Journal of Computational and Applied Mathematics 224 (2009) 702–708

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

On computing PageRank via lumping the Google matrix

Yiqin Lin [a], Xinghua Shi [b], Yimin Wei [b,c,*]

[a] Department of Mathematics and Computational Science, Hunan University of Science and Engineering, Yongzhou 425100, PR China
[b] Institute of Mathematics, School of Mathematical Sciences, Fudan University, Shanghai 200433, PR China
[c] Key Laboratory of Mathematics for Nonlinear Sciences (Fudan University), Ministry of Education, PR China

ARTICLE INFO

Article history:
Received 12 January 2007
Received in revised form 19 May 2008

MSC:
65B99
65F10
65F15
65F50

Keywords:
PageRank
Dangling node
Weakly nondangling node
Power method
Google matrix
Lumping

ABSTRACT

Computing Google's PageRank via lumping the Google matrix was recently analyzed in [I.C.F. Ipsen, T.M. Selee, PageRank computation, with special attention to dangling nodes, SIAM J. Matrix Anal. Appl. 29 (2007) 1281–1296]. It was shown that all of the dangling nodes can be lumped into a single node and the PageRank could be obtained by applying the power method to the reduced matrix. Furthermore, the stochastic reduced matrix had the same nonzero eigenvalues as the full Google matrix and the power method applied to the reduced matrix had the same convergence rate as that of the power method applied to the full matrix. Therefore, a large amount of operations could be saved for computing the full PageRank vector.

In this note, we show that the reduced matrix obtained by lumping the dangling nodes can be further reduced by lumping a class of nondangling nodes, called weakly nondangling nodes, to another single node, and the further reduced matrix is also stochastic with the same nonzero eigenvalues as the Google matrix.

1. Introduction

With the booming development of the Internet, web search engines have become the most important Internet tools for retrieving information. Among thousands of web search engines based on various algorithms that have emerged in recent years, Google has become the most popular and successful one. Google's success should largely be attributed to its simple but elegant algorithm: PageRank. The core of the PageRank algorithm involves computing the PageRank vector, which is the stationary distribution of the so-called Google matrix and a measurement of the importance of the web pages. The dimension of the Google matrices exceeds 11.5 billion, so only a small set of algorithms for computing its stationary distribution can be applied.

A number of numerical methods have been studied for computing the PageRank vector. In spite of its low efficiency, the simple power method stands out for its stable and reliable performances (cf. [16]). To remedy the slow convergence of the power method, some acceleration techniques have been proposed, which include extrapolation [2,4,8,10], aggregation/disaggregation [7,9,12], lumping [15], adaptive methods [9]. Moreover, an Arnoldi-type method has been considered [3]; the power–Arnoldi algorithm can be found in [18]. The Jordan canonical form of the Google matrix has been investigated in [19,20].

We review the original ideal of Google's PageRank [16]. On the basis of the hyperlink structure of the web, the web can be viewed as a direct graph, in which each of the n pages is a node and there is an edge for node i to node j if there is a link

* Corresponding author at: Institute of Mathematics, School of Mathematical Sciences, Fudan University, Shanghai 200433, PR China.
E-mail addresses: yiqinlin@hotmail.com (Y. Lin), 0311004@fudan.edu.cn (X. Shi), ymwei@fudan.edu.cn (Y. Wei).

0377-0427/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2008.06.003

from node i to node j. The elements of the $n \times n$ hyperlink matrix P are defined as follows:

$$p_{ij} \equiv \begin{cases} \dfrac{1}{|O_i|}, & \text{if page } i \text{ links to page } j, \\ 0, & \text{otherwise,} \end{cases}$$

where the scalar $|O_i|$ is the number of outlinks from page i. Thus, each of the nonzero rows of P sums to 1. These pages, which have no outlinks to other pages, are called dangling nodes. Let k be the number of nondangling nodes. If the rows and columns of P are permuted (i.e., the indices are reordered) so that the rows corresponding to dangling nodes are at the bottom of the hyperlink matrix P, then P is of the following form:

$$P = \begin{bmatrix} P_{11} & P_{12} \\ 0 & 0 \end{bmatrix},$$

where the $k \times k$ matrix P_{11} represents the links among the nondangling nodes, and P_{12} represents the links from nondangling to dangling nodes. The $n - k$ zero rows in P are associated with the $n - k$ dangling nodes.

