Cohesive Arc Measures for Web Navigation

Wookey Lee, Nidhi Rustagi Arora and Youngkuk Kim[1]

Inha University, Incheon
[1]Chungnam National University, Seoul, South Korea
E-mail : 206577@inha.ac.kr, rustagi.nidhi@gmail.com

ABSTRACT

Internet has emerged as largest media which provides even a single user to market their products and publish desired information. As a result web holds large amount of related information distributed over multiple web pages. The current search engines search for all the entered keywords in a single webpage and rank the resulting set of web pages as an answer to the user query. But this approach fails to retrieve the pair of web pages which contains more relevant information for users search. We introduce a new search paradigm which gives different weights to the query keywords according to their order of appearance. We propose a new arc weight measure that assigns more relevance to the pair of web pages with alternate keywords present so that the pair of web pages which contains related but distributed information can be presented to the user.

1. INTRODUCTION

The ubiquitous nature of Internet has been largely responsible for its widespread use and emergence as the largest publishing media. The web users not only have free access to this magnanimous information but also have the means to publish their own desired information or propagate their articles. Hence more and more social groups, organizations, educational institutes have already made their representations in this web space. The detailed facts and data have to be organized properly for the ease and convenience of the user, as a result most of the major websites are ionized hierarchically with information spread over multiple pages.

The intelligent web user chooses a search engine of his choice to get the required information from this vast amount of hyperspace. The foundation for most of the currently well known search engines is "keyword-based searching" where query terms entered by the user are processed to produce a flat one dimensional ranked list of web pages as the final answer. The current processing technique involves transforming the entered user query into conjunctive form i.e. as independent keywords joined by and operator. The set of web pages satisfying the conjunctive form of the query are then enumerated using some ranking measure and presented to the user in decreasing order of relevance. But such a system fails to provide appropriate answer from the pool of existing web site structure with keywords spread over multiple web pages. This does not mean that the current engines do not provide relevant information, but we would like to highlight the fact that they fail to utilize the structure of the websites. Even though the URL set returned by major search engines are well-organized, yet it can be very misleading for the user to browse through a set of connected webpages due to the presence of back-links and cross-reference links.

Consider for instance the simple user requirement of collecting information about the university where there are research activities in his specific field of

interest. As a specific case, consider a student who wants to take admission where he can pursue his research interest in the area of data structures. This naïve user search comprises of basically 3 keywords: "*university*" and "*data structures*" or another possible set of search keywords can be *university research data structures*. If this query is issued to the currently renowned search engines they fail to retrieve many relevant documents because none of the university home pages would contain information about a research area or subject. There would be computer science department and professors in many universities whose research area is basically data structures but the existing search engine technology fails to retrieve such document pairs. Secondly, most of the current search engines basically retrieve documents on the basis of presence of all the query keywords rather than the context in which they are related to each other. If additional keywords are added to the query for being more specific like location of university e.g. "USA" or "Australia" , the precision level of result decreases further. In fact some of the web pages will be relevant to one aspect and some other highlighting another aspect of the user query. Millions of students will be wasting lot of their time in going to the individual web pages of various universities and then checking for their required facts unless they search for some particular university or professor. They can either browse through each of the university websites individually or then collect information or search by looking through the names of renowned professors in the respective area. But both these methods are very time consuming and cumbersome. A better alternative would be to provide university home page and the professors or project page paired together so that the web user can contact the related personnel or look for admission details without navigating various search windows.

We would like to propose a new context specific search paradigm where keywords are not perceived as independent terms but are believed to adhere together to form one logical unit which should be the basis for retrieval. We define such kind of query as "Cohesive" query where the query keywords are believed to be logically related to each other to make up a search criteria. Another important and unique aspect of our system is that the order of query keywords is an important indication to the hierarchy or organization of the information content. We believe that the first keyword entered by the user forms the core part of the search requirement in this hyperspace and the remaining set of keywords indicate the search criteria for selection of relevant webpages in the specified domain. This paper focuses on producing web object pairs as the final answer to web users search requirement as opposed to one dimensional ranked list of web pages. Hence this paper utilizes both the content as well as the structure of the websites in order to judge the relevance with respect to user query . The proposed system takes the query keywords in their disjunctive form and performs search for all the keywords in web pages belonging to the same domain or website. The web pages are not any independent set of web pages which are paired together due to the presence of keywords but the web pages paired together for the result are also logically and conceptually related to each other. The aim of the proposed system is to pair web pages which contains information about the query keywords and they all belong to the same domain.

