A Method of Improving Feature Vector for Web Pages Reflecting the Contents of Their Out-Linked Pages

Kazunari Sugiyama[1], Kenji Hatano[1],
Masatoshi Yoshikawa[1,2], and Shunsuke Uemura[1]

[1] Graduate School of Information Science, Nara Institute of Science and Technology,
8916-5 Takayama, Ikoma, Nara 630-0101, Japan
[2] Information Technology Center, Nagoya University,
Furo-cho, Chikusa-ku, Nagoya, Aichi 464-8601, Japan
{kazuna-s, hatano, yosikawa, uemura}@is.aist-nara.ac.jp

Abstract. TF-IDF schemes are popular for generating the feature vectors of documents. These schemes are proposed for characterizing one document. Therefore, in order to characterize Web pages using tf-idf schemes, the feature vectors of the Web pages should be reflected by the contents of Web pages linked with other pages via hyperlinks. In this paper, we propose three methods of generating feature vectors for linked documents such as Web pages. Moreover, in order to verify the effectiveness of our proposed methods, we compare our methods with current search engines and confirm their retrieval accuracy using recall precision curves.

1 Introduction

Many people use the World Wide Web (WWW) as a resource to obtain a variety of information. The amount of information on the Web has been increasing with the development of computer networks; therefore, it becomes more difficult to find valuable information amidst a large amount of information. Under these circumstances, search engines are essential tools to find information on the Web. These search engines are classified into three types: the directory type such as Yahoo![1], the robot-type such as Google[2], and meta-search engines which return the search results of some search engines. When users use the directory-type search engines, even if they can search information in a directory entering a keyword, they often cannot find information they really want. Moreover, even if the directories are meticulously constructed and Web pages are precisely classified, which directory the Web page belongs to is incomprehensible to users because not users but domain analysts determine the criterion of classification of Web pages. As a result, users encounter difficulty in finding information when using directory-type search engines. On the other hand, when users employ a robot-type search engine, the users cannot easily find valuable information because of the large amount of search results unless the users convey exact keywords to the search engines. Current robot-type search engines adopt a method for Web document retrieval such as PageRank [1] and HITS(Hypertext Induced Topic Search) [2] algorithm, which utilize the hyperlink structure of Web pages.

[1] http://www.yahoo.com/
[2] http://www.google.com/

R. Cicchetti et al. (Eds.): DEXA 2002, LNCS 2453, pp. 891–901, 2002.

In these algorithms, each hyperlink has a weight, and the terms included in a Web page is weighted based on the weight of the hyperlinks. In our opinion, however, the Web pages should be weighted by not only the weights of hyperlinks but also the contents of Web pages linked from the root Web page. Therefore, the robot-type search engines that apply these algorithms are inconvenient for users since they return the Web pages that is not relevant to queries users enter. In this paper, we propose three methods of generating a feature vector for Web pages. In our proposed methods, a feature vector of a Web page is reflected on the contents of Web pages which are linked from and are similar to the root Web page. In order to realize these processes, our prototype system analyzes the hyperlink structure of collected Web pages and classifies them using the K-means algorithm [3]. Moreover, we compare our methods with a current search engine that apply the PageRank algorithm and confirm our methods' retrieval accuracy using recall precision curves. In our experiments, we do not utilize an algorithm that reflects the weight of a hyperlink; but utilize only the algorithm that reflects the content of Web pages linked from the root Web page.

2 Related Work

A hyperlink structure is one of the features of Web pages because users can navigate the Web pages on the WWW using hyperlinks. In consideration of the way users employ the Web, similar Web pages are connected to each other by hyperlinks. Many studies in information retrieval have focused on the hyperlink structure of Web pages. In this section, we describe the searching methods using the hyperlink structure of Web pages.

We consider that the current techniques for retrieving Web documents using hyperlink structure can be classified into the following two categories:

- The retrieval technique based on the concept of "information unit."
- The retrieval technique based on the qualities of Web pages.

