DISCOVERY OF SEMANTIC RELATIONSHIPS AMONG WEB PAGES BASED ON WEB TOPIC STRUCTURES

Takeshi Matsukura and Hiroyuki Kondo and Yoichi Hirata
Graduate School of Science and Technology
Kobe University
{matukura,kondo,hirata}@db.cs.kobe-u.ac.jp

Katsumi Tanaka
School of Informatics, Kyoto University
ktanaka@i.kyoto-u.ac.jp

Abstract We propose ways to discover semantic relationships among Web pages, andits applications in Web search, such as detailed-of, simplified-of, similar-topic-of, different-topic-of relationships.
In order to discover those semantic relationships, we propose two methods: One is based on 'topic structures' of Web pages. A topic structure of a Web page is computed by the combination of the term-appearance-density-distribution of each page and the term co-occurrence ratio for all the term-pairs in all retrieved pages. The other is based on the 'inclusion' relationships among feature vectors of Web pages. We describe both of their algorithms and their evaluations.

Keywords: Web, Web search, relevance feedback, topic structure

Introduction

Recently, by rapid development of Web technologies, the number of Web documents has become to be over several hundreds of millions. One of the most important problems is how to effectively search Web documents. By using several search engines, users usually input a few keywords in order to find Web documents relevant to their favorite topics. The problem of precision of search results has been drastically improved by advanced search engines called "Google"[google] and "Clever"[

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35658-7_21
R. Meersman et al. (eds.), *Semantic Issues in E-Commerce Systems*

J.Kleinberg, 1999], which take the link structures into consideration for increasing the precision of search results. These advanced search engines are still based on keyword-based information retrieval techniques, which are not sufficient to retrieve documents by their topics.

Several search engines also offer the "relevance feedback" functions to increase the precision ratio of search results. In most relevance feedback mechanisms of Web search engines, users choose some positive examples (his favorite answers) and/or negative examples among the search results, and then, the system automatically modifies the original query or the original scoring criterion and execute it again. We believe the weak point of current relevance feedback mechanisms is that users can state only *positive* or *negative*. That is, most of conventional relevance feedback mechanisms are just "conformity and nonconformity based". On the other hand, in the relevance feedback process, users may wish to rank example documents by more elaborated terms as follows:

1 *The theme and the content of the document are good. But, I wish to find more detailed documents with the same theme.*

2 *The theme and the content of the document are good. But, I also wish to find other documents with the same theme and different contents.*

3 *The theme and the content of the document are good. But, I also wish to find other document with the different theme and the same contents.*

Intuitively, our aim is to realize a *topic-based* relevance feedback retrieval. In the above, we assume that a topic of a Web page is captured by a set of pairs of *thematic-keyword set* and *content-keyword set.*

In this paper, for the above purposes, we propose new ways to discover semantic relationships among Web pages, and its applications in Web search. We introduce several semantic relationships among Web pages based on 'Web topic structures', such as detailed-of, simplified-of, similar-topic-of, different-topic-of relationships. In order to discover those semantic relationships, we propose two methods: One is based on topic structures of Web pages. A "topic structure" of a Web page is computed by the combination of the *term-appearance-density-distribution* of each page and the *term co-occurrence ratio* for all the term-pairs in all retrieved pages. The other is based on the 'inclusion' relationships among feature vectors of searched Web pages. We describe both of their algorithms and their evaluations.

1. Topic Structure Approach

1.1. Term-Appearance-Density-Distribution

Usually, a Web page describes more than one topic, and so, we need a function to discover multiple topics from a single Web page. In this section, we propose a way to discover several semantic relationships among Web pages based on the discovered topics of each Web page. As for related work, Hearst[Marti A.Hearst, 1994] proposes a way, called *text-tiling*. In this paper, we propose another method of dividing a Web page into topics based on both of the *term-appearance-density-distribution* proposed by Sadao Kurohashi, 1997 and the logical structure of Web pages.

