Weighted PageRank Algorithm

Wenpu Xing and Ali Ghorbani
Faculty of Computer Science
University of New Brunswick
Fredericton, NB, E3B 5A3, Canada

E-mail: {m0yac,ghorbani}@unb.ca

Abstract

With the rapid growth of the Web, users get easily lost in the rich hyper structure. Providing relevant information to the users to cater to their needs is the primary goal of website owners. Therefore, finding the content of the Web and retrieving the users' interests and needs from their behavior have become increasingly important. Web mining is used to categorize users and pages by analyzing the users' behavior, the content of the pages, and the order of the URLs that tend to be accessed in order. Web structure mining plays an important role in this approach. Two page ranking algorithms, HITS and PageRank, are commonly used in web structure mining. Both algorithms treat all links equally when distributing rank scores. Several algorithms have been developed to improve the performance of these methods. The Weighted PageRank algorithm (WPR), an extension to the standard PageRank algorithm, is introduced in this paper. WPR takes into account the importance of both the inlinks and the outlinks of the pages and distributes rank scores based on the popularity of the pages. The results of our simulation studies show that WPR performs better than the conventional PageRank algorithm in terms of returning larger number of relevant pages to a given query.

2. Background

With the rapid growth of the Web, providing relevant pages of the highest quality to the users based on their queries becomes increasingly difficult. The reasons are that some web pages are not self-descriptive and that some links exist purely for navigational purposes. Therefore, finding appropriate pages through a search engine that relies on web contents or makes use of hyperlink information is very difficult.

To address the problems mentioned above, several algorithms have been proposed. Among them are PageRank [10] and *Hypertext Induced Topic Selection* (HITS) [2, 9] algorithms. PageRank is a commonly used algorithm in Web Structure Mining. It measures the importance of the pages by analyzing the links [1, 8]. PageRank has been developed by Google and is named after Larry Page, Google's co-founder and president[10]. PageRank ranks pages based on the web structure.

Google first retrieves a list of relevant pages to a given query based on factors such as title tags and keywords. Then it uses PageRank to adjust the results so that more "important" pages are provided at the top of the page list [10]. The Pagerank algorithm is described in detail in the next section.

HITS ranks webpages by analyzing their inlinks and outlinks. In this algorithm, webpages pointed to by many hyperlinks are called *authorities* whereas webpages that point to many hyperlinks are called *hubs* [4, 5, 11]. Authorities and hubs are illustrated in Figure 1.

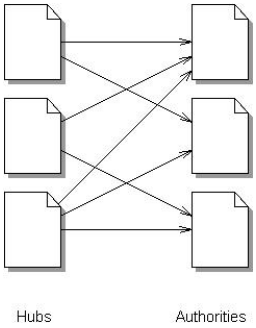

Figure 1. Hubs and authorities

Hubs and authorities are assigned respective scores. Scores are computed in a mutually reinforcing way: an authority pointed to by several highly scored hubs should be a strong authority while a hub that points to several highly scored authorities should be a popular hub [4, 5]. Let a_p and h_p represent the authority and hub scores of page p, respectively. $B(p)$ and $I(p)$ denote the set of referrer and reference pages of page p, respectively. The scores of hubs and authorities are calculated as follows [2, 4, 5]:

$$a_p = \sum_{q \in B(p)} h_q \quad (1)$$

$$h_p = \sum_{q \in I(p)} a_q \quad (2)$$

Figure 2 shows an example of the calculation of authority and hub scores.

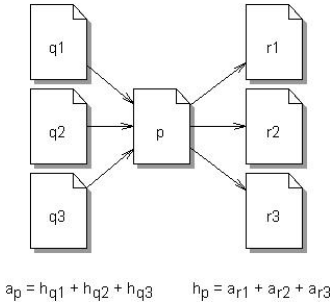

Figure 2. An example of HITS operations

HITS is a purely link-based algorithm. It is used to rank pages that are retrieved from the Web, based on their textual contents to a given query. Once these pages have been assembled, the HITS algorithm ignores textual content and focuses itself on the structure of the Web only. Some difficulties arise from this feature [2]:

- HITS frequently returns more general webpages on an otherwise narrowly focused topic because the web does not contain many resources for the topic,
- Topic drift occurs while the hub has multiple topics because all of the outlinks of a hub page get equivalent weights, and
- Some popular sites that are not highly relevant to the given query gain overhead weight values.

The CLEVER algorithm is an extension of standard HITS and provides an appropriate solution to the problems that result from standard HITS [2]. CLEVER assigns a weight to each link based on the terms of the queries and end-points of the link. It combines anchor text to set weights to the links as well. Moreover, it breaks large hub pages into smaller units so that each hub page is focused on as a single topic. Finally, in the case of a large number of pages from a single domain, it scales down the weights of pages to reduce the probabilities of overhead weights [2].

Another major shortcoming of standard HITS is that it assumes that all links pointing to a page are of equal

weight and fails to recognize that some links might be more important than others. A *Probabilistic analogue of the HITS Algorithm*(PHITS) has been developed to solve this problem[3]. PHITS provides a probabilistic interpretation of term-document relationships and identifies authoritative documents. In the experiment on a set of hyperlinked documents, PHITS demonstrates better results compared to those obtained by standard HITS. The most important feature of the PHITS algorithm is its ability to estimate the actual probabilities of authorities compared to the scalar magnitudes of authority that are provided by standard HITS[3].

3. The PageRank Algorithm

The PageRank algorithm, one of the most widely used page ranking algorithms, states that if a page has important links to it, its links to other pages also become important. Therefore, PageRank takes the backlinks into account and propagates the ranking through links: a page has a high rank if the sum of the ranks of its backlinks is high [8, 10]. Figure 3 shows an example of backlinks: page A is a backlink of page B and page C while page B and page C are backlinks of page D.

