Weights Allocation Optimization of Search Engine Links Sorted Pagerank Algorithm

Xiaoling Luo, Heru Xue

Computer and Information Engineering College,
Inner Mongolia Agricultural University, Hohhot 010018,
Inner Mongolia, China

Abstract
The application of using PageRank algorithm in the links sorted search engine usually has some problems such as low sorting accuracy and bad customer experience. This paper proposes a search engine link scheduling model based on distribution of weights based on ant colony optimization of PageRank algorithm. The colony is initially distributed in a solution space using a certain method. Then, the current ant colony of information distribution is decided by the position solution space of the ant colony. The number of ants in the each interval is decided by the current ant colony scattered distribution of the total amount of information and information in a cycle of legacy and volatile situation. The movement of the ant colony is decided by the ant colony distribution of each interval and the difference between the current ant colony distribution. Finally, the improved ant colony algorithm is used for optimization of PageRank weights allocation algorithm and applied for the sorting of the search engine links.
Key words: PAGERANK ALGORITHM, SEARCH ENGINE, LINK TO SORT, WEIGHT DISTRIBUTION OPTIMIZATION, SEGMENTED ANT COLONY

1. Introduction

Internet network information center survey reports that 82.5% of Internet users often use search engine, 83.4% of users learned new website through search engines [1]. Therfore, the search engine played an important role in our daily network life. A good search engine can find the real knowledge from big data as well as improve the value of the information through screening, processing and purification. However, due to the low performance of current search engine relevance sorting algorithm, the search engine's navigation function not fully performed. For example, users often need to manually select from a large number of returned results related web page [2]. The web pages show in the search results is the key information and knowledge could deliver to users. The relevance of search results and the query demand is the important indicator of search engine performance. Therefore, the research of search engine relevance sort is very important.

There are two basic algorithms for the mining of the network structure and link relations: the first is the PageRank algorithm, which proposed by Sergey Brin and Lawrence Page from Stanford University [3]. This network links analysis algorithm is referenced by the ideas of traditional citation analysis. The algorithm is through the analysis of the link structure of the network to obtain the authority of the web page.

This method was successfully used in the commercial search engine Google. However, this algorithm only use the web link structure to assess the authority of the web, which has a lot of shortcomings, such as lay particular stress on the old website, cannot distinguish the similarity of content, poor convergence due to the link sparse matrix and broken links come from distributed computing. The second algorithm is proposed by Cornell University's Kleinberg j, which introducing hypertext theme for the first time into search and HITS algorithm [4]. The basic idea of the algorithm is mining of information between the chains of web pages [5]. After the development of these two algorithms, several other web page link analysis based algorithms was developed such as ARC, SALSA and PHITS. They all have achieved very good results in practical applications. Moreover, the web information which people need also could achieve by mining of Hub value of the web through the combination of web structure analysis and random sampling method. Successful use in Google PageRank algorithm not only confirmed the feasibility and the validity of the algorithm, but also has sparked the enthusiasm for the extensive research of the algorithm. Many scholars put forward many improved algorithm about PageRank algorithm. Taher h. Haveliwala from Stanford University proposed a topic sensitive PageRank algorithm[6]. Matthew Riehardson and Pedro Dominggos from The University of Washington proposed a PageRank algorithm based on the combining links and content information [7]. Yenyu Chen and Qingqing Gant from The Brooklyn Polytechnic University proposed a PageRank accelerated algorithm based on I/O technology [8]. Sepandar Kamvar proposed an adaptive PageRank algorithm [9]. Juping Song proposed a PageRank algorithm by the combination of the web page HTML [10].

Based on the defects of Rank algorithm in the search engine links the application. This paper proposes a distribution of weights based on ant colony optimization search engine link scheduling model of PageRank algorithm. The experimental simulation is performed for confirming the effectiveness of the improved algorithm

2. Defect analysis of PageRank algorithm

The basic idea of PageRank: The importance of the web page is decided by two aspects. One is number of other web pages which linked. Another one is the quality of the link pages. If the web page is referenced by many other web pages or cited by other high quality web pages, we can call the original web page has high quality. PageRank is through the web link structure to calculate the importance of the web page. Each web page link to its web vote value can be regarded as a directed edge. Each page of the PageRank value is the sum of the vote. Figure 1 is the scheme of an example.