There are various research activities which focus on producing a small web graph or tree as the final answer presented to the web user as against ranked list of single web pages. But unlike many previous research works, our proposed framework takes into account more than two keywords. There has been a very well known proven fact that the user requirements can easily be expressed by maximum of 5 query keywords [13]. Hence we have assumed that the user query can be anything from two keywords up to a maximum of five keywords.

2. RELATED WORK

Integrating hyperlink structure with keyword based searching has been introduced long ago and quite successfully implemented by various search engines. Lawrence Page and Sergey Brin [10] introduced the concept of Page Rank which forms the basis of Google search engine. D. Gibson, J.M. Kleinberg and P.Raghavan [9] have suggested the use of clustering

web pages in order to reduce the number of URLs that match query terms. There have been many other research works that use graph theoretic properties as representation of the web page and hyperlink structures to find out strongly related web pages [4, 11]. Recently there has been a lot of research work published on the rationale of producing web object pairs as answer to user query in order to enhance the efficiency of search systems. Majority of the works are focused on using the link structure of the websites or web pages and many research works use hypertext link information to produce minimal subgraphs or subtrees as solution to the user query.

A major inspiring work has been published by R.J. Bayardo, Y. Ma and R. Srikant [12] for producing all pairs of web pages whose similarity score is approve a given threshold. They have given an algorithm which finds out similar pairs of web pages above a given threshold whereas we have developed a technique which finds out related pairs of web pages with keywords spread over them.

W. Li, K. Candan, Q. Vu and D. Agarwal [1] introduced the concept of "information unit" which consists of multiple physical web pages as one atomic retrieval unit. They have maintained an index of independent keywords and links on the basis of presence of query keywords. Their query processing heuristic involves the application of Steiner tree algorithms and its approximations. The major difference with our search system is that their heuristic involves processing the minimal subgraph for all the query keywords entered by the user. But in our search we consider only the minimal subgraphs based on the foremost query keyword entered by the user. The subgraphs are then processed for the remaining query keywords.

K. Tajima, K. Hatano, T. Matsukura, R. Sano and K. Tanaka [2] have also advocated the use of a connected subgraph as data retrieval unit to be presented to the web user. They have used minimal subgraph semantics for conjunctive query processing and score each subgraph on the basis of locality of keywords within the web page and within the subgraph. Even though their approach considers the subgraph of a document written by same author but they also like various previous searches assign equal weightage to all the query keywords. On the other hand we in our research assume that the order of keywords is an important indication to the relevance of user search requirements.

T. Yumoto and K. Tanaka [6] in another research proposed the notion of "page set ranking", which refers to ranking a set of pages as opposed to individually searched web pages. They have illustrated their approach on two kinds of specialized domain: "overview query" and "comparative query". But their framework basically scores page set using the content-analysis based search criteria, link analysis based work remains as future work. Another important factor is that their research examines all the possible page sets which could result in higher polynomial ranking cost where the complexity is O (2^n). The basic difference between our work and their approach is that our work assigns different weight to query keywords according to their order in the query. Another important factor is that we search for the query keywords in a predefined set of URLs.

3. SYSTEM ARCHETYPE

The proposed system is based on the assumption that the order of query keywords entered by the user is an important indication for his core search area and preference of information. We believe that the foremost keyword entered by the user represents his specific interests from the large amount of information available in hyperspace. Even though there are various web directories available like Google Directory [7] and Open Directory Project (ODP)[8] which contains hierarchically organized information according to the different areas of interest, but they start by providing the list of web site host URLs and requires users to themselves navigate for their specific information. Hence the users are once again left with no choice but to patiently surf the various web pages in order to collect the desired information. We propose a novel search paradigm which first selects the list of websites related to the user interest like in any available web directory and then select the web

object pairs which satisfy specific users requirements organized in multiple physical web pages of the same website.