The *term-appearance-density* is a value that is computed by the frequency of a specific term (word) within a certain scope of a document and the term's *position information*. Intuitively, the higher the frequency of a term is within a specified scope, the bigger the term-appearance-density value of the term is. In other words, a term with the high term-appearance density can be regarded as a domonating term within a specified scope. In order to compute a term-appearance-density of a term, we have to give some weight to the term using the *window function*, such as the *rectangular window function*, and the *triangle window function* etc. In this paper, we use *Hanning window function*[Sadao Kurohashi, 1997] $h_l(i)$, which is given below

$$h_l(i) = \frac{1}{2}(1 + \cos 2\pi \frac{i-l}{W}) \quad (|i-l| \leq \frac{W}{2}) \tag{1}$$

Here, W denotes the width of a window (range which gives a weight) and l denotes the main position of a window. Using the Hanning window function, $h_l(0)$ becomes to be 1 in a center of a window, and $h_l(i)$ becomes smaller for i such that i is far from the center.

The following describes a way to compute the term-appearance-density using the Hanning window function:

1 We regard a document (Web page) as a single long character string of length L. When a specified term appears from the lth character from the head of a document, the value of the function $a(l)$ is set to be 1.

$$a(l) = \begin{cases} 1 & \text{a term appears from the } l\text{th position in the document} \\ 0 & \text{otherwise} \end{cases}$$

2 The term-appearance-density $d(l)$ for the main position is computed as follows. It starts from $a(l) = 0$ which is the head of a

document and each position l is considered as the main position of Hanning window at order. Appearance-density $d(l)$ is defined as follows:

$$d(l) = \sum_{i=l-\frac{W}{2}}^{l+\frac{W}{2}} h_l(i) \cdot a(i) \tag{2}$$

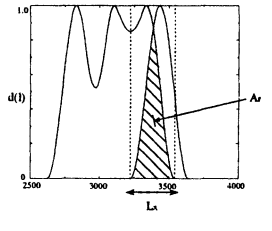
where $a(i) = 0$ when $i < 0$ or $i \geq L$ are satisfied.

3 We normalize the term-appearance-density-distribution for each term by its maximum value. Then we have the normalized term-appearance-density-distribution for each term, whose range is between 0 and 1. This term-appearance-density is called a *term-relative-appearance-density-distribution.*
The *term-relative-appearance-density-distribution* for each term represents the change of the importance of the term for each position.

Using the above method, we are able to compute the appearance-density values in all places of all the terms (words) contained in a Web page.

1.2. Degree of Association Between Keywords

The term-appearance-density-distribution expresses the change of the importance of a term in a Web page. Therefore, when both of the importance of two terms in a certain scope are high, it turns out that there exists a certain semantic relationship between them. Here, we describe a method to compute the degree of association between keywords by the term-appearance-density-distribution.



C

Figure 1. The degree of association based on density-distributions of two terms

1 As shown in Figure 1, for every pair of terms A and B, we compute the overlapping area (denoted by A_r) and the width of the overlapping interval (denoted by L_r) of term-relative-appearance-density-distribution graphs for A and B.

2 Let $\frac{A_r}{L_r}$ be the average of the overlapping areas for all the pairs of terms, which denotes the degree of association between keyword A and B.

The degree of association between terms is also computed by the *co-occurrence ratio* between terms. The co-occurrence ratio for a pair of terms means a probability under which the pair of terms appear in the same document for a given set of documents. For a given set of documents, let $P(W_a)$, $P(W_b)$, and $P(W_a, W_b)$ be the numbers of documents containing the term W_a, W_b, and both of W_a and W_b, respectively. The co-occurrence ratio of terms W_a and W_b for the given set of documents is defined as follows:

$$C(W_a, W_b) = \frac{P(W_a, W_b)}{P(W_a)} \frac{P(W_a, W_b)}{P(W_b)} \tag{3}$$

1.3. Extraction of Thematic and Content Keywords

By the term-relative-appearance-density-distribution, we can expect the "dominant area" and the "dominance degree" of a given term within a Web page. That is, for a term A, if the term-relative-appearance-density-distribution is high for a certain area within a Web page, the obtained area is the dominant area of the term A, and the term A is considered to be a related term of a topic. Also, in such the area, terms related to the topic will also appear with high frequencies. In order to find units of topics within a Web page and to divide the Web page into several topics, first, we compute dominant areas and dominance degrees for each term in the Web page by the term-relative-appearance-density-distribution. The following is a procedure to find units of topics in a given Web page and to divide the Web page into more than one topic.