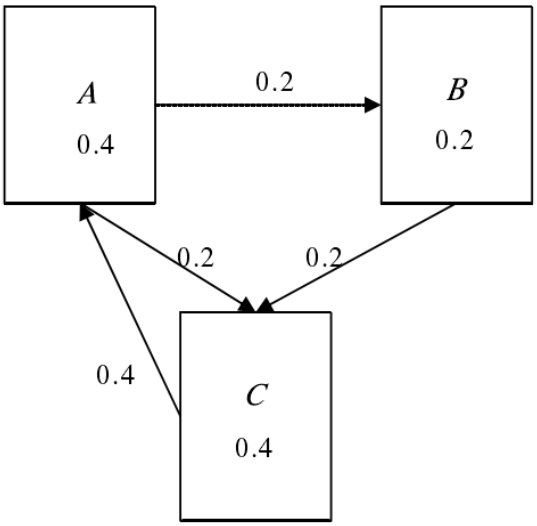

Figure 1. Web links example

From the above example, web B and C are equally received score from web A. Then, web B gives all score to web C. Finally, web C gives all score back to web A. The total score received by web pages is 1. No matter how cycle calculation, the distribution is no longer change, which often referred as the nature of "invariant distribution". That is how a PageRank eventually is convergence. The problem of this is that part of the web page, especially new ones, there is no link or rare link in and link out, which called link sinkage. In order to solve this problem, the computational formula of PageRank can be added a damping factor d and display as:

$$PR_n(A) = \frac{(1-d)}{m} + d \times \left(\sum_{i=1}^{m} \frac{PR_{n-1}(T_i)}{C(T_i)} \right) \quad (1)$$

Where $PR_n(A)$ is the PageRank value of web page A. $PR_{n-1}(T_i)$ is the last iteration PageRank value of web page T_i. The web page T_i is linked to web page A. $C(T_i)$ is the total number of links of web page T_i which link out. d is damping factor, which normally in the range of 0 to 1. It was set as 0.85 in this case.

PageRank calculation involves the mathematical principle of random process, which can be calculated by the method of iteration. In actual operation, people usually using the power method to calculate PageRank. Power method is a numerical iterative method, which mainly used for calculating the eigenvalues and corresponding

eigenvectors of real matrix A. It is especially suitable for large sparse matrix, but sometimes slower convergence speed. Its convergence speed is decided by $r = |\lambda_2 / \lambda_1|$, where $\lambda_1, \lambda_2, L, \lambda_n$ is the eigenvalue of A. Also $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq L |\lambda_n|$.

Set a real matrix $A = (a_{ij})_{n \times n}$ has a complete eigenvalue vector group. The eigenvalue is $\lambda_1, \lambda_2, L, \lambda_n$, which corresponding eigenvalue vector x_1, x_2, L, x_n. The main eigenvalue of A is real root and satisfy $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq L |\lambda_n|$.

The basic idea of the power method is set to a nonzero vector v_0 and construct a vector sequence from matrix A.

$$\begin{cases} v_1 = Av_0 \\ v_2 = Av_1 = A^2 v_0 \\ v_{k+1} = Av_k = A^{k+1} v_0 \end{cases} \quad (2)$$

The vector sequence is an iteration vector. Set v_0 as:

$$v_0 = a_1 x_1 + a_2 x_2 + L + a_n x_n \quad (3)$$

Then,

$$\begin{aligned} v_k &= a_1 A^k x_1 + a_2 A^k x_2 + L + a_n A^k x_n \\ &= \lambda_1^k \left[a_1 x_1 + \sum_{i=2}^{n} a_i (\lambda_i / \lambda_1)^k x_i \right] \\ &= \lambda_1^k (a_1 x_1 + \varepsilon_k) \end{aligned} \quad (4)$$

Among them $\varepsilon_k = \sum_{i=2}^{n} a_i (\lambda_i / \lambda_1)^k x_i$.

The hypothesis indicates $|\lambda_i / \lambda_1| < 1$, therefore, when $k \to \infty$, $\varepsilon_k \to 0$. After that,

$$\lim_{k \to \infty} \frac{v_k}{\lambda_1^k} = a_1 x_1 \quad (5)$$

It indicates sequence $\frac{v_k}{\lambda_1^k}$ more and more convergence in the eigenvector of λ_1 corresponding to A. When k value is big enough, $v_k \approx a_1 \lambda_1^k x_1$. It indicates iteration vector v_k is an approximate vector of the eigenvector of λ_1

Set $(v_k)_i$ as the i component of v_k,

$$\frac{(v_{k+1})_i}{(v_k)_i} = \lambda_1 \left\{ \frac{a_1 (x_1)_i + (\varepsilon_{k+1})_i}{a_1 (x_1)_i + (\varepsilon_k)_i} \right\} \quad (6)$$

Therefore,

$$\lim_{k \to \infty} \frac{(v_{k+1})_i}{(v_k)_i} = \lambda_1 \quad (7)$$

When k is large, the ratio of two adjacent iteration vector components and the main characteristic value is more and close.

