Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

# STRank: A SiteRank Algorithm using Semantic Relevance and Time Frequency

Hongzhi Guo, Qingcai Chen, Xiaolong Wang, Zhiyong Wang, Yonghui Wu
Dept. of Computer Science and Technology
Harbin Institute of Technology Shenzhen Graduate School
Shenzhen, P.R. China
hongzhi.guo@gmail.com, qingcai.chen@gmail.com, wangxl@insun.hit.edu.cn

***Abstract*—Most of the researches on web information processing are concentrated on the web pages and the hyperlinks among them. One of the important facts that a web page is just one building block of the whole website had been ignored. But the situation is gradually changed in recent years for the needs of website reputation calculation, the high level website structure mining etc. It causes the website ranking become one of the hot research topics and various site ranking algorithms, such as SiteRank, AggregateRank etc., had been proposed. But most of existing website ranking algorithm just take use of website link graphs and the content of websites are usually not put into consideration. It is obviously not enough for a reliable ranking of websites. To address this issue, this paper introduces two content based features, i.e., semantic relevance and time frequency and proposes a new STRank algorithm based on these two features. We firstly conduct a series of experiments to verify the feasibility of these two factors in site ranking task. Then the semantic relevance is applied in the calculation of transition probability, and the updating frequency of sites is combined into the ranking task. Since traditional *Kendall's τ distance* and *Spearman's Footrule distance* is not appropriate for the evaluation of site ranking, we make some modifications accordingly to evaluate website ranking algorithms. Finally, our experiments show that the STRank algorithm outperforms existing approaches on both effectiveness and efficiency.**

***Keywords*—STRank, semantic relevance, time frequency, updating frequency, site ranking**

## I. INTRODUCTION

With the explosive increase of Web information, people have been used to finding information with the help of search engines. Among the processing, ranking has been a key technical link in design of S.E., which attracts widespread attention. Different strategies are implemented on this topic. Some of them is based on classical information retrieval technologies, such as Vector Space Model (VSM) [1], extended Boolean Model [1], probability model [2], BM25 [3] etc.; Others analyze Web link structures, for example, the well-known PageRank algorithm, which was proposed by Google in 1998 [4], and the hub and authority method by Kleinberg in 1999 [5]. However, whether Web pages are ranked based on classical information retrieval technologies or the latter ranking algorithms such as PageRank, HITS, website, as part of the Web, has been overlooked.

Traditionally, the Web is composed of two elements: Web pages and hyperlinks between, corresponding to contents and the structure of the Web respectively [6]. And all the ranking algorithms above were designed from these two aspects. In recent years more and more researchers have realized the importance of websites. Generally Web pages from the same website always have more similarity in their contents, hyperlinks, etc., and they can provide lots of semantic information, which is very useful in Web search and Web data mining.

Website ranking is very useful in search engines. Generally speaking Web pages from important websites always have higher weights in results ranking. Furthermore, important websites should be crawled first and have higher updating priority when designing spiders[7]; website ranking can also be used in website gathering and navigation.

However existing technologies of site ranking are limited to one setting of ranking, namely ranking based on the link analysis. These methods took users' browsing behaviors as a random walk model, and then computed the transition probability matrix. In the computing, they suppose that it was of equal probability to click all the hyperlinks in one page. But in fact the choosing for next page is of inequable probability; people tend to select the pages they are interested in. In other words people tend to click the hyperlinks which have higher semantic relevance between the anchor texts and the page contents. Semantic relevance should be considered in the computing of site ranking. i.e. "US stocks slip after economic data, GE rating cut" and "Madonna wins in custody row with Guy" are two anchor texts from SINA BUSINESS page. For most users they would click the first hyperlink, which is more relevant to the page content. Besides, for site ranking the updating frequency of websites is also important. Obviously if a website rarely updates, even though it has lots of out-links, the site should not be given a high ranking. In this paper, we implement semantic relevance in the calculation of site ranks, and combine time frequency into the final ranks. The final experiments verified our method's feasibility.