We compute the term-relative-appearance-density-distribution for each term in a given Web page.

We make a summation of all the term-relative-appearance-density-distributions. The result of the summation denotes a distribution tendency of all the terms.

In the obtained distribution tendency curve, we find *hills*, whose relative height is greater than a specified threshold w_k. From the adjacent hills obtained in the curve, we select candidate *cut points* dividing a page into topics.

Based on the positions of the obtained candidate cut points, we find HTML tags that works as real cut points dividing the Web page. By the real cut points, we divide the Web pages into more than one contiguous areas, each of which is considered to represent a sible topic.

In each selected topic area obtained in the above method, there usually exist one or more dominant terms, say *thematic keywords*. The way to select the thematic keywords is as follows:

We divide a target Web page into one or more contiguous sub-areas, each of which represents a topic based on the obtained dividing points. From each sub-area, we extract all the terms except stop words.

For each obtained term, we compute the area size of the term-relative-appearance-density-distribution within its sub-area.

We select terms such that the the term's area size exceeds a predefined threshold X as thematic keywords of the corresponding topic. Threshold X is equal to the area size of the Hanning window used in the page.

In the beginning portion of each topic sub-area, some terms modified by certain HTML tag have high potentials to be the title of the topic. Therefore, we add those terms as thematic keywords.

Usually, a Web page may have one or more topic sub-areas. Each topic may be represented by one or more thematic keywords. Also, we allow that some thematic keywords appear in more than one topic sub-areas. That is,we allow the duplication of thematic keyword in several topics.

After finding thematic keywords for each topic, we need to find related words, called *content keywords*, that describe each thematic keyword within its topic sub-area. In order to find the content keywords for a given thematic keyword, we select words which are highly related to the thematic keyword. That is, we select the words with high relation of the term-appearance-density-distribution with regard to a thematic keyword and the words with high co-occurrence ratio with regard to a thematic keyword as content keywords. We also allow that some thematic keyword for a topic may become a content keyword for the same topic.

1.4. Discovery of Semantic Relationships between Topics

In order to discover semantic relationships between topics of Web pages, it is necessary to compare the difference of thematic keywords and content keywords of each topic. In this subsection, we propose a *topic graph* which represents the topic structure of a Web page.

A *topic graph* consists of a set of *thematic nodes* and a set of *content nodes*. A *thematic node* means a set of thematic keywords which corresponds to a single topic, and a *content node* means a set of content keywords corresponding to a thematic node. In a topic graph, a *thematic node* and its corresponding *content node* is connected by an edge (see Figure 2).

Formally, a topic graph G_d of a Web page d is defined as follows. A topic t is denoted by $t = (K_m, K_c)$, where K_m is a set of thematic keywords and K_c is a set of content keywords. When a Web page d consists of topics $t_1, t_2, ..., t_k$, it is represented by

$$d = \{t_1 = (K_{m1}, K_{c1}), t_2 = (K_{m2}, K_{c2}), ..., t_k = (K_{mk}, K_{ck})\}. \quad (4)$$

Then, a topic graph $G(d)$ is denoted by:

$$G(d) = (V, E). \quad (5)$$

Here,

$$V = \{K_{m1}, K_{m2}, ..., K_{mk}, K_{c1}, K_{c2}, ..., K_{ck}\}. \quad (6)$$

$$E = \{t_1 = (K_{m1}, K_{c1}), t_2 = (K_{m2}, K_{c2}), ..., t_k = (K_{tk}, K_{ck})\}. \quad (7)$$

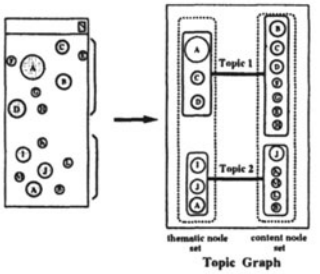


Figure 2. Topic Graph

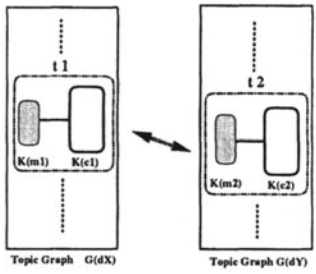


Figure 3. Comparison of topic t_1 and t_2

Table 1. Relationships between topic t_1 and t_2

	$K_{c1} \simeq K_{c2}$	$K_{c2} \in K_{c1}$	$K_{c2} \ni K_{c1}$	$K_{c2} \not\in K_{c1}$
$K_{m2} \simeq K_{m1}$	(1)	(2)	(3)	(4)
$K_{m2} \in K_{m1}$ ($K_{m2} \ni K_{m1}$)	(5)	(6) ((7))	(7) ((6))	(8)
$K_{m2} \not\in K_{m1}$	(9)	(10)	(10)	(11)

