Research on Improved Algorithm of PageRank Based on Vector Space

XIANGWEI TAN, GENGSHENG HUANG and HUIYONG JIANG

ABSTRACT

The traditional PageRank algorithm, which is based on the link analysis algorithm, considers the randomness of user access behavior, but the relevance of the query topic is poor. In order to improve the relevance of the page data search and acquisition, this paper proposes a PageRank algorithm based on the lucent vector space scoring model. The algorithm builds a vector space model based on the web content characteristics, calculates the similarity of the subject content, and combines the original PageRank algorithm, the new PR value is obtained after weighted fusion. Experiments show that the improved algorithm reduces the number of irrelevant pages, and the PR value can better reflect the relevance of the topic.

KEYWORDS

Improved Algorithm, PageRank, Vector Space.

INTRODUCTION

With the rapid development of China's Internet technology, according to the China Internet Network Information Center (CNNIC) in the 40th "China Internet Development Statistics Report" data show that China's Internet users reached 751 million scale, nearly half a total of new Internet users 1992 million people, the Internet penetration rate of 54.3%, compared with 2016 to enhance the first 1.1 percentage points [1]. With the growth of network users, users demand for effective Internet data is also growing. But in the face of massive amounts of data, people get effective information on the time and cost of increasing. How to make users more easily get their interest in the content of the page is the current information collection and information retrieval areas need to address the important issues.

PageRank is a classic algorithm used to solve the problem of page ranking in link analysis. With the in-depth study of the algorithm, it is found that the PageRank algorithm has the following drawbacks: Because the link structure relation has nothing to do with the content of the webpage, it is prone to a large number of irrelevant pages, resulting in "subject drift" problem. Based on the traditional Page Rank algorithm, this paper constructs the content feature vector space of the web page by using the vector space model technique which scoring the search results like Lucene, and calculates the similarity degree of the page and improves the sorting effect of the Page Rank algorithm.

XiangWei Tan, South China Institute of Software Engineering, GU, GuangZhou, China
GengSheng Huang, Guangdong Vocational Institute Of Public Administration, GuangZhou, China
HuiYong Jiang, South China Institute of Software Engineering, GU, GuangZhou, China
PAGERANK ALGORITHM ANALYSIS

The classic PageRank algorithm was proposed by Lawrence Page and Sergey Brin in 1998 [2]. The basic idea of the algorithm is to distinguish between the qualities of the web page by referring to each other (as a link). Assuming that the page Q is linked by a total of \( m \) Pi pages, its weight \( PR(Q) \) is Formula (1):

\[
PR(Q) = \lambda \sum_{i=1}^{m} \frac{PR(P_i)}{od_{pi}} + (1 - \lambda)
\]  

(1)

Where \( (1 - \lambda) \) is called the buffer factor, the probability of jumping from one page to another. \( od_{pi} \) The number of links to the page chain to other pages. If all the page weights are treated as a vector, the matrix of PageRank is expressed as Formula (2):

\[
PR=\lambda M \times PR + (1 - \lambda)I
\]  

(2)

Then let \( x = PR, M = PRT \), Calculated by iterative method as Formula (3):

\[
x = x \times M
\]  

(3)

In the calculation process, give each page an initial value (usually 1.0). During the iteration process, the sum of the page scores for each page is converged to the entire number of web pages, so while starting the iteration, each page \( (1 - \lambda) \) and \( (\lambda N + (1 - \lambda)) \), the iterative process is fast convergence, so it is possible to ensure that each page is calculated by iteration after iteration. The value approximates a fixed value. The formula for calculating the PageRank value of a page A is as Formula (4):

\[
PR(A) = \frac{(1 - \lambda)}{N} + \lambda \left( \frac{PR(T1)}{C(T1)} + \cdots + \frac{PR(Tn)}{C(Tn)} \right)
\]  

(4)

Where \( P \) represents the total number of pages, \( PR(A) \) represents the PageRank value of page A, \( PR(Ti) \) represents the PageRank value of the page Ti linked to A, \( C(Ti) \) represents the number of outbound links of page Ti, \( \lambda \) is the damping coefficient, The general range is: \( 0 < \lambda < 1 \) [3].

The PageRank algorithm makes good use of the web page link information and can converge quickly, so it has achieved good results in measuring the relevance of the page to the search match. But the PageRank algorithm is based on the link reflects the quality of the page method, only reflects the web page creator for the quality of the page evaluation, and does not reflect the web browser for the evaluation of the page, there are the following questions: first, since the topic query word has nothing to do, the query page may be of high authority but not related to the content, the so-called "theme drift"; and the algorithm is biased towards the old page, because the old page exists for a long time, the chance of getting the link becomes larger, the algorithm gets the corresponding PR value may be higher[4].

IMPROVED PAGERANK ALGORITHM ANALYSIS

In order to improve the PageRank algorithm, we will use the document scoring mechanism in the Lucerne framework to establish a similar spatial model, victories the web content and matching subject information into an N-dimensional space, each
vector is one-dimensional, new evaluation. The algorithm considers that the smaller the angle between the two vectors, the greater the correlation between the vectors. The cosine of the angle is the score of relevance, the smaller the angle is, the higher the cosine value, the higher the score, the higher the matching degree of information acquisition [5].

