follows:

$$W'_{K_i} = W_{K_i} \Big/ \sum_{i=1}^{n} W_{K_i}$$

Therefore, formula 3 should be re-written as follows:

$$\lambda_{(d)} = \{ W'_{K_i} \mid K_i \in C,\ 0 <= W'_{K_i} <= 1 \} \qquad (4)$$

Where $\lambda_{(d)}$ is composed of normalized weight W'_{K_i}.

For example, page of activities in figure 4 can have weights as follows:

$$\lambda_{(d)} = \{0.8_{K_1}, 0.52_{K_2}, 0.8_{K_3}, 0.6_{K_4} \ldots\}$$

By normalizing the weight value with formula 4, we prevent the system from being misused.

4. Combination of Ranking Algorithm and Author Input (CRANAI)

Based on formula 2 and 4, the combination of ranking algorithm and author input is defined as follows:

$$V = \varphi_{(N_i)} \times \lambda_{(d)} \qquad (5)$$

Where $\varphi_{(N_i)}$ is the weight of web page given in formula 2, and $\lambda_{(d)}$ is the author input weight and given in formula 3.

We can also express formula 5 in a matrix form as follows:

$$V_{n,m} = \varphi_n \times \lambda_m \qquad (6)$$

Where

- *n* is the number of pages.
- *m* is the number keywords or phrases in the author input.

That is,

$$V_{n,m} = \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \ldots \\ \varphi_n \end{bmatrix} \times \begin{bmatrix} \lambda_1 & \lambda_2 & \ldots & \lambda_m \end{bmatrix}$$

$$= \begin{bmatrix} \varphi_1\lambda_1 & \varphi_1\lambda_2 & . & . & \varphi_1\lambda_m \\ \varphi_2\lambda_1 & \varphi_2\lambda_2 & . & . & \varphi_2\lambda_m \\ \ldots & . & . & . & . \\ \varphi_n\lambda_1 & .\varphi_n\lambda_2 & . & . & \varphi_n\lambda_m \end{bmatrix}$$

From the above matrix, the weight of a page associated with a particular phrase can be easily located. For example, the weight of page 1 associated with phrase of "Game room" which is phrase No.3 will be $V_{1,3}$.

Table 1. The weights of web pages

Node	φ	λ_1	λ_2	λ_3	λ_4	λ_5	V_1	V_2	V_3	V_4	V_5	R
1	0.57	0.76	0.34	0.94	0.76	0.79	0.433	0.194	0.536	0.433	0.450	7
2	0.89	0.99	0.69	0.45	0.89	0.52	0.881	0.614	0.400	0.792	0.463	2
3	0.72	0.57	0.93	0.86	0.22	0.78	0.410	0.670	0.620	0.158	0.562	4
4	0.95	0.88	0.54	0.37	0.54	0.91	0.836	0.513	0.352	0.513	0.865	1
5	0.31	0.25	0.79	0.81	0.99	0.44	0.078	0.245	0.251	0.307	0.136	9
6	0.69	0.50	0.36	0.39	0.80	0.94	0.345	0.248	0.269	0.552	0.649	5
7	0.18	0.33	0.94	0.57	0.85	0.90	0.059	0.169	0.103	0.153	0.162	10
8	0.85	0.93	0.11	0.20	0.64	0.78	0.791	0.094	0.170	0.544	0.663	3
9	0.63	0.52	0.97	0.67	0.68	0.45	0.328	0.611	0.422	0.428	0.284	6
10	0.35	0.34	0.61	0.79	0.94	0.27	0.119	0.214	0.277	0.329	0.095	8

5. Evaluation

The evaluation is carried out in a simulating environment. There are 100 nodes and 5369 paths in the web graph. For simplicity, Max (m) in formula 6 is denoted as 5. Table 1 shows the result of top 10 nodes. φ is the weight of pages; λ is the author input weight in terms of different keywords; V_i represents the weight of the combination of φ and λ. For instance, in the first row of the table, $V_1=\varphi_1\lambda_1$, $V_2=\varphi_1\lambda_2$, etc. R reflects the original rank based on φ without considering the author input. As table 1 shows, page No.4 has the highest rank.

The new ranks based upon V_i, the value of the combination of φ and λ are shown in table 2.

R_i in table 2 is ranked based on V_i in the table 1. For example, R_1 is based on V_1, R_2 is based on V_2, etc. As we can see, the new ranks are slightly different from the original ranks in column R which without considering the author input. The page (node) No.4 is not the top highly ranked pages in terms of phrases No.1,2,3 and 4 in columns V_1, V_2, V_3 and V_4.