Now, we describe a way to discover several semantic relationships between topics (Web pages) by comparing the inclusion relationships between two corresponding sets of thematic keywords and/or between two corresponding sets of content keywords,

For example, let $t_1 = (K_{m1}, K_{c1})$ and $t_2 = (K_{m2}, K_{c2})$ be two topics extracted from Web pages d_X and d_Y, respectively. We compare the inclusion relationships between K_{m1} and K_{m2} as well as those between K_{c1} and K_{c2} (see Figure 3). The results of comparing two node sets A and B are classified into the following three relationships.

1 $B \simeq A$: The sets A and B are regarded as the almost same ones.

Assume that a set of common elements of A and B is denoted by K_{AB}. Also, we denote the rates of $|K_{AB}|$ to $|A|$ and to $|B|$ by w_A and w_B, respectively. When both w_A and w_B are greater than a predefined threshold w_1, this relationship is regarded as $B \simeq A$.

2 $B \Subset A$: The set B almost contains the set A.

When w_A is greater than w_1 and w_B is less than w_2, this relationship is regarded as $B \Subset A$.

3 $B \not\simeq A$: The set A is almost different from the set B.

When both of w_A and w_B are less than a predefined threshold w_3, this relationship is regarded as $B \not\simeq A$.

When we restrict our comparison of two topics to the above three relationships, we have the following eleven relationships (see Table1).

(1) Two topics are almost the same.

(2) Both topics describe the same theme, but that one topic explains about the theme more simply than the another topic.

(3) One topic explains about the theme in more details than the another topic.

(4) Two topics describe different aspects of the same theme.

(5) One topic overstates the theme compared with another one.

(6) One topic describes a part of the other one's theme and content. For example, this case will occur when one topic explains the whole history of a country and another topic explains a portion of its history.

(7) One's theme is narrower than the other one's theme although one's content is wider than the one's content. For example, one topic describes briefly the whole history of a country. The other one describes the whole history, but also describes more details about a part of it.

(8) One topic is more specialized than the other one. For example, the one topic describes a computer generally. The other one describes features of computers of a certain vendor.

(9) Each topic has the same meaning substantially, but belongs to the different theme. For example, this corresponds to the case when one topic is an introduction of a certain person as a president of a company, and the other one is also an introduction of the same person as a university professor.

(10) For example, this corresponds to the case when one's theme is a and the other one's theme is concerned with a certain part of the computer.

(11) Two topics are different.

1.5. Experiments and Evaluations

In order to test the precision of discovered semantic relationships among Web pages, we performed an experiment, in which we used Web pages obtained by the search engine goo[goo]. The specified keyword for query is `hotspring` and `use`. We calculated co-occurrence ratio for all pairs of terms in given Web pages. Next, we computed the term-relative-appearance-density-distributions for all the terms appearing in each page. Then, we divide 1000 pages into topic sub-areas and discover several semantic relationships between topics.

In this experiment, we evaluated 100 sampled pages for the case when $w_k = \{0, 0.1, 0.2\}$. We define the precision ratio as the ratio of number of correctly divided pages to 100 total pages.

Also, we define the recall ratio as the ratio of the number of correctly extracted topics to the number of correct topics that should be extracted. Table 2 show the evaluation results of the precision ratio and the recall ratio, respectively. We find that in the case of $w_k = 0.1$, both of the precision and the recall are best. The following pages are not sufficiently divided, and so, they are regarded as *bad* pages.

- Web pages containing many links to other pages which have no semantically relationships with each other.
- Web pages consisting of HTML tables as its major part.
- Web pages which contain many words which can't be analyzed by morphological analysis software, or which are numbers or proper nouns.