In regard to the first category, Tajima et al. [4] proposed the concept of "cut" to the documents on network environments such as E-mail, Netnews, and the WWW. The "cut" is composed of more than one document that has similar content and is used as the unit of document retrieval. Following this research, many researchers have proposed retrieval techniques based on the same concept, analyzing the hyperlink structure of the WWW and determining semantics of Web pages [5,6]. These information retrieval systems find the minimal sub-structures of the WWW hyperlink structure including all keywords users enter as unit of retrieval results and calculate their scores in order to rank them. However, the analysis of hyperlink structure and discovering the semantics of Web pages is time-consuming, and though they can find the minimal sub-structures exactly, the retrieval systems often find minimal sub-structures irrelevant to users' query.

Following these studies, a lot of studies have been performed based on the quality of the Web documents analyzing hyperlink structure. One of the most famous studies focuses on HITS algorithm [2]. The basic idea of HITS is that Web documents are classified into "authority" and "hub" pages, and if the degree of authority of a document has high value, the weight of terms in the document is high because the document is informative enough for query. Moreover, PageRank algorithm is applied to the search engine “Google.” The PageRank algorithm depends on the vast hyperlink structure of WWW as an indicator of an individual Web document’s value, and expresses the quality of a Web document in terms of the probability of following linked pages and that of navigating to irrelevant documents from a Web document of original interest [1]. These algorithms use only the hyperlink structure of the WWW to calculate the quality of Web documents. If the Web documents contain one specific topic as Amento et al. [7] have reported, we may use only the Web’s hyperlink structure. However, we propose the use of not only hyperlink structure but also the general contents of each Web document in order to characterize it, since actual Web documents contain a variety of content. Chakrabarti et al. [8,9] have improved the HITS algorithm paying attention to this point. They have proposed a method considering the contents of Web documents extracted by both hyperlink structure and document structure. However, it is difficult to determine the extracted partial structure of Web documents, and the retrieval accuracy of their approach fluctuates in terms of the extraction method of the partial structure.

Considering these points, we propose a simple method of improving the feature vectors of Web pages. The feature vector of a Web page should reflect the contents of its out-linked pages because the contents of a Web page are similar to those of out-linked Web pages. When we reflect the contents of out-linked Web pages on the root Web page, we classify the out-linked pages using the K-means algorithm so that our proposed method performs as an adaptive retrieval algorithm for Web pages. In our method, it is expected we can extract keywords of a Web page more effective, and can obtain higher retrieval accuracy because of exploiting linked pages from a root Web page.

3 Proposed Method

In this section, we explain our methods of improving the feature vectors of a Web page. The basic idea is that we improve the feature vectors of Web pages generated by tf-idf schemes in advance by utilizing the feature vectors of its out-linked pages, which are generated likewise by tf-idf schemes. The pages out-linked from a certain Web page can be recognized because the contents that exist near the root Web page also exist near its linked Web pages. From now on, we refer to the Web page whose feature vectors we intend to improve as the “root page”; let this “root page” be p_r. Then, a Web page which exists in $i-$th hierarchy from p_r has N_i pages, $p_{i1}, p_{i2} \cdots, p_{iN_i}$. Though the hierarchy of numbers from p_r, can establish some paths; we define i as the number of links of the shortest path. Moreover, we denote the feature vector $\boldsymbol{w}^{p_r}$ of the root page p_r as follows:

$$\boldsymbol{w}^{p_r} = (w^{p_r}_{t_1}, w^{p_r}_{t_2}, \cdots, w^{p_r}_{t_m}), \tag{1}$$

where the element $w^{p_r}_{t_k}$ of $\boldsymbol{w}^{p_r}$ is defined as follows:

$$w^{p_r}_{t_k} = \frac{tf(t_k, p_r)}{\sum_{k=1}^{m} tf(t_k, p_r)} \cdot \log \frac{N_{web}}{df(t_k)}, \qquad (k = 1, 2, \cdots, m) \tag{2}$$

where $tf(t_k, p_r)$ is the frequency of term t_k in root page p_r, N_{web} is total number of collected Web pages, and $df(t_k)$ is the number of Web pages in which term t_k appears.