To sum up, our main contributions in this paper are:



978-1-4244-2794-9/09/$25.00 ©2009 IEEE

First, based on state of art site ranking algorithms, using anchor texts semantic relevance is implemented in computing rank values;

Second, time labels in Web pages are considered in computing the updating frequency of websites. And then the updating frequency of websites is further imported in computing of Site Ranking.

Finally, evaluation criteria for site ranking are discussed, traditional *Kendall's τ distance* and *Spearman's Footrule distance* are not appropriate enough, some modifications are made to the evaluation in Web Search.

The rest of the paper is organized as follows. It starts with a brief review of related works in Section 2. Then in Section 3, semantic relevance and time frequency are discussed and imported into website ranking, the STRank algorithm is proposed. The experimental results and discussions are provided in Section 4 in details. Finally, Section 5 concludes this paper and gives directions for future works.

## II. RELATED WORKS

Lots of previous work concentrated on the granularity of Web pages. The Web abstracted as a Graph, called DocGraph [8] (Web pages from different sites as vertices and hyperlinks between as edges). By analyzing the link structure in the DocGraph, in 1998 Brin and Page [4] proposed the famous PageRank algorithm to rank Web pages. In PageRank, users' random Web surfers are abstracted as a random walk model over the DocGraph. After people realized the importance of websites ranking, they began to study the graph on the granularity of websites by treating the Web graph as HostGraph [8] [9] (vertices are websites, and edges are the hyperlinks from the pages in one website to the other if the hyperlinks exist).

Based on the PageRank algorithm, several site ranking algorithms have been raised. In 2003, due to the fact that computing PageRanks for the whole Web graph is both time-consuming and costly, Jie Wu and Karl Aberer proposed the SiteRank algorithm [10][11], which performed the task of global ranking computation in a decentralized fashion and had been successfully used in website ranking and Web data mining. Since the number of websites is much less than the number of Webpages, the cost is largely reduced. However the SiteRank computing just described the browsing behaviors of the Web surfers, leaving some transition information of the random surfer behind. Actually the Web surfers over websites and Webpages are different; Guang Feng et al. [9] revealed this problem and proposed the AggregateRank algorithm. They proved that the probability of visiting a website equaled to the sum of PageRanks of the Web pages in that website, and gave an approximate computing using the theory of stochastic complement. Compared to SiteRank, AggregateRank is more comprehensive theoretically, taking into account the impaction on the probability of visiting websites, which comes from hyperlinks inside a website and that between two websites. The approximate calculation method not only maintained the approximation with PageRankSum (the sum of PageRanks of all the pages in that website), but also contributed to the calculation of website ranking.

Although the AggregateRank algorithm performs better than the SiteRank algorithm, it still has some deficiencies: First the algorithm only considers the impaction from the structure facet, like hyperlinks. The distribution of the link weight is tedious; the characteristics of websites and some semantic information are neglected, like website hierarchy structure [12], the relevance between Anchor Texts and Web pages [13]. Second, the AggregateRank algorithm just evaluates the ranking of the static websites. But websites on the Web might be updated everyday; the algorithm doesn't reflect the dynamic characteristics of websites [14].

## III. THE STRANK ALGORITHM

### A. HostGraph

First, the definition of HostGraph (the link graph of websites) is given. A set *G (V, E)* denotes a HostGraph. *V* is a set of vertices, where every vertex $v_i \in V$ denotes a website; E is a set of edges, where each edge $e_i \in E$ denotes a site link if and only if there are links from the pages in one site to another. Fig. 1 shows a HostGraph [8] [9] of two websites.

