Structured Index System at NTCIR Workshop 2: Information Retrieval Methods Using Ordered Co-occurrence of Words and their Dependency Relationships

Atsushi MATSUMURA† Atsuhiro TAKASU Jun ADACHI
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan
{am, takasu, adachi}@nii.ac.jp

Abstract

We propose two Japanese-language information retrieval methods that enhance retrieval effectiveness by using relationships between words. The first method uses dependency relationships between words in a sentence, while the second method uses proximity relationships, in particular the ordered co-occurrence information of words in a sentence as an approximation to the dependency relationships between them. We construct these two methods on the Structured Index, which represents dependency relationships between words in a sentence as a set of binary trees. Structured Index is created by morphological analysis, dependency analysis, and compound noun analysis. We show the result of retrieval experiments using NTCIR–2, and discuss the effect of using relationships between words on Japanese information retrieval.

Keywords: *compound noun analysis, co-occurrence, dependency relationships, information retrieval, morphological analysis, natural language processing, NTCIR, phrases, proximity operation, Structured Index.*

1 Introduction

Because a large amount of electronic documents has become accessible to users directly through the Internet, it has become more important for users to retrieve the information they want efficiently and simply by phrasing their information needs in natural language. The Boolean model, which is a simple retrieval model based on set theory and Boolean algebra, does not meet these requirements, because it requires users to write complex logical expressions for query representation and presents the search output in a disordered manner. Although there are some search methods that arrange the output using a vector space model[3], there are obvious limitations in retrieval effectiveness. One reason for this is that in such retrieval systems, only query words and their statistical characteristics, such as Term Frequency and the Inverted Document Frequency (TF–IDF), are used and the relationships between the query words have been lost.

A great deal of work has been carried out on constructing information retrieval (IR) systems using relationships between words. Farradane proposed Relational Indexing and defined nine categories of relations, which were based on an analysis of thought processes, as investigated in the psychology of thinking[1, 2]. However, the assignment of relations was done manually and there was no significant improvement in retrieval effectiveness. Lu used lexical-semantic relations to connect words and build a structured representation of documents and queries[9]. However, relations between words were selected manually and both the size of the test database and the number of queries were small. On the other hand, Myaeng et al. have been developing a conceptual IR system that converts a large volume of natural language text into Conceptual Graph representation[14]. In this project, natural language processing (NLP) techniques were the main focus of evaluation. The IR system is still at the initial stage of development.

On the other hand, to avoid the complex and high-cost task of NLP, some IR methods use either statistical phrases, which were derived using techniques other than NLP, or proximity relationships between words as an approximation to syntactic or semantic relationships between words. Mitra et al. defined a phrase to be any pair of non-function words that appear in at least 25 documents of the TREC–1[1] collection[12]. However, they showed that the use of phrases in IR improved performance by only 1%. These experiments were repeated in a separate study by Smeaton and Kelledy[15].

†Presently with University of Library Information Science
1-2 Kasuga, Tsukuba-shi, Ibaraki 305-8550, Japan
matsumur@ulis.ac.jp

[1]Text REtrieval Conference.
URL:http://trec.nist.gov

Nonetheless, while IR for English text is at least being actively investigated, studies with Japanese text are tardy and as yet inconclusive. Hyoudo et al. compared proximity operations and dependency operations in Japanese text retrieval[4]. However, because evaluation was only from the viewpoint of whether correct dependency relationships were included in the retrieved documents, the effect on the general IR task was not clear. Hyoudo et al. also examined three other proximity operations: a phrase in the same document, sentence and clause[5]. They compared these methods and the method using dependency operations on the Japanese IR test collection IREX[2]. However, the effects of the methods were not clearly analysed.

From this perspective, we think it is necessary to conduct detailed analysis of IR methods that use relationships between words, especially for Japanese text. In our previous research, we proposed an IR method using dependency relationships between words and its approximation, namely an IR method using ordered co-occurrence information of words in a sentence. At NTCIR workshop 1, performance of our method was quite low and the difference between our method and the TF–IDF method was also small[10]. We analysed the result of retrieval experiments using NTCIR-1 and discovered the effective scoring method using relationships between words on Japanese IR[11].

In this paper, we first describe our two methods. Next, we show the result of the official and unofficial runs using NTCIR–2 (Preliminary Version)[3], and discuss the effect of using relationships between words on Japanese IR.

2 Overview of the Structured Indexing method

We call the method using dependency relationships between words *ST*. To utilize dependency relationships between words in *ST*, we propose a *Structured Index* represented by a set of binary trees that show dependency relationships between words. Figure 1 shows an example of a *Structured Index* for the sentence '情報検索における自然言語処理の効果' (effect of natural language processing on information retrieval).