From now on, we use $\boldsymbol{w}^{p_r}$ to indicate "initial feature vector". Then, we denote the improved feature vector $\boldsymbol{w}'^{p_r}$ as follows:

$$\boldsymbol{w}'^{p_r} = (w_{t_1}'^{p_r}, w_{t_2}'^{p_r}, \cdots, w_{t_m}'^{p_r}).$$

From now on, we use $\boldsymbol{w}'^{p_r}$ to indicate "improved feature vector". We propose the following three methods to improve the "initial feature vector" generated by tf-idf schmes:

1. The method of reflecting each linked page into a root page (**Method I**),
2. The method of reflecting centroid vectors of a group of out-linked Web pages from a root page (**Method II**),
3. The method of reflecting centroid vectors of a group of Web pages constructed by each i-th hierarchy from a root page (**Method III**).

3.1 Method I

This method reflects the contents of each page out-linked from root page. This method is based on the ideas that there are Web pages which are similar to the contents of pages out-linked from p_r and some pages which are similar to p_r exist closely linked pages, others may exist in distant out-linked pages. Using the ideas, we reflect the distance between p_r and its out-linked pages into each element of $\boldsymbol{w}^{p_r}$. For example, Figure 1 shows that $\boldsymbol{w}'^{p_r}$ is generated by reflecting the contents of all Web pages which exist in the second hierarchy from root page p_r. In this method, each element $w_{t_k}'^{p_r}$ of $\boldsymbol{w}'^{p_r}$ is defined as follows:

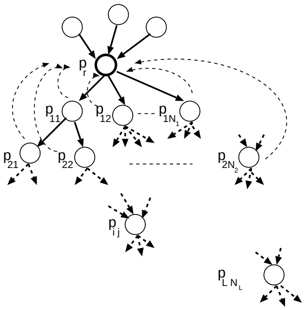

Fig. 1. The improvement of feature vectors as performed by method I.

$$w_{t_k}'^{p_r} = w_{t_k}^{p_r} + \sum_{i=1}^{L} \sum_{j=1}^{N_i} \frac{1}{dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{p_{ij}})} w_{t_k}^{p_{ij}}. \tag{3}$$

Equation (3) shows that the product of weight $w_{t_k}^{p_{ij}}$ of term t_k in out-linked page p_{ij} and the reciprocal number of the distance between $\boldsymbol{w}^{p_r}$ and $\boldsymbol{w}^{p_{ij}}$, $dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{p_{ij}})$ in vector space is added to weight $w_{t_k}^{p_r}$ of term t_k $\boldsymbol{w}^{p_r}$ calculated by equation (2), with regard to all out-linked pages which exist in the L-th hierarchy from p_r. We define $dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{p_{ij}})$ as follows:

$$dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{p_{ij}}) = \sqrt{\sum_{k=1}^{m} (w_{t_k}^{p_r} - w_{t_k}^{p_{ij}})^2}.$$

3.2 Method II

In this method, we cluster the set of all Web pages which exist up to i-th hierarchy from p_r, and generate $\boldsymbol{w}'^{p_r}$ by reflecting its centroid vectors on the $\boldsymbol{w}^{p_r}$ of root page

p_r. This method is based on the idea that when we observe the linked pages from p_r, the out-linked pages can be classified into certain topics. Using this idea, we reflect the distance between $\boldsymbol{w}^{p_r}$ and the centroid vector of the cluster on each element $w_{t_k}^{p_r}$ of $\boldsymbol{w}^{p_r}$. In other words, we make a group of Web pages, G_i as defined by equation (4),

$$\begin{aligned} G_i = \{&p_{11}, p_{12}, \cdots, p_{1N_1}, \\ &p_{21}, p_{22}, \cdots, p_{1N_2}, \\ &p_{i1}, p_{i2}, \cdots, p_{iN_i}\}, \end{aligned} \tag{4}$$

and make K clusters in G_i by means of the K-means algorithm [3]. We make $\boldsymbol{w}'^{p_r}$ by reflecting the distance between each centroid vector, $\boldsymbol{w}^{g_c}(c = 1, 2, \cdots, K)$ and $\boldsymbol{w}^{p_r}$. For instance, Figure 2 shows that we make G_2, a group of Web pages which exist up to two links away from p_r, and make an improved feature vector by reflecting the K centroid vectors constructed by clustering G_2 on $\boldsymbol{w}^{p_r}$. In this method, each element $w_{t_k}'^{p_r}$ of $\boldsymbol{w}'^{p_r}$ is defined as follows:

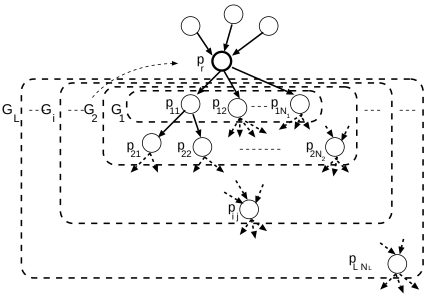

Fig. 2. The improvement of feature vectors as performed by method II.

$$w_{t_k}'^{p_r} = w_{t_k}^{p_r} + \sum_{c=1}^{K} \frac{1}{dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_c})} w_{t_k}^{g_c}. \tag{5}$$

Equation(5) shows that the product of element $w_{t_k}^{g_c}$ of the centroid vector generated from the group of Web pages, and the reciprocal of the distance between $\boldsymbol{w}^{p_r}$, and $\boldsymbol{w}_c^{g}$, $dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_c})$ is added to weight $w_{t_k}^{p_r}$ calculated by equation (2), with regard to the number of clusters, K. We define $dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_c})$ as follows:

$$dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_c}) = \sqrt{\sum_{k=1}^{m} (w_{t_k}^{p_r} - w_{t_k}^{p_{ij}})^2}.$$

3.3 Method III

In this method, we construct G_i, the group of Web pages which exist in each hierarchy linked from p_r, and generate $\boldsymbol{w}'^{p_r}$ by reflecting its centroid vector on $\boldsymbol{w}^{p_r}$. This method is based on the idea that the Web pages which exist in a certain hierarchy from p_r can be classified into certain topics. Using this idea, we reflect the distance between $\boldsymbol{w}^{p_r}$ and the centroid vector of a cluster on each element of the initial feature vector. In other words, we create a group of Web pages G_i as defined by equation (6),

$$G_i = \{p_{i1}, p_{i2}, \cdots, p_{iN_i}\}, \tag{6}$$

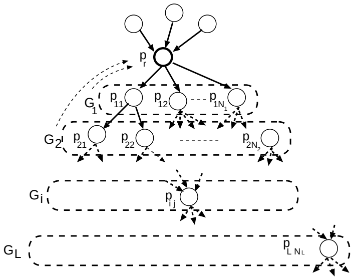

Fig. 3. The improvement of feature vectors as performed by method III.

and make K clusters in G_i by means of a K-means algorithm. We create an improved feature vector by reflecting the distance between each centroid vector $\boldsymbol{w}^{g_{ic}}(c = 1, 2, \cdots, K)$ and initial feature vector $\boldsymbol{w}^{p_r}$ of root page p_r. For instance, Figure.3 shows that we make a group of Web pages, G_1 and G_2, which exist in each hierarchy up to two links away from the root page p_r, and make an improved feature vector by reflecting the centroid vector constructed in each Web page group, G_1 and G_2, on $\boldsymbol{w}^{p_r}$. In this method, each element $w'^{p_r}_{t_k}$ of $\boldsymbol{w}'^{p_r}$ is defined as follows:

$$w'^{p_r}_{t_k} = w^{p_r}_{t_k} + \sum_{i=1}^{L}\sum_{c=1}^{K} \frac{1}{dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_{ic}})} w^{g_{ic}}_{t_k}. \tag{7}$$

Equation (7) shows that the product of element $w^{g_{ic}}_{t_k}$ of the centroid vector, which is constructed from each group of Web pages constructed in each hierarchy up to i link away from p_r, and the reciprocal of the distance between $\boldsymbol{w}^{p_r}$ and $\boldsymbol{w}^{g_{ic}}$ in vector space, $dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_{ic}})$ is added to weight $w^{p_r}_{t_k}$ of term t_k calculated by equation (2), with regard to all centroid vectors constructed in each hierarchy up to the L-th link away from p_r. We define $dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_{ic}})$ as follows:

$$dis(\boldsymbol{w}^{p_r}, \boldsymbol{w}^{g_{ic}}) = \sqrt{\sum_{k=1}^{m}(w^{p_r}_{t_k} - w^{g_{ic}}_{t_k})^2}.$$

4 Experimental Results

We conducted the experiments in order to verify whether or not a term which has high value in the improved feature vector can be the keyword of a Web page (**Experiment I**), and to verify the precision of a search using an improved feature vector compared with the search results of an existing search engine (**Experiment II**). Our method described in section3is implemented using Perl on a desktop PC (CPU: AMD Athron 1.4GHz, Memory: 1GBytes, OS: Vine Linux2.1.5), and the experiments are conducted for 0.8GByte Web pages (about 250,000URLs).