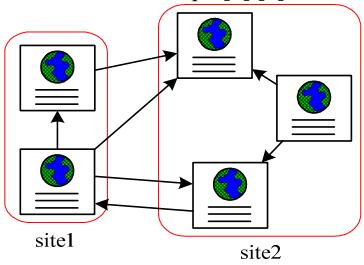


Figure 1. Example of a HostGraph of two websites

### B. Semantic Relevance

In AggregateRank, the random surfer model is described as a Markov chain, and the probability of visiting different websites is formalized as a transition probability matrix. The random surfer assumes that it is of equal probability to click all the hyperlinks inside one Web page, i.e. if the page contains *n* links, then the probability of clicking each hyperlink is *1/n*. But in fact the probability of clicking each hyperlink is different. It is related to the relevance between the anchor text and the browsing Web page.

The anchor text or link label is the visible, clickable text in a hyperlink. Anchor text usually gives the user relevant descriptive or contextual information about the content of the link's destination [15]. When users browse Web pages, using anchor texts they could select the next Web pages which they are interested in, e.g. "finance", "stock", "4000000000000 investments into A-share might be doping" and etc. The first two anchor texts are navigation links in Sohu homepage[1], which point to Sohu's secondary homepages, "Sohu Finance"

[1] http://www.sohu.com/

[2] and "Sohu Stock"[3]; and the last anchor text points to a financial news page, it is also a brief overview of the body of that page.

Anchor text has been widely used in Topic Prediction for target Web pages, and the composition of anchor texts is very similar to query words in search engines, they are both of phrase structure and a brief description of target Web pages, some researchers use anchor texts to index Web pages to overcome the drawbacks in the traditional relevance computing of Web page contents.

Based on the AggregateRank algorithm, a site ranking algorithm is proposed using the similarity between anchor texts and Web pages. The core thought of the algorithm is that: the probability of clicking each hyperlink inside one Web page is not equal, and it is related to two factors, namely the number of hyperlinks inside that page and the correlation between anchor texts and the body of the Web page. The probability of a Web page being clicked is inversely proportional with the number of hyperlinks, and in direct proportion with the correlation.

The revised formula for calculating the page transition probability is shown below in (1) [9].

$$p'_{ij} = \begin{cases} \alpha \times s(i,j) + (1-\alpha) \times 1/n; & \text{L(i,j)} \neq 0 \\ (1-\alpha) \times 1/n; & \text{L(i,j)=0} \end{cases} \quad (1)$$

Where, $s(i,j)$ denotes the jumping probability, the corresponding calculation is shown in (2) (if the hyperlink from page *i* to page *j* exists); *n* is the number of all the Web pages; $L(i,j)$ is the number of hyperlinks between page *i* and page *j*; $\alpha$ is the damping factor here, $0 < \alpha < 1$, usually set to *0.85*.

$$s(i,j) = \beta \times 1/d_i + (1-\beta) \times Sim(at_j, con_i) \quad (2)$$

Where $d_i$ denotes the out-degree of page *i*; $Sim(at_j, con_i)$ is the similarity between the anchor text which points to page *j* and the body of page *i*, which is computed using vector space model, as shown in (3) [1]; $\beta$ is another damping factor, $0 < \beta < 1$, its value will be discussed in Section 5.

$$Sim(D_1, D_2) = \cos\theta = \frac{\sum_{k=1}^{N} W_{1k} W_{2k}}{\sqrt{(\sum_{k=1}^{n} W_{1k}^2)(\sum_{k=1}^{n} W_{2k}^2)}} \quad (3)$$

Where $D_1$, $D_2$ separately denote two documents; A document is composed of terms, formalized as $D(t_1, t_2, \ldots, t_k, \ldots, t_n)$, $t_k$ is a term, and $1 \le k \le n$. Each term $t_k$ is given a weight $W_k$.

For non-topical Web pages, there are few contents existed, then all the anchor texts in that page are used as page contents. This replacement coincides with the situation when users choose to click the hyperlinks in non-topical pages, which only have relationship with the similarity between the anchor text and all anchor texts in that page.