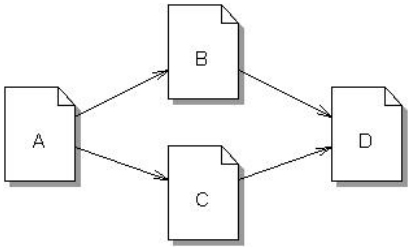

Figure 3. An example of backlinks

3.1. Simplified PageRank

A slightly simplified version of PageRank is defined as [8]:

$$PR(u) = c \sum_{v \in B(u)} \frac{PR(v)}{N_v} \tag{3}$$

where u represents a web page. $B(u)$ is the set of pages that point to u. $PR(u)$ and $PR(v)$ are rank scores of page u and v, respectively. N_v denotes the number of outgoing links of page v. c is a factor used for normalization. Figure 4 shows an example in which $c = 1.0$ to simplify the calculation.

In PageRank, the rank score of a page, p, is evenly divided among its outgoing links. The values assigned to the outgoing links of page p are in turn used to calculate the ranks of the pages to which page p is pointing. The rank scores of pages of a website could be calculated iteratively starting from any webpage. Within a website, two or more pages might connect to each other to form a loop. If these pages did not refer to but are referred to by other webpages outside the loop, they would accumulate rank but never distribute any rank. This scenario is called a *rank sink* [8].

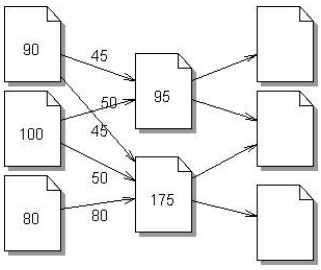

Figure 4. An example of simplified version of PageRank

3.2. PageRank

To solve the *rank sink* problem, we observed the users' activities. A phenomenon is found that not all users follow the existing links. For example, after viewing page a, some users may not decide to follow the existing links but directly go to page b, which is not directly linked to page a. For this purpose, the users just type the URL of page b into the URL text field and jump to page b directly. In this case, the rank of page b should be affected by page a even though these two pages are not directly connected. Therefore, there is no absolute *rank sink*.

Based on the consideration of the phenomenon mentioned above, the original PageRank is published [8, 10]:

$$PR(u) = (1 - d) + d \sum_{v \in B(u)} \frac{PR(v)}{N_v} \tag{4}$$

where d is a dampening factor that is usually set to 0.85. We also could think of d as the probability of users' following the links and could regard $(1 - d)$ as the pagerank distribution from non-directly linked pages.

To test the utility of the PageRank algorithm, Google applied it to the Google search engine [8]. In the experiments, the PageRank algorithm works efficiently and effectively because the rank value converges to a reasonable tolerance in the roughly logarithmic $(\log n)$ [8, 10].

The rank score of a web page is divided evenly over the pages to which it links. Even though the PageRank algorithm is used successfully in Google, one problem still ex-

ists: in the actual web, some links in a web page may be more important than are the others.

4. Weighted PageRank (WPR)

The more popular webpages are, the more linkages that other webpages tend to have to them or are linked to by them. The proposed extended PageRank algorithm–a Weighted PageRank Algorithm–assigns larger rank values to more important (popular) pages instead of dividing the rank value of a page evenly among its outlink pages. Each outlink page gets a value proportional to its popularity (its number of inlinks and outlinks). The popularity from the number of inlinks and outlinks is recorded as $W^{in}_{(v,u)}$ and $W^{out}_{(v,u)}$, respectively.

$W^{in}_{(v,u)}$ is the weight of $link(v,u)$ calculated based on the number of inlinks of page u and the number of inlinks of all reference pages of page v.

$$W^{in}_{(v,u)} = \frac{I_u}{\sum_{p \in R(v)} I_p} \tag{5}$$

where I_u and I_p represent the number of inlinks of page u and page p, respectively. $R(v)$ denotes the reference page list of page v.

$W^{out}_{(v,u)}$ is the weight of $link(v,u)$ calculated based on the number of outlinks of page u and the number of outlinks of all reference pages of page v.

$$W^{out}_{(v,u)} = \frac{O_u}{\sum_{p \in R(v)} O_p} \tag{6}$$

where O_u and O_p represent the number of outlinks of page u and page p, respectively. $R(v)$ denotes the reference page list of page v.

Figure 5 shows an example of some links of a hypothetical website.

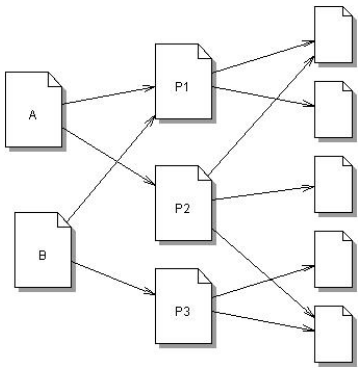

Figure 5. Links of a website

In this example, Page A has two reference pages: $p1$ and $p2$. The inlinks and outlinks of these two pages are $I_{p1} = 2$, $I_{p2} = 1$, $O_{p1} = 2$, and $O_{p2} = 3$. Therefore,

$$W^{in}_{(A,p1)} = I_{p1}/(I_{p1} + I_{p2}) = \frac{2}{3}$$

and

$$W^{out}_{(A,p1)} = O_{p1}/(O_{p1} + O_{p2}) = \frac{2}{5}$$

Considering the importance of pages, the original PageRank formula is modified as

$$PR(u) = (1-d) + d \sum_{v \in B(u)} PR(v) W^{in}_{(v,u)} W^{out}_{(v,u)} \tag{7}$$