We segmented the page into a series of terms, each Term, have to give a Term weight, different Term according to their weight in the document to affect the relevance of the document score calculation. So we regard the term weight of term in all pages as a vector, as Formula (5):

\[
\text{Document} = \{\text{term}_1, \text{term}_2, \ldots, \text{term}_N\}
\]

\[
\text{Document Vector} = \{\text{weight}_1, \text{weight}_2, \ldots, \text{weight}_N\} \tag{5}
\]

Similarly, we look at the query as a simple document, but also with a vector to represent, as Formula (6):

\[
\text{Query} = \{\text{term}_1, \text{term}_2, \ldots, \text{term}_N\}
\]

\[
\text{Query Vector} = \{\text{weight}_1, \text{weight}_2, \ldots, \text{weight}_N\} \tag{6}
\]

We put all the search for the document vector and query vector into an N-dimensional space, each term is one-dimensional. As shown in Figure 1:

By calculating the cosine of angle \( \theta \) as the score of correlation, the smaller the angle, the larger the cosine, the higher the score, the greater the correlation. Cosine formula such as formula (7):

\[
\text{score}(q,d) = \cos \theta = \frac{\vec{V}_q \cdot \vec{V}_d}{|\vec{V}_q| |\vec{V}_d|} \tag{7}
\]

![Figure 1. N-dimensional space figure.](image)
the document vectors are: \( V_d = \langle w(t_1, d), w(t_2, d), ..., w(t_n, d) \rangle \), the vector space dimension is \( n \), which is the length of the union of the query and the document. When a Term does not appear in the query, \( w(t, q) \) is zero. When a Term does not appear in the document, \( w(t, d) \) is zero. \( w \) is the weight, the formula is generally \( tf \times idf \), where \( tf(t in d) \) is the frequency at which Term \( t \) appears in document \( d \), and \( idf(t) \) is the number of documents that Term \( t \) has appeared.

Since the numerator of the formula is the dot product of two vectors, then \( t_1, t_2, ..., t_n \) here only have the non-zero values of the query and the union of the document, only if the query appears or only appears in the document Term of the value of the zero; in the query, very few people will enter the same word in the query, so you can assume that \( tf(t, q) \) are 1; \( idf \) is the number of Term in the number of documents appeared, Which also includes the query phrase this small document, so \( idf(t, q) \) and \( idf(t, d) \) is the same, is the total number of documents in the index plus one, when the total number of documents in the index is large enough, Query phrase This small document can be ignored, so you can assume that \( idf(t, q) = idf(t, d) = idf(t) \); When calculating the vector length of a document, the weight of each term is no longer taken into account But all for one. The so the cosine formula becomes the formula (8):

\[
\text{score}(q, d) = \cos \theta = \frac{1}{\sqrt{\sum_{t \in q} \text{idf}(t)^2}} \times \frac{\sum_{t \in q} (tf(t, d) \times \text{idf} \sum_{t \in q} \text{idf}(t)^2)}{\sqrt{\sum_{t \in f} \text{idf}(t)^2}} ~ (8)
\]

According to the formula (4) of the traditional Rank Rank algorithm, the Page Rank algorithm which can increase the content similarity of web page content can be modified into formula (9):

\[
PR(A) = \left( \frac{1-\lambda}{N} + \lambda \left( \frac{PR(T1)}{C(T1)} + \cdots + \frac{PR(Tn)}{C(Tn)} \right) \right) \times f + \text{score}(q, d) \times (1 - f) \tag{9}
\]

Score \((q, d)\) to measure the similarity between the page and the subject; \( f \) is the fusion weight coefficient, used to adjust the original PageRank value and \( \text{score}(q, d) \) in the operation of the proportion of the value of the range 0-1.

The improved Rank algorithm has a certain ability of distinguishing results due to the combination of the similarity of web content, which can reduce the number of unrelated pages and improve the sorting effect of search results.

**EXPERIMENTAL SIMULATION**

In order to verify the feasibility and effectiveness of the improved PageRank algorithm, the my eclipse and Heretic tools are used to build the simulation experiment environment, and after a certain number of thematic web pages are scanned, the simulation experiment is compared with the algorithm. The experimental procedure is as follows: firstly, the traditional PageRank algorithm and the improved PageRank algorithm are used in the Java language. The parameter \( f \) of the formula (9) is set to 0.3, and then the two algorithms are applied to the Heretic by using the extended interface provided by the Web crawler framework Heretic. Take the
processor chain, crawling a specific topic of pages, and finally by crawling the subject of the relevance of the comparison to analyze the performance of the two algorithms.

The performance evaluation of the experimental evaluation is the precision and recall [6]. Described as formula (10):

\[
\text{Precision} = \frac{\text{total number of related links}}{\text{total number of links extracted}} \\
\text{Recall} = \frac{\text{total number of related links}}{\text{total number of links in the page}} 
\]  

(10)

To verify the results, the crawl theme is set to “Military training”, the crawling initial URL is http://www.sise.edu.cn/, the experimental results are shown in Figure 2 and Figure 3.

Figure 2. Precision data comparison.

Figure 3. Recall data comparison.
From the results, it is shown that the improvement rate of the improved PageRank is not very different from that of the conventional tunneling system, but the accuracy of the study is better.

CONCLUDING REMARK

Based on the analysis of the traditional PageRank algorithm, an improved PageRank algorithm based on vector space is proposed. Fusion of Lucerne vector space retrieval evaluation model, the algorithm takes into account the link structure of the web page and the relevance of the subject content, effectively suppressing the subject drift. The author will work on how to calculate the content similarity according to the semantic text of the page rather than the document weight, so as to further improve the relevance of the query subject content.

REFERENCES