Through the formulae above, the premise that the sum of transition probability equals to 1 is satisfied. Link analysis and the similarity between anchor texts and Web pages are all used in ranking computing. The new algorithm uses the same mathematical model as AggregateRank(the stochastic complement is used and the convergence of the method can be guaranteed). However the computing of transition probability is more comprehensive.

### C. *Time Frequency*

If a website updates frequently or contains a wealth of Web pages, or have a clear topic and link hierarchy structure, then it should be treated as a valuable and authoritative website. It usually has a higher activity and might be visited with higher probability. Furthermore these sites usually have more opportunity to be crawled and indexed by search engines.

In order to quantitatively describe website updating, the concept of website updating frequency is proposed, also called website activity. Website updating mainly focused on two aspects: the mount of updated pages during a period of time, and the quality of updated pages in a website.

The quality of website updating refers to that, when the updated pages are thematic, there would be higher contributions to the updating quality, navigation pages vice versa. This is because, navigation pages and non-topic pages are very few, and they usually give little contribution on the quality of website updating. Due to this, the cheating for higher site updating frequency, which comes from adding or accumulating lots of non-topic pages, can be prevented.

According to the theories above, the formula for calculating site updating frequency is given, as shown in (4).

$$Freq(s) = \delta \times \frac{N_a}{D} + (1-\delta) \times \frac{N_{na}}{D} \quad (4)$$

Where $N_a$ denotes the count of updated thematic pages in a website, and $N_{na}$ denotes the count of updated non-topic pages. *D* is the updating time interval for calculating updated pages. $\delta$ is a damping factor, $0 < \delta < 1$, usually set to *0.85*.

In order to reveal the relevance between the updating frequency of websites and website ranking, we did some experiments first.

Using the dataset from Haitianyuan[4], the values of updating frequency of each website were computed based on the formula (4) above, and were ranked then. To have quantitative comparison between the ranking and the benchmark, we adopted the *Kendall's τ distance* and *Spearman's Footrule* as the evaluation ctriteria. The results are shown in Fig. 2 and Fig. 3. In the figures, the x-axis denotes the number of chosen websites. More details on experiments will be discussed in section 5.

[2] http://business.sohu.com/
[3] http://stock.sohu.com/

[4] http://www.haitianyuan.com/frank/siterank.php

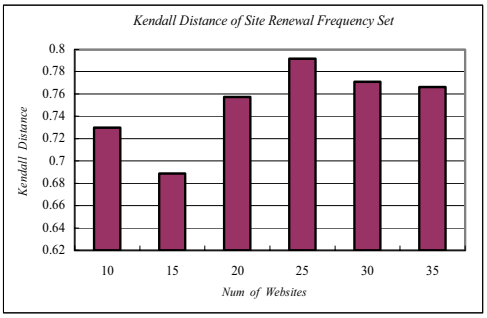


Figure 2. *Kendall Distance* of Site Renewal Frequency Set.

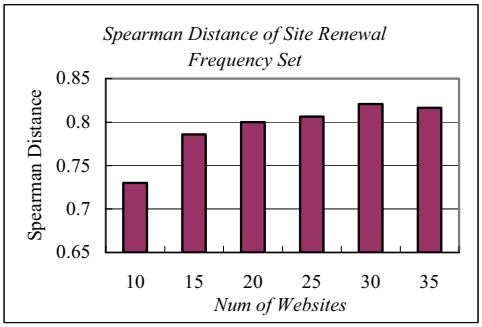


Figure 3. *Spearman Distance* of Site Renewal Frequency Set.

From Fig. 2 and Fig. 3, it could be found that the *Kendall's τ distance* between the rankings, which come from the site updating frequency, and the baseline is above *0.69*, and the corresponding *Spearman's distance* is above *0.73* under different number of websites; As shown in Table I, the average *Kendall's τ distance* and *Spearman's distance* are separately above *0.75* and *0.79*. Generally if the distance is above *0.65*, the results can be seen to have a high approximation with benchmark. According to these experiments, it can be concluded that the site updating frequency as an improvement for site ranking algorithms is feasible.