The proposed system is based on two key aspects:

(1) Selection of domain specific websites on the basis of the first keyword entered by the user query.

(2) Processing of each of the website obtained above to find out the most relevant pair of web pages as per user requirement.

We first explain the basic definitions and terminologies used in this paper and then explain our query processing technique.

- D represents the dictionary of all the searchable keywords.
- The user query Q is perceived as a set of cohering keywords <k_1, k_2, …, k_n>. The foremost keyword mentioned in this paper refers to k_1 i.e. the first keyword entered by the user.
- W represents the set of domain specific websites which contain information about k_1. For e.g. if k_1 is company, then W contains the homepages for various international or national companies around the world.
- Each website URL $W_i \in W$ is modeled as a directed graph $G_i (V_i, E_i)$ where V_i refers to the set of webpages which belong to the domain of W_i.

We restrict V_i to the set of webpages belonging to the same domain only because combining webpages from different domains might be very misleading to the user. The web pages of two universities might be linked to each other because their students are friends or through numerous other social networks that exist on the web. Hence pairing admission office of one university and computer science faculty of another university will be very misleading and undesirable for the user.

Next we define G_i^m: (V_i^m, E_i^m), where $V_i^m \subset V_i$ and $E_i^m \subset E_i$, as the minimal subgraph obtained for each W_i. V_i^m is obtained after removing the set of webpages which lie outside the domain of W_i. E_i^m refers to the set of edges connecting V_i^m with all the back links and cycles removed.

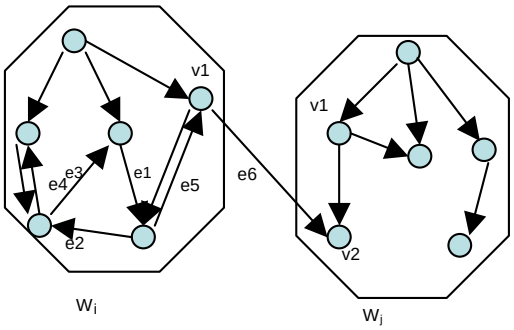

Fig. 1: Edges e1, e2 and e3 in W_i form a cycle. Edge e6 connects vertices of two different web sites.

For example, Fig. 1 shows graph representations for two websites W_i and W_j where edges e1, e2 and e3 in W_i form a cycle and the edges e4 and e5 refer to back links. Edge e6 connects v1 of W_i with v2 of W_j, hence we remove edge e6 from our consideration and do not consider vertex v2 in the subgraph of W_i and v1 in the subgraph of W_j.

There is a keyword to minimal subgraph mapping σ : D to G_i^m which maps each of the dictionary term to the root node of the minimal subgraph.

We in our research believe that the removal of cycles and cross-reference links to other domains is independent of the user query, hence this task can be done in the preprocessing phase to reduce latency in query response time. Since k_1 represents the core interest area of the users search requirements, there would be hundreds or thousands of website URLs, processing each subgraph for removal of cycles and backward links at query processing time will be an expensive approach because it results in delay for user response.

Each of the root nodes is then assigned to the inner core for enumeration of the most relevant web page pair from each W_i. A new measure called the cohesive arc measure is developed for enumerating all web pages in V_i^m.

3.1 Selecting websites specific to user interest

The first key aspect of our search paradigm is to obtain a set of basic websites which represent the

core search domain of the requirements. We illustrate this concept with a sample web query, for e.g for a query like "child care and baby blues", the set of basic websites W should contain all the relevant child related websites. Consider another query, let us say "windows xp service pack", again the set of basic websites W should contain websites related to windows with Microsoft website being one of the topmost website.

