Computing Personalized PageRanks

Franco Scarselli
Dep. of Information Engineering
University of Siena
Siena, Italy
franco@ing.unisi.it

Ah Chung Tsoi
Office of Pro Vice-Chancellor IT
University of Wollongong
NSW 2522, Australia
act@uow.edu.au

Markus Hagenbuchner
Office of Pro Vice-Chancellor IT
University of Wollongong
NSW 2522, Australia
markus@uow.edu.au

ABSTRACT

A recently published approach to adaptive page rank, using the solution of quadratic optimization methods with a set of simple constraints [3], is modified to permit classification of web pages according to their page contents, URLs. This modification allows the approach to be more adapted to the needs of focussed crawlers, or personalized search engines.

Categories and Subject Descriptors: H.3.3 Information Systems: Information storage and retrieval–*Information search and retrieval*

General Terms: Algorithms, Human factors, Measurement.

Keywords: Interface personalization, PageRank, Search engines.

1. INTRODUCTION

Web page ranking algorithms are used by search engines to arrange the URLs returned in response to a user query. The page rank usually depends on two components: the relatedness of the document to the user query and the "quality" of the document.

A number of algorithms have been proposed which attempt to measure some aspects of document "quality" [2, 5]. PageRank, used by Google [2], is a well-known approach in this class. PageRank introduces the concept of document authority. A page is considered authoritative if it is pointed by many other pages, and, conversely, if the referring pages are authoritative.

While PageRank and most of the other page ranking algorithms are designed for general purpose search engines, some recent approaches provide specialized rankings which are suited to particular requirements of a vertical search engine. Few of those solutions also allow to automatically adapt the score to user needs [3, 4].

In [3], we proposed a method to adapt page ranks to allow modification of PageRank to satisfy a set of user requirements, e.g., one particular page should have a higher rank to others. A critical ingredient in the solution of the ensuing quadratic optimization problem [3], to allow the algorithm to scale to large scale problems as presented by the Internet, is to cluster web pages according to their PageRanks such that pages within a certain range are placed in the same cluster. In this paper, we will modify this approach by allowing clustering based on contents, and some page features, e.g., URLs, anchor texts.

2. ADAPTIVE PAGERANK

The PageRank $\boldsymbol{X} \in \mathcal{R}^n$ is computed as follows:

$$\boldsymbol{X} = d\, W \boldsymbol{X} + (1-d)E \tag{1}$$

where W is an $n \times n$ matrix with elements $w_{i,j} = 1/h_j$ if there is a hyperlink from node j to node i, and h_j is the total number of outlinks of node j, and $w_{i,j} = 0$ otherwise.

The vector $E = [e_1, \ldots, e_n]$, which we will call *forcing factor*, contains the *default energies* e_i assigned to pages [1]. The PageRank [2] is computed by assigning $e_i = 1$ for every i. The PageRanks can be modified by assigning alternative values to E. In [3], we suggested clustering the web pages with similar PageRanks together in the same cluster, which correspond to E having a value of 1 for pages in the same cluster, and 0 otherwise. In this paper, we modify the approach taken in [3] to allow clustering to occur according to page contents, and some page features, e.g., URLs, anchor texts. This corresponds to different E being used: e.g., if we wish to measure the authority of web pages for the topic *wine*, we can set $e_i = 1$ if the ith page is about "wine", and 0 otherwise.

In our approach, the page scores are computed as a linear combination of a set $\boldsymbol{X}^1, \ldots, \boldsymbol{X}^m$ of *specialized rankings*

$$\boldsymbol{X}(p) = \sum_i \alpha^i \boldsymbol{X}^i$$

where the α^i are real parameters and $p = [\alpha^1, \ldots, \alpha^m]$. $\boldsymbol{X}^i$ is the solution of (1) for a particular forcing factor E_i. The ranking $\boldsymbol{X}(p)$, called *adaptive PageRank*, can be personalized by varying the set of parameters p.