TABLE I. AVERAGE DISTANCE EVALUATION OF SITE RENEWAL FREQUENCY SET

| Baseline | *Kendall's τ distance* | *Spearman's distance* |
|---|---|---|
| Alexa | 0.75098933 | 0.7932005 |

### D. The Proposed STRank Algorithm

According to the discussions above, the semantic relevance between anchor texts and Web pages are used in computing the page transition probability, and when the rank values are obtained, we combine site updating frequency with them. Among all this processing the determination of page type has been implemented. The detailed algorithm of STRank is discussed below.

Step 1. Calculate the page transition probability $p_{ij}^{'}$ from

$$p_{ij}^{'} = \begin{cases} \alpha \times s(i,j) + (1-\alpha) \times 1/n; & \text{L(i,j)} \neq 0 \\ (1-\alpha) \times 1/n; & \text{L(i,j)=0} \end{cases}$$

And

$$s(i,j) = \beta \times 1/d_i + (1-\beta) \times Sim(at_j, con_i) \ ,$$

Then get the new $n \times n$ transition probability matrix $Q^{'}(a)$.

Step 2. Use matrix transforming and the theory of stochastic complement; calculate the rank values $\left\| \phi_i^{'}(a) \right\|_1$ with semantic relevance included.

Step 3. Calculate the updating frequency of each website, and combine the rank values with it, as shown in (5) [9].

$$SR(s_i) = \lambda \times \left\| \phi_i^{'}(\alpha) \right\|_1 + (1-\lambda) \times Freq(s_i) \qquad (5)$$

Where $\left\| \phi_i^{'}(a) \right\|_1$ denotes the rank value of site *i*, $\lambda$ is a damping factor, $0<\lambda<1$. Its value is determined by the type of websites. i.e. if the site is a Blog, due to the characteristics of Blogs, the information from Blogs is not so authoritative as some big news websites. But Blogs always update frequently, they should have a higher site updating frequency. The factor $\lambda$ is usually set above *0.7*. But for news websites or financial websites, the authority might be more important. $\lambda$ is often set between *0.2* and *0.7*. At last if the websites are official corporation websites, which might have little updating for a long time. The factor might be set under *0.2*.

## IV. EXPERIMENTS

### A. Datasets

In our experiments, the data corpus is the financial data from Haitianyuan Knowledge-Service platform[5], which was crawled from Chinese financial websites in the year of 2008. After the data was purified, it still contains *1,712,739* pages in total.

The Web pages are partitioned into websites; the rules of domain definition and domain classification are used here. Meanwhile considering that there might be subsites existed, the adjacent word before the domain part are used as the name for a website.

Finally we get *2,662* websites, wherein the largest site contains *489,028* Web pages while the smallest sites only have 1 page; 478 sites have more than 50 pages, and the count of the pages in the front 35 largest websites holds 96.1% of all the Web pages. The distribution of websites size nearly follows a power law [6]. We just use the front *35* largest websites in our experiments.

Site ranking may be impacted by lots of things, and there might be subjective preferences here, so there still hasn't been an authoritative benchmark until now. These years some third-party websites provide a relatively benchmark, such as Alexa[6]. The ranking in Alexa is based on three months of aggregated historical traffic data from millions of Alexa Toolbar users and data obtained from other, diverse traffic data sources, and is a combined measure of page views and users (reach). As a first step, Alexa computes the reach and number of page views for all sites on the Web on a daily basis. The main Alexa ranking is based on a value derived

[5] http://www.haitianyuan.com/
[6] http://www.alexa.com/site/ds/top_sites

from these two quantities averaged over time (so that the rank of a site reflects both the number of users who visit that site as well as the number of pages on the site viewed by those users) [16]. Due to the considerable quantity of samples, Alexa has been widely used to assess the popularity of a site and has a very high authority. In this paper, the ranking of *35* websites from Alexa is used as our baseline.