4.1 Experiment I

We conduct the experiment to verify whether a keyword of the root Web page is extracted more precisely compared with tf-idf schemes. The procedure for this experiment is as follows:

1. We choose five terms beforehand as the correct keywords per Web page.
2. We calculate the cumulative rate of right answer which denote how many keywords are contained up to the top ten terms in the improved feature vector.

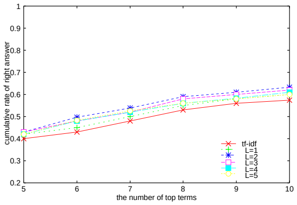

Fig. 4. Cumulative rate of right answer obtained by Method I.

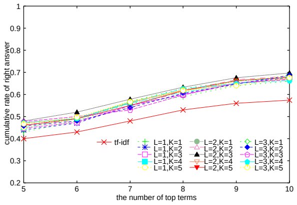

Fig. 5. Cumulative rate of right answer obtained by Method II.

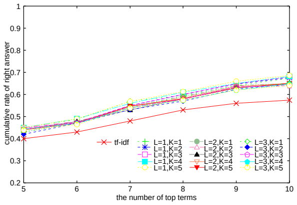

Fig. 6. Cumulative rate of right answer obtained by Method III.

Figures 4, 5, and 6 show the results obtained using proposed Methods I, II, III described in section3, respectively. Figure 4 shows the following characteristics of Method I:

- The keywords of a Web page are more exactly extracted by utilizing the contents of each Web page which exists up to $L = 2$, i.e., two links away from the p_r as compared with tf-idf schemes.
- The result of $L = 3$, i.e., in the case of reflecting the contents of a Web page three links away from p_r on its initial feature vector, is almost the same as the result of $L = 2$. Therefore, we found that, in Method I, an improved feature vector generated by urilizing Web pages at least two links away from p_r effectively extracted the keywords of the Web page.

Figure 5 shows the following characteristics of Method II:

- The result of $L = 2, K = 3$, i.e., in the case of generating an improved feature vector by exploiting the centroid vectors of three clusters constructed from a group of Web pages up to two links away from p_r, shows the most effective result in extracting the keywords of the Web page.

Finally, Figure 6 shows the following characteristic of Method III:

- The result of $L \geq 2$, i.e., in the case of generating improved feature vectors by exploiting the centroid vectors of clusters constructed from a group of Web pages which exists more than two links away from p_r, shows that it is less effective in extracting keywords of a Web page than $L = 1$, i.e., in the case of generating an improved feature vector by exploiting the centroid vector of clusters from a group of Web pages up to one link away from p_r, in extracting keywords of a Web page. In Method III, we cluster Web pages which exist in each hierarchy from p_r like Figure 3, so the continuity of contents between Web pages is lost. That is why we cannot obtain better results in the case of $L \geq 2$.

4.2 Experiment II

To compare our method with existing search engines, we conducted an experiment to verify whether an improved feature vector is effective as Web page index. Figure 7 shows an overview of our system. This system contains the following functions:

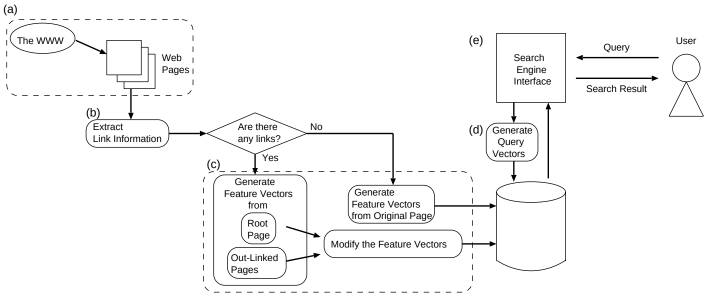

Fig. 7. Overview of our system.