2.1 Quadratic optimization

In the following, we assume that user requirements can be formulated as a quadratic optimization problem

$$\begin{array}{l} \min_p p^T H p + t^T h \\ Ap \le b \end{array} \tag{2}$$

where $H \in \mathcal{R}^{n\times n}$, $t \in \mathcal{R}^n$, $h \in \mathcal{R}^n$ $A \in \mathcal{R}^{c\times n}$ and $b \in \mathcal{R}^c$.

In fact, a number of different requirements can be represented by a linear constraint and/or the minimization of a quadratic function. For example, the fact "page i is more important than page j" can be easily enforced by the inequality $vp \le 0$, where $v = [x_j^1 - x_i^1(p), \ldots, x_j^m - x_i^m(p)]$, and $x_i(p)$ and x_i^j denote the i-th components of vectors $\boldsymbol{X}(p)$ and $\boldsymbol{X}^j$, respectively. Furthermore, the approach is more readily suited to implement constraints such as "the sum of all scores of a Web site cannot exceed L", or personalized constraints such as "increase the PageRank of pages addressing Wine". In addition, the quadratic function in (2) can be used to represent more user requirements, e.g., it may be useful to keep $\boldsymbol{X}(p)$ close to the PageRank X^{pr}. A solution consists of inserting X^{pr} into the set of specialized rankings. Then, we can keep

WWW2004, May 17–22, 2004, New York, New York, USA.
ACM 1-58113-912-8/04/0005.

α^{pr} close to 1 and the other parameters close to 0 as follows:

$$\begin{array}{ll} \min_p (1-\alpha^{pr})^2 & \text{subject to} \\ \sum_i \alpha^i = 1 & \\ \alpha^i \geq 0, \text{for each } i & \end{array} \tag{3}$$

More generally, the quadratic function can be used to represent soft constraints. For example, (2) may not be satisfiable due to the presence of contradictory requirements. In this case, some requirements can be moved into the quadratic function and represented by soft constraints (see [3] for more details).

Our approach consists of three steps: (a) compute a set of specialized rankings; (b) transform user requirements into an optimization problem the solution of which produces the optimal parameters $\bar{p}$; and (c), $X(\bar{p})$ is used to produce customized results.

Notice that step (a) is computationally expensive, since it consists of the computation of m PageRanks, but must be carried out once for a given set of documents. Step (b) must be carried out for every user. The worst case computational effort needed to solve problem (2) is $O(c^2m^2)$ where c, the number of constraints, and m, the number of the specialized rankings are small. In addition, we were able to confirm claims made in [6] that in practice the algorithm performs much better than the worst-case bound . For this reason, the method is suitable for building personalized rankings. Finally, step (c) has a low computational cost provided that $B(p)$ is not stored and the components of $B(p)$ are computed on line from the specialized rankings.

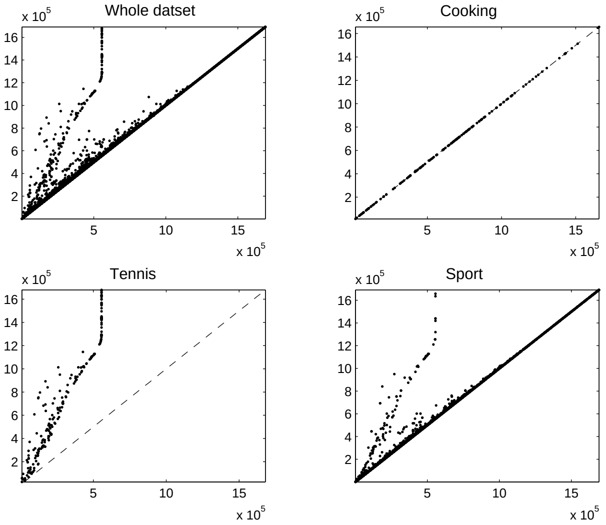
3. EXPERIMENTAL RESULTS

The approach was evaluated on the dataset WT10G, which contains $1,692,096$ Web pages downloaded from $11,680$ servers. Nine topics were considered: “Tennis”, “Sports”, “Linux”, “Windows”, “Cooking”, “Wine”, “Recipes”, “Surgery”, “Cancer”. For sake of simplicity, the pages were classified using their URLs. For example, the class “cancer” consists of the URLs that contained the words “cancer” or “tumor”.