Since many of such websites will contain around thousands of webpages and hyperlinks connecting them, processing each such website at query time is nearly infeasible. Secondly, removal of all the unnecessary cycles and backward links is independent of the user query, hence we pre-process each of these websites and reduce it to their minimal subgraphs. The root node of each G_i^m along with pointers(hyperlinks) to other web pages are stored in the keyword to minimal subgraph mapping σ. This mapping is later used at query time to obtain W_i for k_1. We obtain the set of basic websites manually by using search engine Google[9] along with Google directory and Open Directory Project.

Currently the set of website URLs W for each foremost keyword are arranged according to their relevance by the page rank algorithm given by Lawrence Page and Sergey Brin [10]. Hence whenever the user query is parsed, the initial set o f basic websites W is stored in the order of relevance. Hence only the relevant web page pairs from the minimal subgraph need to be selected at query time.

3.2 Cohesive Arc Measure (CAM)

CAM is used to assign edge weights to each edge in the minimal subgraph G_i^m for enumerating all the web page pairs in V_i^m. The proposed measure gives higher weight to the pair of web pages containing one of the query keywords in each web page. The measure is based upon the traditional content analysis approach of the traditional information retrieval system. The measure works in two stages as follows:

(a) Stage 1: the information content for the each web page is quantified in the scale of 0 to 1 for each keyword entered by the user. We call such quantification of the information content as the feature vector for each web page.

(b) Stage 2: this stage enumerated all the web page pairs in G_i^m on the basis of their feature vectors. This stage can be assumed to be the actual ranking stage to find the most appropriate web page pair.

3.3 Creating feature vector

As mentioned earlier, the information content for each webpage is quantified and represented in the form of a feature vector. The conventional term frequency–inverse document frequency (*tf-idf*) and Page Rank [4] weight method is used to construct the feature vector for each web page. Since k_1 has been used already to obtain the initial set of websites W_i, we have to construct the feature vector for the remaining set of keywords $<k_2, k_3, \ldots, k_n>$. In our research we construct the feature vector in the reduced queried space and not on the basis of the overall information contained in the webpage.

let $tf(k, j)$ and $idf(k, j)$ represent the term frequency and inverse document frequency of k^{th} keyword in the j web page, then the contribution of keyword k to web page $p_j \in V_i^m$ can be obtained as

$$c_k^j = tf(k) * idf(k) \quad ...(1)$$

The feature vector $f(p_j)$ for each page $p_j \in V_i^m$ is simply an ordered collection of each of the query keyword to the web page as given in equation (2) below.

$$f(p_j) = (c_2^j, c_3^j, ..., c_n^j) \quad ...(2)$$

The feature vectors are considered to be one of the best means to analyze textual data for query processing. Their normal usage is only to find the similarity between various documents but in our proposed system we have used this for reduction of documents that can be candidate for answer to users query. For example if the feature vector for a web page is of the form (0, 0, …, 0) *i.e*. it contains none of the query keywords then there is definitely no need to take that web page for further query processing. Let *p'* be the set of web pages obtained after removing web pages with none of the queried keywords.

The conventional cosine similarity measure [5] is the most widely used similarity measure using real valued vectors or feature vectors for textual data representation. The cosine similarity measure produces high quality results across various domains for ranking single web pages as an answer to user query, where each of the documents contains almost all the queried keywords. But it fails to retrieve pair of web pages with keywords spread among them. The problem with cosine similarity measure is that it is based on the assumption of independence of keywords and each keyword forms one dimension of the vector space. Hence, if the document pairs are present orthogonally in the vector space, like the documents *a* and *b* as shown in fig. 2, then the traditional cosine measure treats them as dissimilar to each other. We have proposed our new measure which gives much higher weight to such document pairs which are present almost orthogonal to each other, so that it selects pair of web pages with keywords spread among them. The basic idea is that the cosine measure takes the product of corresponding keywords to judge the relevance, but we want to assign more weight to the alternate pairs, hence we work on the lines of taking the cross-product.