From formula (6) can be deduced that the convergence speed of $\frac{(v_{k+1})_i}{(v_k)_i} \to \lambda_1$ is decided by $r = |\lambda_2 / \lambda_1|$. The convergence speed increases when increasing r. In contrast, the convergence speed decreases when $r = |\lambda_2 / \lambda_1| \approx 1$.

The formula (1) can be used for solving $\lim_{n \to \infty} A^n x$. Where A is $A = d \times P + (1-d) \times ee^T / m$, d is damping coefficient, P is probability transfer matrix, e^T is the all 1 lines in n dimension, m is number of web page, x is PageRank value of each initial web pages. Because the PageRank value of each initial page is 1, the vector product of ee^T / m and all web page PageRank value at every iteration could keep a vector quantity of $1-d$ at n dimension.

PageRank is a static algorithm, which has not been related to query. The PageRank values of all web pages are calculated offline, which effectively reduce the online computation of the query, greatly shorten the response time. But in reality, the value of a web page referenced by authoritative web page or advertising, and even rubbish web page is different. PageRank algorithm distributes current web page weight evenly to all its links, not considering the authority of the web page itself. Therefore, this method of average weight distribution page, greatly reduces the search effectiveness of the algorithm.

Weight distribution of ant colony optimization

3.1. Sectional optimization of ant colony algorithm

After the initial distribution of ant colony optimization process, the sectional ant colony algorithm should also include the information distribution function, analysis of the amount of information distribution status and the direction of ant colony decision cycle. We use an one-dimensional function $y = f(x)$ of maximum (minimum) optimization as an example for one dimensional continuous space ant colony algorithm study. Based on this function, the multidimensional space function optimization can be extended. The definition of the searching optimization as follows:

$$D_{RL} = \frac{End - Start}{N} \quad (8)$$

By the above description, The length of every single ant with mobile interval is $D_{MRL} = D_{RL}$ and the Initial coordinate distribution of ant colony is:

$$x_i = Start + \left(\frac{i}{N} - \frac{1}{2} \right) D_{RL} \quad (9)$$

The left border of interval *i* is:

$$x_i = Start + (i-1) D_{RL} \quad (10)$$

The right border is:

$$x_{iR} = Start + iD_{RL} \quad (11)$$

When single ant move Δx, the actual number of ants of two adjacent interval single ant moving N_{iR} can be deduced by the coincidence degree change Δn with its adjacent subinterval mobile interval:

$$\Delta n = \frac{\Delta x}{D_{MRL}} = \frac{\Delta x}{D_{RL}} \quad (12)$$

When move to the right, the actual ant number increase Δn in the right side of the intervals. Meantime, the actual ant number decreases Δn in the left side of the intervals. Vice versa.

The information distribution of the current ant colony is decided by the pros and cons of the position solution space ant colony.

According to the function value $f(x_i)$ of current ant position x_i and different categories of optimization problem, the corresponding peak value and the information distribution function can be deduced. For example, the minimum value range of the specific optimization function can define the corresponding peak information distribution function

$$M_i = C - f(x_i) \quad (13)$$

Where C is the constant value of function $f(x_i)$, which satisfy $C > f(x_i)$. For smaller function value, its peak value of the information distribution function is bigger. For maximum function optimization, when $f(x_i) > 0$, it can be express as:

$$M_i = C_i f(x_i) \quad (14)$$

Where C_1 is the constant value of specific problem, when $f(x_i) < 0$, it can define:

$$M_i = \frac{C_3}{C_2 - f(x_i)} \quad (15)$$

Where C_1 and C_2 are the constants value of specific problem. For one dimensional space function optimization problem, a single ant corresponding information distribution function can be expressed as :

$$T_i(x) = \frac{M_i e^{-k_i(x-x_i)}}{\left[1 + e^{-k_i(x-x_i)}\right]^2} \quad (16)$$

The information distribution function displays a straw hat shape. The peak value is M_i, center offset value is x_i, waveform compression coefficient is k_i.