### *B. Performance Evaluation*

To evaluate the performance of the ranking, the modified *Kendall's τ distance* and *Spearman's distance* are used. Mathematically the evaluation of the ranking results can be abstracted as calculating the similarity between two ranking lists, the ranking results and the baseline. *Kendall's τ distance* and *Spearman's distance* have been widely implemented in mathematics [17]. However most of search engine users only care about *top k* results. We have to make some modification on *Kendall's τ distance* and *Spearman's distance* first.

a) The modified *Kendall's τ distance*

In search engines, the calculation of the similarity between two ranking results is different from that in mathematics. i.e. both the number of two ranking lists is *k*, but the ranking elements might be different, some elements might only exist in one of the two lists. Concretely according to the distribution of the elements in the two lists, we divided them into four situations: □. The elements *i* and *j* both exist in the lists $\tau_1$ and $\tau_2$; ②. The elements *i* and *j* both exist in one list $\tau_1$ or $\tau_2$, but only one element exists in the other list $\tau_2$ or $\tau_1$; ③. The element *i* only exists in one list $\tau_1$ or $\tau_2$, and the element *j* exists in the other; ④. The elements *i* and *j* only exist in one list $\tau_1$ or $\tau_2$, but none exists in the other.

In ① and ②, the calculation of $\overline{K}_{i,j}(\tau_1,\tau_2)$ has no difference, however in ③ $\overline{K}_{i,j}(\tau_1,\tau_2)$ is directly set to 1, in ④ $\overline{K}_{i,j}(\tau_1,\tau_2)$ is set to *p*, and in practice is usually set to *0*. To sum up, the calculation of the modified *Kendall's τ distance* is shown in (6) [17].

$$K^{(0)}(\tau_1,\tau_2)=(k-z)(2k+1)+\sum_{i,j\in z}\overline{K}^{(0)}_{i,j}(\tau_1,\tau_2)-\sum_{j\in S}\tau_1(j)-\sum_{j\in T}\tau_2(j) \tag{6}$$

Where $D=D_1\cup D_2$ , $Z=D_1\cap D_2$ , $S=D_1\setminus D_2$ , $T=D_2\setminus D_1, z=|Z|, |S|=|T|=k-z$ , $|D|=2k-z$ .

b) The modified *Spearman's distance*

In mathematics, the *Spearman's Footrule distance* between two lists $\tau_1$ and $\tau_2$ is defined as shown in (7).

$$F(\tau_1,\tau_2)=\sum_{i=1}^{n}|\tau_1(i)-\tau_2(i)| \tag{7}$$

When *n* is even, the max. of $F(\tau_1,\tau_2)$ is $n^2/2$; and it is *(n+1)(n-1)/2* while *n* is odd. The same as *Kendall's τ distance*, its max. appears when one sequence is the reverse of the other.

Let $l$ be a real number greater than $k$, based on the given lists $\tau_1$ and $\tau_2$ we define two function $\tau_1'$ and $\tau_2'$ over $D_1\cup D_2$, as shown in (8).

$$\tau_1'(i)=\begin{cases}\tau_1(i), & i\in D_1\\ l, & i\notin D_1\end{cases} \tag{8}$$

Then the calculation of the modified *Spearman's Footrule distance* is defined as shown in (9) [17].

$$F^{(l)}(\tau_1,\tau_2)=2(k-z)l+\sum_{i\in z}|\tau_1(i)-\tau_2(i)|-\sum_{i\in S}\tau_1(i)-\sum_{i\in T}\tau_2(i) \tag{9}$$

It has been proved that, when $l=\frac{3k-z+1}{2}$, $F^{(l)}(\tau_1,\tau_2)$ has its minimum $F^{(\frac{3k-z+1}{2})}(\tau_1,\tau_2)$.