Based on the above evaluation result, we used the threshold ($w_k = 0.1$) in the next experiment.

We extracted thematic keywords and content keywords of topics. In order to test the usefulness of our topic extraction, we selected 100 topics as sample topics. And we evalated topics which were judged semantic relationships with sample topics (see Table 1). We show the precision

ratio in the case of (w_1, w_2, w_3). In this experiment, we excluded topics which weren't divided correctly. Also, we didn't evaluate the (11) relationship in Table 1.

The symbol () in the table represents the number of evaluated topics and $\emptyset$ represents that no topic was judged by our system.

- In the case of $(w_1, w_2, w_3) = (0.5, 0.3, 0.2)$ We show precisions in Table 3 (1). Total precision was 64.1%. There were some $\emptyset$ and a few topics in some relationships related to ($\simeq$). We set ($w_1 = 0.4$) to relax the definition of ($\simeq$), and experimented again, which is shown below.

- In the case of $(w_1, w_2, w_3) = (0.4, 0.3, 0.2)$ We show precisions in Table 3 (2). Total precision was 71.0%. The $\emptyset$ appeared only in (7). It can be thought that the low precision for (5) shows that it becomes to be approximately closed to the real precision ratio because of increased topics.

  To examine the effect of decreasing w_1, we set $(w_1, w_2, w_3) = (0.3, 0.2, 0.1)$ in the next experiment.

- In the case of $(w_1, w_2, w_3) = (0.3, 0.2, 0.1)$

  We show precisions in Table 3 (3). Total precision was 37.1%. We can observe that the usefulness of these thresholds is low.

Based on the above results, we can see that the precision ratio becomes high when $(w_1, w_2, w_3) = (0.4, 0.3, 0.2)$.

Between topics, which belong to the same community or are written by the same author, semantic relationships were extracted frequently. The relationship (7) was bad in any case. This seems to be due to the fact that originally there are few topics which have such a relationship.

Table 2. The result of dividing a page into topics

threshold	*precision*(%)	*recall*(%)
0	40.0	35.4
0.1	61.0	42.7
0.2	51.0	33.7

Table 3. The result 1 of extracting semantic relationships

	$K_{c1} \simeq K_{c2}$	$K_{c2} \in K_{c1}$	$K_{c2} \ni K_{c1}$	$K_{c2} \not\simeq K_{c1}$
$K_{m2} \simeq K_{m1}$	94.1(16/17)	$\emptyset$	40.0(2/5)	44.1
$K_{m2} \in K_{m1}$	66.7(2/3)	71.4	$\emptyset$	55.2
$K_{m2} \not\simeq K_{m1}$	$\emptyset$	$\emptyset$		

(1)

90.9(20/22)	100(2/2)	60.0(6/10)	61
50.0(7/14)	71.7	$\emptyset$	49.7
100(1/1)	66.7		

(2)

75.6	41.7	42.3	27.3
29.0	27.9	0(0/3)	30.0
$\emptyset$	50.0(8/16)		

(3)

precision (%)

2. DISCOVERING SEMANTIC RELATIONSHIPS USING VECTOR SPACE MODEL

Second, we propose a vector model approach for discovering semantic relationships of search results.

By using vector space model, for each user-specified searched page P, the Web search result (a collection of searched Web pages) is dynamically organized into four types of group: $Similar(P)$, $Different(P)$, $Detailed(P)$ and $Summarized(P)$.

- $Similar(P)$ denotes a group consisting of pages that contain the similar content as the user-specified page P.
- $Different(P)$ denotes a group consisting of pages that contain different content compared with P.
- $Detailed(P)$ denotes a group consisting of pages that contain the similar and more detailed content compared with P.
- $Summarized(P)$ denotes a group consisting of pages that contain the similar simpler content compared with P.

It is important how to decide these group elements. In this section we use content-based technique by keyword feature vectors. The keyword

feature vector of Web page P_i is denoted by $\mathbf{F}(P_i)$ and is defined as follows:

$$\mathbf{F}(P_i) = (W_1^i, \cdots, W_n^i) = \frac{1}{N_i}(f_1^i, \cdots, f_n^i)$$
Here, $f_j^i, j \in (1, \cdots, n)$is a normalized appearance frequency of word w_j in Web page P_i (8)

N_i denotes the total number of words in P_i.
W_j denotes the frequency of appearance of the corves ponding word.
We propose a way to compare keyword feature vectors between two pages. Each calculates the degree of similarity, difference, detail, summary relationships of page P_1 for page P_0, and it decides the type of P_1. Figure 4 shows a typical keyword feature vector.