The each interval number of ants is decided by the current ant colony scattered distribution of the total amount of information and information in a cycle of legacy.

First of all, collect the current ant colony scattered distribution of the total amount of information function in the integral value of each interval.

$$IN_i = \int_{iL}^{R} \sum_{i=1}^{N} T_i(x)dx \quad (17)$$

The total amount of information of each subinterval can be calculate as follow:

$$I_i = IN_i + \eta I_{iLast} - E_v \quad (18)$$

It includes the current ant colony information in subinterval (IN_i) plus the left part of the total amount of information ηI_{iLast} (η is coefficient of amount of information retained), then subtract amount of volatile constants E_v. Finally, calculate the total amount of information in the sum of the whole problem interval.

$$I_\Sigma = \sum_{i=1}^{N} I_i \quad (19)$$

According to the ratio of actual total amount of each subinterval I_i and I_Σ, the number of ants of each subinterval can be calculated.

According to the distribution of each interval and the difference between the current ant colony distribution, determine the moving direction of the ant colony.

First of all, the movement of ant i is decided by the number of ants in the left interval N_{iML} and the number of ants in the right interval N_{iMR}. For perticular:

$$N_{iML} = \sum_{j=1}^{i-1} N_{jM}, N_{iMR} = \sum_{j=i+1}^{N} N_{jM} \quad (20)$$

According to the know number of ants N_{iR} to calculate the actual left ants number of N_{iRL} and right ants number N_{iRR}. For perticular:

$$N_{iRL} = \sum_{j=1}^{i-1} N_{jR}, N_{iRR} = \sum_{j=i+1}^{N} N_{jR} \quad (21)$$

Then, the movement direction of ant and the coordinate change are according to the difference between actual ants number at both sides and the ants number should be at both sides.

3.2. Optimization of ant colony sorting PageRank algorithm

After keyword searching, users make their own choice according to subjective judgment. Because everyone's judgment is based on the web link and the connection degree of the information. Most users will click the really high quality web site, which leads to the high information content. We can use a certain function transform to influence the web page ranking, which provides a relative information to users.

Over a period of time, $P_i(T, key)$ shows whether the web T is clicked by search engine use

the key word function i . It can display as :

$$P_i(T,key)=\begin{cases}0, false\\ 1, true\end{cases} \quad (22)$$

$\sum_{i=1}^{n} C_i(T,key,s)\times P_i(T,key)$ is the number of web T clicking times after weighting trhough function $C_i(T,key,s)$ using key word. We can modify the weight of a web page through this value.

Amending the weight key of structural weight correction function $M(T,key)$ through web T can be express as:

$$M(T,key)=\alpha\left[\lg\sum_{i=1}^{n} C_i(T,key,s)\times P_i(T,key)\right]^2 \quad (23)$$

Where α is damping coefficient, which is used for restriction of the number of clicks to impact the weights of the web page. Because the influence of PageRank is very big even the value is 1, we set α as 0.02.

Therefore, we can obtain a final value formula between web T and key word key.

$$PR(T)=d+(1-d)\sum_{i=1}^{n}\frac{PR(T_i)}{C(T_i)}+M(T,key) \quad (24)$$

This method use function $C_i(T,key,s)$ to calculate the different weights of the serach results and bring the low-ranking web page to the front by multiple clicking, which could let user to observe the result.

4. Algorithm performance simulation

In order to verify the performance of the improved algorithm proposed in this paper, the selected keyword "computer" was used for simulation. We use the traditional PageRank algorithm and the improved PageRank algorithm for web ranking. Figure 2 shows the results of both search query.

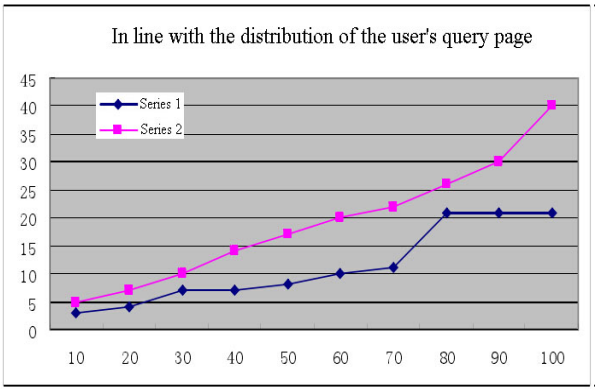

Figure 2. Query results compare two algorithms

In order to further verify the feasiblity, accuracy and recall rate of the algorithm , we use several keywords in the query, calculating accuracy and recall rate of each algorithm.The experimental results are shown in table 1. Figure 3 shows the comparison of accuracy and recall rate of two different kinds of algorithms.