(a) Collecting Web pages
This function collects Japanese Web pages by exploiting a Web crawler.

(b) Extracting link information
The link information of each collected Web pages are extracted. In this function, only forwarded links are extracted.

(c) Generating feature vectors of Web page
A feature vector of a Web page generated by tf-idf schemes is improved by exploiting our proposed method described in section 3.

(d) Generating query vector
We denote query vector $\boldsymbol{Q}$ as follows:

$$\boldsymbol{Q} = (q_{t_1}, q_{t_2}, \cdots, q_{t_m}), \qquad (8)$$

where t_k is an index term, and the base of equation (8) is as same as that of equation (1). Each element q_{t_k} of equation (8) is defined as follows:

$$q_{t_k} = \left(0.5 + \frac{0.5 \cdot Qf(t_k)}{\sum_{k=1}^{m} Qf(t_k)}\right) \times \log \frac{N_{web}}{df(t_k)} \quad (k = 1, 2, \cdots, m), \qquad (9)$$

where $Qf(t_k)$, N_{web}, and $df(t_k)$ is the number of index terms t_k, the total number of collected Web pages, and the number of Web pages in which the term t_k appears, respectively. As reported in [10], equation (9) is the element of a query vector which brings best search result.

(e) Search Engine interface
A user enters queries through this interface, and the system shows the search results to the user. This system calculates the similarity $sim(\boldsymbol{w}^p, \boldsymbol{Q})$ between feature vector $\boldsymbol{w}^p$ of Web page p and query vector $\boldsymbol{Q}$, and shows search results in descending order of $sim(\boldsymbol{w}_p, \boldsymbol{Q})$. The $sim(\boldsymbol{w}_p, \boldsymbol{Q})$ is defined as follows:

$$sim(\boldsymbol{w}^p, \boldsymbol{Q}) = \frac{\boldsymbol{w}^p \cdot \boldsymbol{Q}}{|\boldsymbol{w}^p| \cdot |\boldsymbol{Q}|}. \qquad (10)$$

In our study, considering that Web pages have varied contents, we choose 14 Japanese terms as queries in order to cover a various contents of Web pages. We prepare ten relevance Web pages with regard to each query.

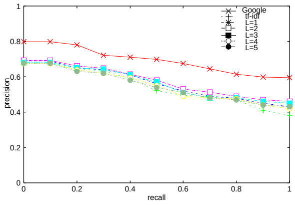

Fig. 8. Comparison of search accuracy obtained using Method I and that using an existing search engine.

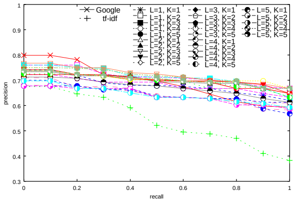

Fig. 9. Comparison of search accuracy obtained using Method II and that using an existing search engine.

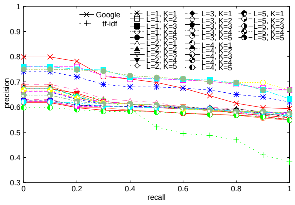

Fig. 10. Comparison of search accuracy obtained using Method III and that using an existing search engine.

We compare our method with the existing search engine Google[3], and evaluate retrieval accuracy using recall precision curves based on "precision at 11 standard recall levels" described in [11,12]. Figures 8, 9, and 10 show the results obtained by using the proposed Method I, II, and III described in section 3, respectively. Figure 8 shows the following fact with regard to Method I:

- The result of $L = 2, 3$, in the case of reflecting the contents of each Web page which exists two or three links away from root page p_r become better than $L = 1$, i.e., reflecting the content of each Web page which exists one link away from root page p_r.
- There is little difference between the result of $L = 2$ and that of $L = 3$.
- The result of $L \geq 4$ is worse than that of $L = 2, 3$.