Acting as a user interested in tennis, we selected three pages on tennis and designed three constraints to increase their scores. The constraints consisted of inequalities which hold the adaptive PageRank to be at least three times larger than PageRank. Moreover, in order to keep the scores of the documents as close as possible to their PageRanks, the optimization problem contained also the quadratic function and constraints in (3).

Figure 1 shows the results achieved by the algorithm. The table confirms that the scores of the three pages were actually tripled. Moreover, for each pages, the absolute position in the ordering established by the adaptive PageRank versus the position determined by PageRank is shown. In fact, each point in the graph stands for a page. The dashed line corresponds to the line $y = x$, and the points above such a line represented pages which have gained higher ranks using adaptive PageRank, whereas the points under it represent pages which have achieved worse ranks.

The top left hand graph in Figure 1 plots all the pages, the others graphs plot the pages which belongs to tennis, sport and cooking, respectively. The “tennis” plot shows that most of the pages on this topic gained a higher rank. On the other hand for “sports”, which is a related topic, there are different kinds of behaviors. There is a set of pages whose distribution closely resemble those observed in “tennis”. In fact, some pages on sports belongs also to the class “tennis” and received the same scores. Moreover, there are pages displayed above the line $y = x$: those documents are not related to the tennis pages. Finally, some documents are not about “tennis”, but they are pointed by pages on tennis. In fact, they have an intermediate behavior and lay just between the dashed line and the pages similar to “sports”.



	$page\,1$	$page\,2$	$page\,3$
$PageRank$	7.56	5.602	7.06
$Adaptive$	22.69	20.18	22.19

Figure 1: PageRanks before and after optimization.

For the documents in classes completely unrelated to tennis, the rankings established by adaptive PageRank and PageRank are close. Figure 1 show the results for the class “Cooking”. The plots of the other topics are similar.

Due to space limitations, we do not include other experiments. However, the approach has been tested with similar results also using different document features (e.g. features distinguishing between homepages, index pages, and so on) and different constraints (e.g. inequalities that force the score of a page to be larger than the score of another page).

4. CONCLUSIONS

We have presented a new approach to the personalization of PageRank. A user profile is computed by solving a quadratic optimization problem which represents the user requirements. The experiments demonstrated the viability of the method. Moreover, the experiments show that the user requirements can be expressed by constraints on a few sample pages, since the algorithm is able to generalize the requirements to the whole document set. The method extends a previous works [3], in that the optimization of PageRank parameters is carried out considering also the page content and the features of the document.

Further experiments are currently being conducted to investigate the behavior of our approach to more complex constraints.

5. REFERENCES

[1] Bianchini, M., Gori, M., Scarselli, F. “Inside PageRank”, *ACM Trasanctions on Internet Technology*, 2004 (to appear).

[2] Brin, S., Page, L. “The anatomy of a large scale hypertextual web search engine”. *Proceedings of the 7th WWW conference*, April, 1998..

[3] Tsoi, A. C., Morini, G., Scarselli, F., Hagenbuchner, M., Maggini, M. “Adaptive ranking of Web pages”, *in Proceedings of the 12th WWW Conference*, 2003.

[4] Chang, H., Cohn, D., McCallum A.K., “Learning to Create Customized Authority Lists”, *Proc. 17th International Conf. on Machine Learning*, 2000.

[5] Kleinberg J., “Authoritative sources in a hyperlinked environment”, *Proc. 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[6] Vandenberghe, L., Boyd S., “Semidefinite programming”, SIAM, 38(1): 49-95, March 1996.