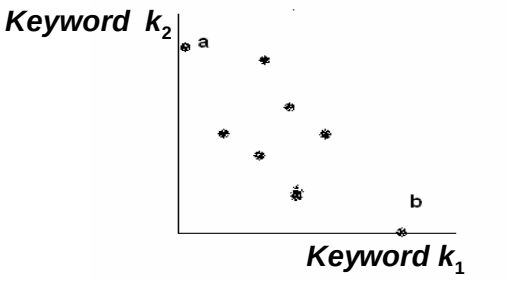

Fig. 2: Cosine measure treats a and b as dissimilar to each other

The formula for arc measure is given in equation (3) as follows:

$$w_{a,b} = f(a) \otimes f(b) \quad ...(3)$$

where $\otimes$ represents the cross product of the two feature vectors as defined below:

$$f(a) \otimes f(b)$$

$$= (c_2^a, c_3^a ..... c_n^a) \otimes (c_2^b, c_3^b ..... c_n^b)$$

$$= \sum_{x=2}^{n} c_x^a * [\sum_{y=2}^{n} c_y^b - c_x^b]$$

We explain the above formula with three query keywords for simplicity. The foremost keyword k_1 is used to select the initial set of websites W_i.

Then the feature vectors are created for the remaining two keywords (k_2,k_3) as follows:

$$f(a) = (c_2^a, c_3^a) \text{ and } f(b) = (c_2^b, c_3^b)$$

$$w_{a,b} = f(a) \otimes f(b)$$

$$= c_2^a * c_3^b + c_3^a * c_2^b \quad ...(4)$$

Where * refers to the normal multiplication between any two real values. The sample set of calculations for few web pages for a query with 2 search terms is shown in table I.

Table 1:

f(*a*)	*f*(*b*)	weight
(0.9 , 0.9)	(0.9 ,0.9)	0.81
(0.9 , 0)	(0 ,0.9)	0.486
(0.9 , 0.9)	(0 ,0.9)	0.486
(0.7, 0)	(0, 0.9)	0.378
(0 , 0.9)	(0.9 , 0)	0.324
(0.7 , 0)	(0 ,0.7)	0.294
(0.5 , 0)	(0 , 0.9)	0.27
(0.9 , 0.9)	(0.7 , 0)	0.252
(0.5 , 0)	(0, 0.7)	0.21

It can be easily inferred from table 1 that web page pairs with feature vectors like (0.9, 0) and (0,0.9) have higher edge weights as opposed to the conventional cosine similarity measure. Another unique aspect of our measure can also be observed from row 2 and row 5, that the sequence of keywords is important.

Hence our basic algorithm for query processing is explained as follows:

Step 1. Parse user query *Q* as $\{k_1, k_2, \ldots, k_n\}$

Step 2. Obtain the website urls, *W* from $\sigma(k_1)$

Step 3. For each $W_i \in W$ *do*

Step 4. Obtain minimal acyclic subgraph $G_i^m(V_i^m, E_i^m)$

Step 5. for every web page $p_i \in V_i^m$ do

Step 6. construct $f(p_i)$

Step 7. for every pair $i,j \in E_i^m$ do

Step 8. $w_{i,j} = f(p_i) \otimes f(p_j)$

Step 9. Present the top pair from each W_i to the user.

4. EXPERIMENT

We present the effectiveness of our CAM by conducting experiments on synthetic graphs and real web data. The synthetic graphs are generated randomly for a set of 100, 200 and 500 vertices. The edges for each of the graphs are generated by randomly assigning 0 or 1 to the adjacency matrix for each graph. In our experiments, we have deliberately omitted loops i.e. there is no edge from a page to itself. Each node is then assigned a feature vector consisting of two query keywords with randomly generated tf-idf values in the range of 0 to 1.