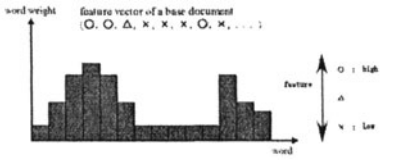


Figure 4. The keyword vector of the standard page

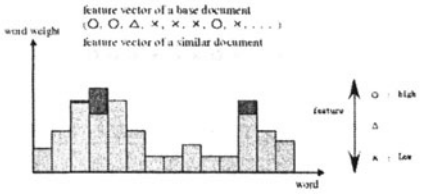


Figure 5. The keyword vector of the 'similar' page

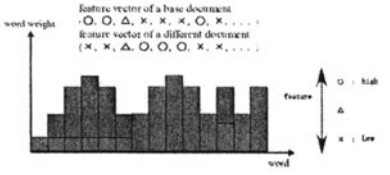


Figure 6. The keyword vector of the 'different' page

The classification procedure is described as follows.

1 For a user-specified page P_0 and the compared page P_1, the similarity degree λ_{sim} is obtained by taking inner product value $\mathbf{S}(P_0, P_1)$ of keyword vectors of P_0 and P_1. λ_{sim} is defined as: λ_{sim} is defined as:

$$\lambda_{sim} = \mathbf{S}(P_0, P_1) = \frac{\mathbf{F}(P_0) \cdot \mathbf{F}(P_1)}{||\mathbf{F}(P_0)|| \cdot ||\mathbf{F}(P_1)||} \tag{9}$$

If λ_{sim} is high, P_1 is regarded to be similar to P_0 (see Figure 5.)

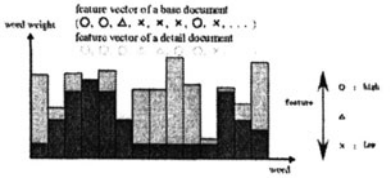


Figure 7. The keyword vector of the 'more detailed' page

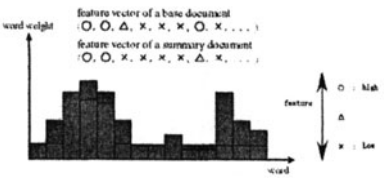


Figure 8. The keyword vector of the 'more summarized' page

2 As shown in Figure 6, intuitively if the histogram pattern of $\mathbf{F}(P_1)$ is *opposite* to that of $\mathbf{F}(P_0)$, then page P_1 is regarded to contain different topics. The degree of difference λ_{diff} is defined as:

$$\lambda_{diff} = 1 - \mathbf{S}(P_0, P_1) \tag{10}$$

If λ_{diff} is high, P_1 is regarded as different topic from P_0.

3 We assume that the 'more detailed' page has more additional information than the given page. Intuitively, we consider that almost all the word's weights of the 'more detailed' page vector are greater than those of a given pages' vector (see Figure 7,).
Let us denote the vector of a user-specified page by:

$$\mathbf{F}(P_0) = (w_1^0, w_2^0, \cdots, w_i^0, \cdots, w_n^0) \tag{11}$$

The keyword vector of a compared page is:

$$\mathbf{F}(P_1) = (w_1^1, w_2^1, \cdots, w_i^1, \cdots, w_n^1) \tag{12}$$

Without loss of generality, we assume the following:

- For each k in $\{1, \cdots, i\}$,
  $w_k^0 > \theta_0$ and $w_k^1 > \theta_0$,
  and $|w_k^1 - w_k^0| \leq \theta_1$
- For each k in $\{i+1, \cdots, j\}$,
  $w_k^1 - w_k^0 \leq 0$