In our method, words are classified into two groups, namely *concept words* and *relation words*. Each *concept word* represents a concept and is placed on a leaf node in the *Structured Index*. Each *relation word* associates two *concept words*. *Relation words* are also classified into *categories* according to their semantic

[2]Information Retrieval and Extraction Exercise.
URL:http://cs.nyu.edu/cs/projects/proteus/irex/

[3]NII Test Collection 2 (Preliminary Version) constructed by NTCIR(NII-NACSIS Test Collection for IR Systems) Project.
URL: http://research.nii.ac.jp/ntcir/index-en.html

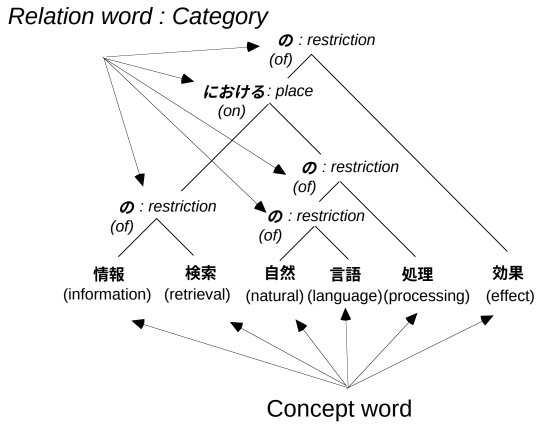

Figure 1. Sample of a *Structured Index*.

similarity. A *relation word* and its category are placed on an internal node in the *Structured Index*.

Using the *Structured Index* method, we can retrieve a document by using dependency relationships between words. We can also retrieve compound nouns by using their meanings represented by dependency relationships between their constituent *concept words*.

However, the *Structured Index* method is expensive because the index is quite large and the retrieval process becomes complex. We therefore propose another solution that uses a proximity relationship, which is defined as using ordered co-occurrence information of two words within a sentence. Because the cost of indexing and retrieval is reduced, this method would be more practical than the method using dependency relationships, provided that there is little difference in retrieval effectiveness between the two methods. Also, because this method is free from the problem of improving the accuracy of dependency analysis, the contribution of the method to retrieval effectiveness can be clearly analysed. We call this method *CO*.

3 Indexing

Because the *CO* method uses part of the information of the *Structure Index* that is made for the *ST* method, we describe the indexing method of *ST* in this section.

3.1 Morphological analysis

To determine dependency relationships between words, we must divide a sentence into *concept words* and *relation words*. In our definition, *concept words* include nouns, adjectives, adverbs and constituents of compound nouns. *Relation words* include postpositional particles, auxiliary verbs, verbs and their combinations.

Table 1. *Categories* of *relation words* and typical elements.

Category name	Typical elements
restriction	の (of), な (of), された (-ed), される (-ed)
place	における (on, for), での (in, on), 上の (on), から見た (in terms of)
method	による (by), を用いた (using), に基づく (based on), を利用した (using)
and	と (and), および (and), ならびに (and), も (too)
purpose	のための (for), を目指した (for), を指向した (oriented)
content	に関する (about), についての (about)
destination	への (to, for), 向きの (for)
source	からの (from), から (from)
consideration	を考慮した (considering), に着目した (from –'s viewpoint)
subject	に対する (of, on, for), を対象とした (for)
possession	を持つ (with, of, using), を有する (with, based on), を持った (with)
sharing	間の (between), で共有された (sharing with), 間での (between)
apposition	としての (as)
support	を支援する (supporting), をサポートした (supporting)
nominative	が[a], は [a]
adaptation	に対応した (for), に適した (suitable for), に応じた (according to)
possibility	可能な (-able), を可能とする (capable), が可能な (capable of)
or	や (or)
other	で表現された (expressed in), よりも優れた (which is superior to)

[a]There is no English translation equivalent to this Japanese morpheme

We employed ChaSen 1.51[4] as the Japanese morphological analyser. Morphemes are identified as *concept words* and *relation words* by using their parts of speech and a database of *relation words* which was constructed manually from the 3666 titles of scientific and technical documents. We also define 18 categories into which *relation words* can be classified according to their semantic similarity. We define an *other* category for *relation words* that cannot be classified in the above 18 categories. Table 1 shows the 18 categories and typical elements.

3.2 Dependency analysis

To define the dependency relationships between *concept words*, we used the order of *relation words* in a sentence, or *title template*. For example, the sentence '情報検索における自然言語処理の効果' (effect of natural language processing on information retrieval) belongs to the *title template* 'A における B の C' (C of B on A) [5], where A, B and C are *concept words* or their combinations (compound nouns). We manually assigned the dependency relationships between words to any *title template* that had two or three *relation words* and appeared more than three times in the 3666 titles that were used to make the database of *relation words*. The comparative table of *title templates* and dependency patterns contains 105 *title templates* (62 contain two *relation words* and 43 contain three *relation words*).

There were two relationship types for titles of at least two *relation words*. For the first type, we could assign only one dependency relationship, while for the second type, more than two dependency relationships were possible. For the latter type, we determined the dependency pattern according to the existence of a *general word* at the end of the sentence. A *general word* is a less important word, such as '研究' (study) or '効果' (effect), which does not have a dependency relationship with a particular word in the sentence, but with the whole sentence that precedes it. We defined 53 words as *general words*, including the above, '提案' (proposal) and '実現' (implementation).