Table 1. Arithmetic precision and recall rate comparison

No.	Algorithm	Accuracy/%	Recall/%
1	PageRank	83.25	72.42
2	Improved PageRank	91.46	93.03

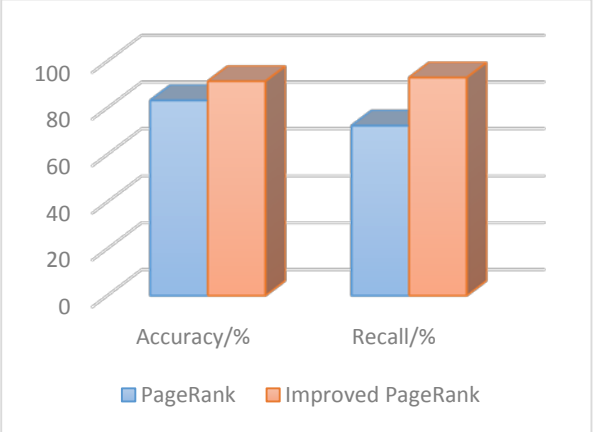

Figure 3. Arithmetic precision and recall rate comparison

From the comparison of the traditional PageRank algorithm and the improved PageRank algorithm in accuracy and recall rate, it can be seen that the improved PageRank algorithm is superior to the traditional PageRank algorithm. Although traditional PageRank algorithm is fully considering the link structure of the web page score, but has no judgment for web similarity.

5. Conclusion

With the improvement of computer system performance and progress of network technology, the world wide web is the world's largest repository of information. How to provide such a huge information resource efficient navigation services, to help users quickly find needed information in the vast amounts of data is an urgent problem in the search engine. Often users only care about search result, but the current search engine returned results and the relevance of the user requirements is not ideal. Based on the existing defects of PageRank algorithm in the application, this paper proposes a distribution of weights based on ant colony optimization search engine link scheduling model of PageRank algorithm. The experimental simulation results show that the proposed algorithm is superior to the traditional PageRank algorithm in terms of accuracy and recall rate.

Acknowledgements

The scientific research project of Inner Mongolia Autonomous Region colleges and

Universities Evaluation model Chinese search engine (NJZY13074).

References

1. Shiguang J U, Xia L V. (2014)Temporal link-analyze based on Web page ranking algorithm. *Application Research of Computers*, 35(7), p.p.2438-2441.
2. Fuyong Y, Yuanyuan Z. (2013) Correlation arrangement method research and improvement based on link analysis. *Computer Engineering and Design*, 28(7), p.p.1630-1631.
3. MingJun X, (2014) SHITS:a WebPage Ranking Method Based on Hyperlink and Content. *Mini-micro Systems*, 27(12), p.p. 2177-2182.
4. Hongwei W. (2015) Countering page ranking spam for search engine based on text content and link structure analysis. *Systems Engineering-Theory & Practice*, 35(2), p.p.445-457.
5. Zhongmei S. (2014) Study on extraction and ranking of temporal semantics and system implementation. *Computer Engineering & Science*, 36(8), p.p.1609-1614.
6. Guilin L, Yuqi Y. (2014) Personalized Representation and Rank Algorithm for Enterprise Search Engines. *Journal of Computer Research and Development*. 51(1), p.p.206-214.
7. Naizhou Z. (2014) A temporal-aware model for search engine. *Journal of Shandong University*, 48(11), p.p.80-86.
8. Cunhe L, Keqiang Lu (2013) Research of page-ranking modifying method on search engine Nutch. *Computer Engineering and Design*, No. 6, p.p.1343-1346.
9. Xianying H, Jinpeng Z. (2014) Research on page ranking algorithm based on K-means clustering algorithm and information entropy. *Computer Engineering and Design*, 34(5), p.p.1695-1699.
10. Jianxia C, Ri H. (2014) Optimization and Implementation of Lucene Ranking Pages Algorithm Based on PageRank. *Computer Engineering & Science*, 34(10), p.p.123-127.
11. Li F, Tasnuva T, Wen J H, Aimin Y (2014) Chemical preparation and applications of silver dendrites. *Chemical Papers*, 68(10), p.p.1283-1297.