### *C. Results and Discussions*

STRank algorithm has two damping factors $\beta$ and $\lambda$ to be determined. Experiments are conducted on semantic relevance or time frequency separately. As shown in Fig. 4 and Fig. 5, with different $\beta$ and $\lambda$, the modified *Kendall's τ distance* between our ranking results and the baseline was computed, wherein *ar* represents the result of the AggregateRank algorithm.

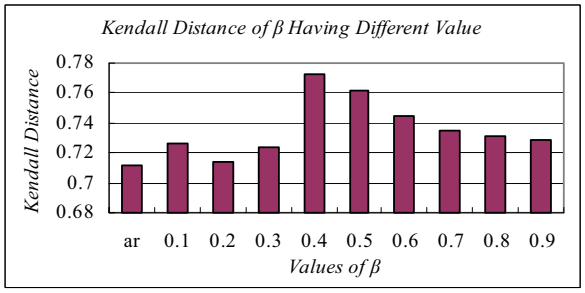


Figure 4. Kendall Distance of $\beta$ Having Different Value

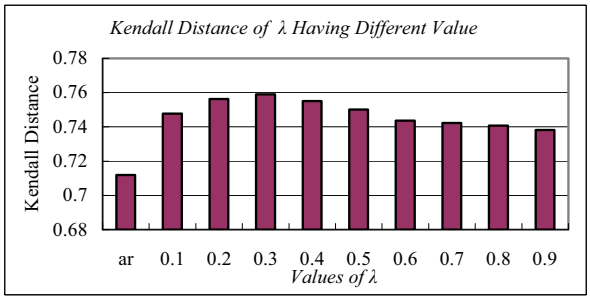


Figure 5. Kendall Distance of $\lambda$ Having Different Value

When $\beta$ is between *0.4* and *0.6*, the ranking is satisfied and is improved by *5%-7%* compared with the AggregateRank algorithm; when $\lambda$ is between *0.2* and *0.5*, the ranking improves *3%-5%*. And in our experiments below, the values of $\beta$ and $\lambda$ are set to *0.5* and *0.4* respectively.

Figure 6 lists the performance evaluation of SiteRank,

AggregateRank, STRank and PageRankSum based on the modified *Kendall's τ distance*. From this figure, we can see that the PageRankSum is the best approximation to the baseline, and the STRank algorithm has a previous advantage over SiteRank and AggregateRank. When we got different top *k* sites, the STRank algorithm all has an improvement by *3%-10%* than AggregateRank.

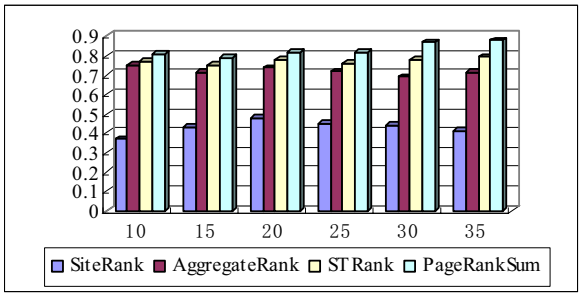


Figure 6. Kendall Distance Contrast Chart of Different Site Number

Using the modified *Spearman's Footrule distance*, we got the same results averagely improved by *6.2%*. More details on the average performance evaluation on different top *k* sites (separately based on the modified *Kendall's τ distance* and the modified *Spearman's Footrule distance*) are shown in Table II.

TABLE II. AVERAGE KENDALL DISTANCE AND SPEARMAN DISTANCE CONTRAST

| The ranking algorithm | the modified *Kendall's τ distance* | the modified *Spearman's distance* |
|---|---|---|
| **SiteRank** | 0.413495102 | 0.497268 |
| **AggregateRank** | 0.711938111 | 0.753809 |
| **STRank** | 0.773753 | 0.863333 |
| **PageRankSum** | 0.878028021 | 0.8653111 |

After the comparison on similarity, we also examine the running time of our algorithm. Due to the offline characteristic of ranking computing and distributed parallel computing, the STRank algorithm can converge faster than other aforementioned algorithms. The result is satisfied.