We performed the experiments on synthetic graphs and realized that our proposed measure assigns high score to the pair of web pages with alternate keywords present. The measure has given satisfactory results and assigned higher weight to web page pairs of (1,0) and (0,1) kind. But the experiments conducted also highlighted the fact that our measure assigns equal preference to all the query keywords. In other words, our measure assigns equal edge weight to the pair of nodes with (1,0) - (0,1) feature and (0, 1)-(1,0) feature. Hence we assigned a preference coefficient α to our measure, so that nodes containing keywords in same sequence as entered by the user are assigned higher weight. We illustrate this by modifying equation 4 for a two-keyword query as follows:

$$R^*_{a,b} = c_2^a * c_3^b * \alpha + c_3^a * c_2^b * (1-\alpha) \qquad ...(5)$$

where $0 \leq \alpha \leq 1$

The value of α can vary in the range of 0.5 to 1, depending upon the preferential weightage for the sequence of terms. The value of $\alpha = 0.5$ would mean that both the terms are equally important and equally contribute to evaluate the edge weight. But we have already mentioned that a key aspect of our system is

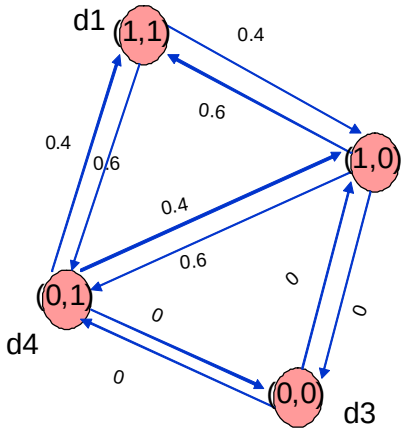

Fig. 3: Graph representation for computation of various pair of web pages.

that the sequence of query keywords entered by the user are important, hence we assign this preference coefficient α to each of the query terms while evaluation of edge weights. For example, $\alpha = 0.6$ implies that we take 60% contribution from the first term and remaining 40% contribution from the second term.

The initial set of experiments was conducted for boolean data set of 0 and 1 to see the correctness of our algorithm. The boolean values make a maximum of four possible combinations: (1,1), (1,0), (0,1) and (0,0). Hence the experiments for boolean data set need only four vertices with maximum of eighteen edges. The results of our experiments were very much satisfactory as shown in figure 3 and thus we proceed further for real tf-idf values normalized to the range of (0,1).

As mentioned earlier the basic assumption of our model is to assign high score to the pair of web pages which have alternate query keywords. But various other content based similarity measures like tf-idf based cosine measure [14], Edit distance [15] and Jaro rule [16] provide efficient results for documents containing all the query keywords. We present the effectiveness of our algorithm by comparing the results of CAM with the cosine measure for a query with two keywords. The sample of few of the tf-idf values generated for a graph with 100 vertices is shown in table II.

The top 10 similar web page pairs as evaluated by CAM and the conventional cosine measure are presented in table III. The most relevant pair according to CAM is for node pair(0, 54), the first entry under CAM which refers to the feature vectors of (0.5204, 0) and (0, 0.8799). The most relevant pair of vertices according to the cosine measure is the node pair (1, 8) which refers to the feature vectors of (0.8088, 0.7341) and (0.4554, 0.2401). Similarly the other nine pair of web pages can be interpreted for both the CAM and the cosine measure by using both table II and III together. It can easily be inferred that our CAM measure has successfully assigned higher rank to pair of vertices which contain alternate query keywords.

Table 2:

Node No.	tf-idf values for	
	k_1	K_2
0	0.5204	0
1	0.8088	0.7341
6	0.4082	0
8	0.4554	0.2401
9	0	0.3325
10	0	0.6621
13	0	0.1675
14	0	0.5418
15	0.06	0
18	0	0.5114
19	0.3307	0
25	0.6855	0
27	0.2407	0.82
30	0	0.5229
32	0.1506	0.4755
43	0.5481	0.532
54	0	0.8799
76	0	0.4341
84	0	0.1629

Since it is not feasible to present all the generated feature vectors for various web page pairs generated for our experiment. We visualize the trend for similar web page pairs according to their feature vectors in decreasing order of similarity in figure 4 as follows.

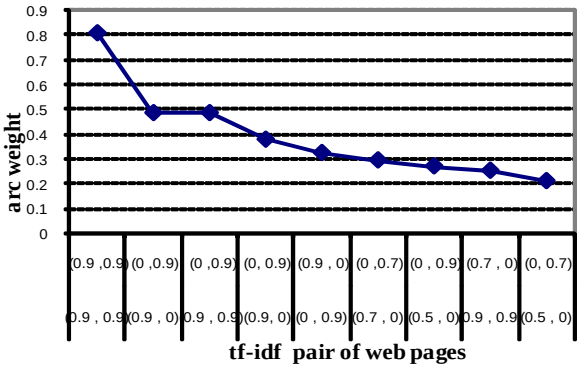

Fig. 4: Decreasing order of web page pairs according to the distribution and features of query keywords.