To make P a transition probability matrix, it is modified as

$$\hat{P} \equiv P + dw^{\mathrm{T}},$$

where w is an n-dimensional stochastic vector (i.e., $w \geq 0$ and $\|w\| = 1$) and $d^{\mathrm{T}} = [0^{\mathrm{T}}, e^{\mathrm{T}}]$. Here, the zero vector is k-dimensional and $e^{\mathrm{T}} = [1, 1, \ldots, 1]$. In [6], the vector w is called the dangling node vector. Note that the transition probability matrix $\hat{P}$ is usually reducible, and therefore its stationary distribution is not unique. To remedy this, and thereby guarantee the existence and uniqueness of the stationary distribution vector, a further modification of $\hat{P}$ is made as follows:

$$G = \alpha\hat{P} + (1 - \alpha)ev^{\mathrm{T}},$$

where $\alpha \in [0, 1)$ and v, which is called a personalization vector, is also an n-dimensional stochastic vector. The stochastic matrix G is usually called the Google matrix. The PageRank vector π is the stationary distribution vector of G, i.e., $\pi^{\mathrm{T}} = \pi^{\mathrm{T}}G$, $\pi \geq 0$ and $\|\pi\| = 1$. Here and in the sequel, $\|\cdot\|$ denotes the 1-norm. Although the Google matrix G may not be primitive or irreducible, its eigenvalue 1 is distinct and the magnitude of all other eigenvalues is bounded by α [5,17], and therefore the PageRank vector is unique.

After partitioning w and v as $w^{\mathrm{T}} = [w_1^{\mathrm{T}}, w_2^{\mathrm{T}}]$ and $v^{\mathrm{T}} = [v_1^{\mathrm{T}}, v_2^{\mathrm{T}}]$ with w_1, v_1 being $k \times 1$ and w_2, v_2 being $(n - k) \times 1$, the Google matrix has the following block structure:

$$G = \begin{bmatrix} G_{11} & G_{12} \\ eu_1^{\mathrm{T}} & eu_2^{\mathrm{T}} \end{bmatrix},$$

where

$$G_{11} \equiv \alpha P_{11} + (1 - \alpha)ev_1^{\mathrm{T}}, \qquad G_{12} \equiv \alpha P_{12} + (1 - \alpha)ev_2^{\mathrm{T}},$$
$$u \equiv \alpha w + (1 - \alpha)v = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \alpha w_1 + (1 - \alpha)v_1 \\ \alpha w_2 + (1 - \alpha)v_2 \end{bmatrix}.$$

Note that $u_1 = v_1$ and $u_2 = v_2$ if the dangling vector equals the personalization vector.

For an extensive exposition of the PageRank problem, see the survey papers [1,13] and the book [14].

Recently, Ipsen and Selee [6] have shown that all of the dangling nodes can be lumped into a single node and the PageRank of the nondangling nodes can be computed separately from that of the dangling nodes. They have presented a simple algorithm, which applies the power method to the smaller lumped matrix and has the same convergence rate as that of the power method applied to the full matrix G, for computing the PageRank vector π.

The following important results, which we will make use of, are given in [6].

Theorem 1.1 (*[6]*). *With the above notation, let*

$$X \equiv \begin{bmatrix} I_k & 0 \\ 0 & L \end{bmatrix},$$

where $L \equiv I_{n-k} - \frac{1}{n-k}\hat{e}e^{\mathrm{T}}$ *and* $\hat{e} = e - e_1 = [0, 1, 1, \ldots, 1]^{\mathrm{T}}$. *Then*

$$XGX^{-1} = \begin{bmatrix} G^{(1)} & * \\ 0 & 0 \end{bmatrix},$$

where

$$G^{(1)} = \begin{bmatrix} G_{11} & G_{12}e \\ u_1^{\mathrm{T}} & u_2^{\mathrm{T}}e \end{bmatrix}.$$

The matrix $G^{(1)}$ *is stochastic of order* $k + 1$ *with the same nonzero eigenvalues as* G.