## V. CONCLUSIONS AND FUTURE WORKS

We propose the STRank algorithm in this paper, which take use of semantic relevance and time frequency for website ranking. Our experiments conducted on the benchmark dataset verified the effectiveness and efficiency of the proposed algorithm. We also discuss the evaluation criteria of site ranking in the last part of this paper. Since the traditional *Kendall's τ distance* and *Spearman's Footrule distance* are not appropriate enough, we designed a modification version of evaluation measures for our purpose. For future work it should be valuable to combine site hierarchy structure analysis and web page blocking into website ranking algorithm.

## ACKNOWLEDGMENT

This investigation was supported in part by the National Natural Science Foundation of China (No. 60703015) and the National 863 Program of China (No. 2006AA01Z197).

## REFERENCES

[1] E.D.Greengrass. Information Retrieval: A Survey. Information Retrieval Laboratory. 2000, 1(3):112~255.

[2] Robertson S. E., Sparck Jones K. Relevance weighting of search terms. Journal of the American Society for Information Science, 1976, 27(3), Pages: 129-146.

[3] Robertson S. E., Walker S., Beaulieu M. M., et al. Okapi at TREC-4. In Proceedings of the 4th Text REtrieval Conference (TREC-4), 1999, Pages: 73-96.

[4] Page L., Brin S., Motwani R., et al. The PageRank citation ranking: Bringing order to the Web (1999-66): Stanford Digital Library Technologies Project. 1998.

[5] Kleinberg J. M. Authoritative sources in a hyperlinked environment. Journal of the ACM, 1999, 46(5), Pages: 604-632.

[6] Krishna Bharat, Bay-Wei Chang, Monika Henzinger, and Matthias Ruhl. Who links to whom: Mining linkage between Websites. In IEEE International Conference on Data Mining (ICDM '01), San Jose, California, November 2001.

[7] Qiancheng Jiang and Yan Zhang. SiteRank-Based Crawling Ordering Strategy for Search Engines. Seventh International Conference on Computer and Information Technology. 2007, Pages: 259-263.

[8] Jie Wu, Karl Aberer. Using SiteRank for Decentralized Computation of Web Document Ranking. In Proceeding of the 3rd Intl. Conference on Adaptive Hypermedia and Adaptive WebBased Systems. Lecture Notes in Computer Science. 2004, ISSU 3137, pages 265-274.

[9] Guang Feng, Tieyan Liu, Ying Wang, et al. AggregateRank: Bringing Order to Websites. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, Seattle, Washington, USA. 2006, Pages: 75 - 82.

[10] Karl Aberer and Jie Wu. A framework for decentralized ranking in Web information retrieval. In Web Technologies and Applications: Proceedings of 5th Asia-Pacific Web Conference, APWeb 2003, volume LNCS 2642, pages 213-226, Xi'an, China, September 2003. Springer-Verlag. September 27-29, 2003.

[11] Jie Wu and Karl Aberer. Using siterank in p2p information retrieval. Technical Report IC/2004/31, Swiss Federal Institute of Technology, Lausanne, Switzerland, March 2004.

[12] Nan Liu, Christopher C. Yang. Extracting a Website's Content Structure From its Link Structure. EEE/WSE'07, 2007: 73~80.

[13] Tao Qin, Tie-Yan Liu, et al. Learning to rank relational objects and its application to web search. In Proceeding of the 17th international conference on World Wide Web, Beijing, China, April 21–25, 2008. Pages: 407-416.

[14] Lei Yang, Lei Qi, et al. Link analysis using time series of web graphs. In Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, Lisbon, Portugal, November 6-8, 2007. Pages: 1011-1014.

[15] http://en.wikipedia.org/wiki/Anchor_text

[16] http://www.alexa.com/site/help/traffic_learn_more

[17] Ronald Fagin, Ravi Kumar and D. Sivakumar. Comparing Top k Lists. Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, SIAM, Philadelphia, 2003: 28~36.