The calculations shown in table 3 and fig. 4 respectively, have been done by choosing the preference coefficient $\alpha = 0.6$. We obtained this value empirically for emphasizing on the sequence of query keywords. The higher values of α can be set if the system wants to exclusively focus on the foremost keyword with minor contributions from the remaining terms. We have not performed any experiments to find out the optimal values of α. We obtained similar results for a query with 3 keywords but have to see the effectiveness on real web sites and real web data.

5. CONCLUSION

In our research work, we have developed a new search paradigm which produces a pair of web pages more appropriate for an answer to the user query as against a ranked list of documents given by the current search engine. Although our experiments on synthetic graphs

Table 3:

CAM	Cosine Measure
(0 , 54)	(8 , 15)
(6 , 18)	(15 , 8)
(6 , 56)	(52 , 27)
(15 , 13)	(9 , 43)
(19 , 34)	(1 , 14)
(19 , 93)	(43 , 32)
(25 , 76)	(74 , 8)
(45 , 84)	(43 , 0)
(51 , 54)	(30 , 43)
(51 , 95)	(27 , 9)

have shown that pair of web pages with alternate keywords present are ranked higher along with pair of web pages where all the query keywords are present. we are currently doing experiments on real websites i.e we have maintained the index for websites in some specific domains and are currently evaluating the effectiveness of our approach on real world applications.

6. ACKNOWLEDGEMENT

This research was supported by the Ministry of Knowledge Economy, Korea, under the Information Technology Research Center support program supervised by the Institute of Information Technology Advancement. (grant number IITA-2008-C1090-0801-0031)

REFERENCES

1. W. Li, K. Candan, Q. Vu and D. Agrawal, "Retrieving and Organizing Web Pages by Information Unit," WWW (2001) 230-244
2. K. Tajima, K. Hatano, T. Matsukura, R. Sano and K. Tanaka. "Discovery and Retrieval of Logical Information Units In Web," WOWS (1999) 13-23
3. J. Sun, X. Wang, D. Shen, H. Zeng and Z. Chen. "CWS: A Comparative Web Search System", WWW (2006) 467-476
4. R. Andersen, F. Chung, K. Lang, "Local Graph Partitioning using Page Rank Vectors," FOCS (2006) 475-486
5. B. Neto and R. Baeza-Yates. "Modern Information Retrieval", Addison-Wesley, 2001.
6. T. Yumoto and K. Tanaka, "Page Sets As Web Search Answers," ICADL (2006), 244-253.
7. http://directory.google.com/
8. Open Directory Project. http://www.dmoz.org/
9. D. Gibson, J. M. Kleinberg and P. Raghavan. "Inferring Web Communities from Link Topology," Hypertext, pp. 225-234
10. L. Page and S. Brin. "The Anatomy of a Large Scale Hyper textual Web Search Engine" , WWW (1998), 107-117.
11. S. Chakrabarti, A. Frieze and J. Vera, "The Influence of Search Engines on Preferential Attachment," SODA (2005), 293-300.
12. R. J. Bayardo, Y. Ma and R. Srikant. "Scaling up All Pairs Similarity Search," WWW (2007), 131-140.
13. T. Phelps and R. Wilensky. "Robust Hyperlinks: Cheap, Everywhere, Now", Digital Documents and Electronic Publishing, LNCS 2023, pp. 28-43, 2000.
14. Gerry Salton. Introduction to modern information retrieval. McGraw Hill, 1987.
15. D. Gusfield algorithms on strings, trees and sequences, Cambridge University press 1998.
16. M. A. Jaro. "Advances in record linkage methodology as applied to matching the 1985 census of Tampa", Journal of American statistical association 84, 1989.