In the above cases, we make w_k^0 and w_k^1 to be 0. That is, let $\mathbf{F}'(P_0)$ and $\mathbf{F}'(P_1)$ are defined as follows

$$\begin{aligned} \mathbf{F}'(P_0) &= (\underset{0}{0}, \cdots, \underset{i}{0}, \underset{i+1}{0}, \cdots, \underset{j}{0}, w_{j+1}^0, \cdots, w_n^0) \\ \mathbf{F}'(P_1) &= (\underset{0}{0}, \cdots, \underset{i}{0}, \underset{i+1}{0}, \cdots, \underset{j}{0}, w_{j+1}^1, \cdots, w_n^1) \end{aligned} \tag{13}$$

The degree of 'more detailed' relationship of P_1 compared with P_0 is denoted by λ_{detail}. It is expressed with an average of the difference of each value in vectors $\mathbf{F}'(P_0), \mathbf{F}'(P_1)$, and is defined by

$$\lambda_{detail} = \mathbf{D}(P_0, P_1) = \frac{1}{n} \sum_{k=j+1}^{n} (w_k^1 - w_k^0) \tag{14}$$

If λ_{detail} is high, P_1 is regarded to have 'more detailed' relationship with P_0.

4 The degree of the summarization relationship is intuitively opposite to the degree of the 'more detailed' relationship.

The degree of 'more summarized' relationship of P_1 compared with P_0 is denoted by λ_{summ} , and is defined by

$$\lambda_{summ} = \mathbf{D}(P_1, P_0) = \frac{1}{n} \sum_{k=j+1}^{n} (w_k^0 - w_k^1) \tag{15}$$

If λ_{summ} is high, P_1 is regarded to have 'more summarized' relationship with P_0.(See Figure 8)

We experimented in order to test the usefulness of the above degrees of relationships. Before we start the experiment, we make the query for the search engine, and for each document we examined to which group it belongs by hand.
We used search result data from 100 to 500 pages, and used 5 keywords. The top 10 ranking list is returned as answer of the system (see Table 9). Figure 10 shows precision and recall in 500 pages of searching result.

Table 9 shows the ratio of how many the correct answer which investigated a thing called a similar relation and a detailed relation for the inside of the solution prepared beforehand is contained. Moreover, when using as 500 pages of retrieval pages at 10, the ratio of conformity when adopting 10 affairs, 20 affairs, 50 affairs, and 100 affairs as solution to and a recall ratio are shown.

3. CONCLUDING REMARKS

In this paper, we proposed two methods to discover several semantic relationships among Web pages, which will be useful to realize more elaborated relevance feedback. The first method is based on topic structures of Web pages. A "topic structure" of a Web page is computed by the combination of the *term-appearance-density-distribution* of each page

Number of search results	similarity	detail	summary
100	90%	73%	30%
200	82%	70%	30%
300	82%	59%	21%
500	73%	45%	17%

Figure 9. Result of the experiment (precision %) difference is ignored

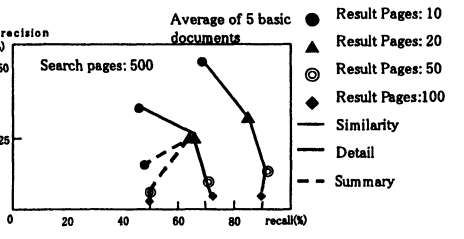


Figure 10. Precision and recall

and the *term co-occurrence ratio* for all the term-pairs in all retrieved pages. The second method is based on the 'inclusion' relationships between feature vectors of searched Web pages. We describe both of their algorithms and their evaluations.

As future works, first, we need to incorporate the proposed discovery mechanism as a relevance feedback system of search engines. Second, we will need to refine the definition of the topic.

Acknowledgements

This research is partly supported by the Research for the Future Program of Japan Society fro the Promotion of Science under the project: Researches on Advanced Multimedia Content processing, and the grant of Scientific Research (12680416) from Ministry of Education, Science , Sports and Culture of Japan.

References

goo.
http://www.goo.ne.jp/.

google.
http://www.google.com/.

J.Kleinberg (1999).
Authoritative sources in a hyperlinked environment.
the Journal of the ACM.

Marti A.Hearst, Christian Plunt. (1994).
Multi-paragraph segmentation of expository text. *ACL'94.*

Sadao Kurohashi, Nobuyuki Shiraki, Makoto Nagao. (1997).
A method for detecting important descriptions of a word based on its density distribution in text.
Transcations of Information Processing Society of Japan.