In addition, we cannot obtain better retrieval accuracy than that of Google in exploiting the improved feature vector generated by Method I. Therefore, the Method I is not appropriate for creating an index for searching Web pages. Figure 9 shows the following characteristic of Method II:

- We can obtain higher retrieval accuracy at the point where recall is high than Google.
- The higher retrieval accuracy is obtained in the case of $L = 1, K = 3$, i.e. in the case of generating an improved feature vector by exploiting the centroid vector of three clusters constructed from a group of Web pages one link away from p_r, and $L = 2, K = 3$, i.e., in the case of generating an improved feature vector by exploiting the centroid vectors of three clusters constructed from a group of Web pages two links away from p_r. In particular, the result of $L = 2, K = 3$ shows that the contents of a Web page can be integrated up to two links away from p_r.

Moreover, we note that by using an improved feature vector, Web pages which correspond to the contents of a query can be ranked in an upper level of preference in comparison with Google which tends to rank many-linked Web pages with many links to other pages in an upper position. Finally, Figure 10 shows the following characteristic of Method III:

[3] http://www.google.com/intl/ja/

- The retrieval accuracy in the case of $L \geq 2$ is inferior to that of Method II. In Method III, Web pages are clustered in terms of their hierarchy from p_r as seen in Figure 3; thus, the continuity of the contents between Web pages is lost. This is why we cannot obtain better retrieval accuracy in the case of $L \geq 2$.

5 Concluding Remarks

We proposed three methods of generating feature vectors for Web pages. In our proposed methods, a feature vector of a Web page is reflected in the contents of out-linked Web pages that are similar to the root page. We also conducted experiments in order to verify the effectiveness of our proposed methods, obtaining the following results:

- We could extract appropriate index terms in the task of extracting the keywords of Web pages by using contents of their out-linked pages.
- We could obtain higher retrieval accuracy by improving feature vector of a root page exploiting feature vectors of Web pages two or three links away from the root page.

Our future work focuses on the following:

- In this paper, we use all the clusters generated from Web pages; however, we should select clusters which are more appropriate for generating feature vectors in order to obtain higher search accuracy.
- We have to verify the effectiveness of our method by conducting experiments using reference collections for Web pages.

References

1. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *In proc. of the 7th International World Wide Web Conference*, pages 107–117, 1998.
2. J. M. Kleinberg. Authoritative sources in a hyperlinked environment. In *In proc. of ACMSIAM Symposium on Discrete Algorithms*, 1998.
3. J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proc. of the 5th Berkeley Symposium on Mathmatical Statistics and Probability*, pages 281–297, 1967.
4. K. Tajima, Y. Mizuuchi, M. Kitagawa, and K. Tanaka. Cut as a querying unit for www, netnews, e-mail. In *Proc. of the 1998 ACM Hypertext Conference*, pages 235–244, 1998.
5. K. Tajima, K. Hatano, T. Matsukura, R. Sano, and K. Tanaka. Discovery and retrieval of logical information units in web. In *Proc. of the 1999 ACM Digital Libraries Workshop on Organizing Web Space*, 1999.
6. W. Li, K. Selc uk Candan, Q. Vu, and D. Agrawal. Retrieving and organizing web pages by 'information unit'. In *Proc. of the 10th International World Wide Web Conference*, pages 230–244, 2001.
7. B. Amento, L. Terveen, andW. Hill. Does 'authority' mean quality? predicting expert quality ratings of web documents. In *Proc. of the 22nd annual international ACM SIGIR Conference(SIGIR2000)*, pages 296–303, 2000.
8. S. Chakrabarti. Integrating the document object model with hyperlinks for enhanced topic distillation and information extraction. In *proc. of the 10th International World Wide Web Conference*, pages 211–220, 2001.

9. S. Chakrabarti, M. Joshi, and V. Tawde. Enhanced Topic Distillation using Text, Markup Tags, and Hyperlinks. In *Proc. of the 23nd annual international ACM SIGIR Conference(SIGIR2001)*, pages 208–216, 2001.
10. G. Salton and C. Buckley. *Term-weighting approaches in automatic retrieval*. Information Processing & Management, 24(5):513-523, 1988.
11. I. H. Witten and A. Moffatand T. C. Bell. *Managing Gigabytes*. Van Nostrand Reinhold, 149-150, 1994.
12. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, 1999.