Table 2. Ranks based on Vi

Node	φ	R	R_1	R_2	R_3	R_4	R_5
1	0.57	7	4	8	2	5	6
2	0.89	2	1	2	4	1	5
3	0.72	4	5	1	1	9	4
4	0.95	1	2	4	5	4	1
5	0.31	9	9	6	8	8	9
6	0.69	5	6	5	7	2	3
7	0.18	10	10	9	10	10	8
8	0.85	3	3	10	9	3	2
9	0.63	6	7	3	3	6	7
10	0.35	8	8	7	6	7	10

6. Conclusions and Future Work

In this paper, we propose a new search model by combining hyperlink-based rank algorithm and author input. This approach considers the Web as a directed graph. Hyperlink-based page ranking algorithm measures page importance by calculating the page weight based on incoming and outgoing hyperlinks in the page. The author input takes into consideration the relevance in terms of phrase. Our proposed approach considers not only the importance of a page, but also the relevance of a page associated with query phrase. Our evaluation shows that the proposed search method performs better than that using only the page ranking algorithm.

Other factors besides hyperlinks, also determines the importance of web pages. The formula 4 and 6 can be optimized by adding a proper coefficient which is worth investigating. Our future work also includes getting users to evaluate the proposed approach because the quality of a search method necessarily involves human factors.

7. References

[1] L.Page, S.Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web". Technical report, Stanford University Database Group, Jan. 1998. http://dbpubs.stanford.edu/pub/1999-66

[2] D.Winder., "Better than Google", PCAuthority, Issue.89, Apr 2005, pp.181-122.

[3] Kleinberg, J.: "Authoritative Sources in a Hyperlinked Environment", Journal of the ACM(JACM). Vol.46, No.5, 1999

[4] R.Botafogo, E.Rivlin, B.Shneiderman, "Structural analysis of hypertext: Identifying hierarchies and useful metrics", ACM Transactions and Information Systems. Vol.10, No.2, 1992

[5] Bharat, K.Henzinger, M.: "Improved Algorithms for Topic Distillation in a Hyperlinked Environment", Proceedings of ACM 21st International SIGIR'98, 1998, pp.104-111

[6] Charkrabarti, S., Dom, B.: "Automatic Resource Compilation by Analysing Hyperlink Structure and Associated Text", Proc. The 7th International World Wide Web Conference, 1998, pp.389-401

[7] Hanani, U., Shapira, B. and Shoval, P., "Information Filtering: Overview of issues, research and systems", User Modeling and User-Adapted Interaction 11, pp. 203-259.

[8] Belkin N.J., and Croft, W.B. (1992), "Information filtering and information retrieval two sides of the same coin", Communications of the ACM, 35(12), pp.29-38.

[9] Masahiro Morita, Yoichi Shinoda, "Information filtering based on user behavior analysis and best match text retrieval", Proc. The 17th annual international ACM SIGIR conference on Research and development in information retrieval, 1994

[10] Peter W. Foltz, Susan T. Dumais, "Personalized information delivery: an analysis of information filtering methods", Communications of the ACM, Volume 35 Issue 12, Dec, 1992

[11] M. E. Frisse, "Searching for Information in a Hypertext medical handbook", Communications of the ACM, Vol.31, No.7, 1988

[12] Angelaccio, M.; Buttarazzi, B., "Local searching the internet", Internet Computing, IEEE Vol.6, No.1, Jan.-Feb. 2002, pp:25 – 33

[13] N. Sankaran, "Speculation in the biomedical community abounds over likely candidates for nobel", The Scientist. 9 (19), Oct 1995. http://www.the-scientist.com/1995/10/02/1/1

[14] E.S.Keeping, "Introduction to Statistical Inference", Dover Publication, 1995.

[15] J. Mostafa, S. Mukhopadhyay, M. Palakal, W. Lam, "A Multilevel Approach to Intelligent Information Filtering: Model, System, and Evaluation", ACM Transactions on Information Systems, Vol. 15, No. 4, 1997, pp. 368–399.

[16] Robertson. S. and Walker. S., "Threshold setting in adaptive filtering", Journal of Documentation, 2000, 56, 3:312-331.

[17] J, A Hartigan. "Clustering Algorithms. WILEY Publication, 1975.

[18] Kobayashi, M and Takeda, K, "Information Retrieval on the Web", ESSIR 2000, LNCS 1980, Springer-Verlag 2000, pp. 242-285.

[19] Baeza-Yates, R. and Ribeiro_neto, B., Modern Information Retrieval, Addison-Wesley, Reading, MA, 1999.

[20] Cho, J., Garcia-Molina, H. and Page, L. "Efficient Crawling Through URL Ordering", in Proceedings of the 7th World Wide Web Conference, Brisbane, Australia, 1998.

[21] M.R.Henzinger, "Hyperlink Analysis for the Web", Internet Computing, IEEE Jan.-Feb. 2001, pp.45-50

[22] J. Carriere and R. Kazman, "Webquery: Searching and Visualizing the Web through Connectivity", Proc.6th Int'l WWW Conf., Elsevier Science, New York, 1997, pp.701-711