If the dependency pattern was not identified by a *title template*, we used an *extended title template* in which a relation word is replaced with its category name. We defined 73 *extended title templates* (40 contain two *relation words* and 33 contain three *relation words*).

When the dependency pattern is not determined even by *extended title templates*, we divide the sentence into small parts using several heuristics, then assign a dependency pattern to each part using *title templates* or *extended title templates*. This method is important for maintaining the effectiveness of the *ST* method, because the dependency pattern given by this

[4]Japanese Morphological Analyzer 'ChaSen'
URL: http://cactus.aist-nara.ac.jp/lab/nlt/chasen.html (in Japanese)

[5]The syntactic arrangement of Japanese is often different from that of English.

method is correct locally in most cases, even if the total dependency pattern is incorrect.

3.3 Compound noun analysis

A compound noun is translated into a sentence by supplementing it with suitable *relation words* between its constituent words [13, 7].

While compound nouns are divided into their constituent *concept words* in morphological analysis, the major problem with this method is in determining which *relation words* can be inserted between *concept words*. We add the *relation word* 'の' (of) as a general principle, because of our statistical investigation concerning co-occurrences of *concept words* and *relation words* in compound nouns[6].

For the dependency analysis of compound nouns, we propose the use of *word bigram statistics*, which are statistics of the frequency with which two *concept words* in compound nouns are juxtaposed. The bigram value can be considered to represent the strength of the relationship between the two *concept words* in compound nouns, if enough samples are used. In our system, we used data from 814 bigrams, which appear more than 10 times in 60507 bigrams collected from the 13615 titles.

The whole process is performed automatically. All the sentences are given their dependency pattern and organized in the form of a *Structured Index*.

4 Retrieving and scoring method

4.1 Query form

Queries are in the form of pseudo-natural language, such as article titles. Therefore, they are structured in the form of a binary tree by the same process as indexing. Retrieval in *ST* consists of matching the binary tree of a query and a set of binary trees derived from documents. On the other hand, *CO* only uses the ordered co-occurrence information of each two *concept words* in those binary trees.

4.2 Total scores of documents

First, we calculate the score of each document element to reflect its difference in the scoring of documents. Next, the score of each element is divided into two parts: the score for the words in that element and the score for the relationships between the words. As a result, we can verify the effect of the relationships between words in retrieval performance by comparing our method with the keyword-based method. The total score of document d is shown in Eq. (1).

$$S_d = \sum_b \left(xw_b \times SW_b + xr_b \times SR_b\right) \qquad (1)$$

where the variables in the equation are

b : the document element (title, abstract, etc.) ,
SW_b : the score on words given to the document element b ,
SR_b : the score on relationships between words given to the document element b ,
xw_b : the weight of SW_b , and
xr_b : the weight of SR_b .

4.3 Scoring on words

When a *concept word* of the query appears in a document element, our system scores the element by the modified TF–IDF weighting defined as Eq. (2).

$$SW_b = \sum_{j=1}^{n} tfidf(C_j)\delta_j \qquad (2)$$

where $tfidf(C_j)$ is the modified TF–IDF scoring function, which is in the form of

$$tfidf(C_j) = \log\left(tf(C_j)+1\right)\log\left(\frac{N_{all}}{df(C_j)}\right) \qquad (3)$$

where the variables in the equations are

C_j : j-th *concept word* in the query ,
n : the total number of *concept words* in the query ,
$$\delta_j = \begin{cases} 1 & \text{when the \emph{concept word} } C_j \text{ appears in the document element } b \\ 0 & \text{otherwise ,} \end{cases}$$
$tf(C_j)$: the frequency of the *concept word* C_j in the document element ,
$df(C_j)$: the number of documents that contain the *concept word* C_j , and
N_{all} : the total number of the documents .

4.4 Scoring on relationships between *concept words*

A sentence in a document includes several dependency relationships, each of which is represented by a triplet of two *concept words* and a *relation word*. This triplet is the smallest scoring unit of *ST*. In our preliminary experiment, the retrieval effectiveness was quite small when our system used only exact dependency relationships in scoring documents. To gain more effectiveness, we assign relationships to pairs of *concept words* that do not have dependency relationships. Considering such pseudo-dependency relationships, we can define triplets for all pairs of *concept words* in a sentence. Using this definition, *ST* is equivalent to *CO* with dependency relationships and pseudo-dependency relationships. On the other hand, the scoring unit of *CO* is an ordered pair of *concept words* in a sentence, without the *relation word*.

4.4.1 Scoring in *ST*

In the *ST* method, the score on each triplet that is matched by a triplet in the query is calculated according to the semantic similarity measure. For this, we define two matching criteria.

First, we consider the level of matching between two triplets. Even if two triplets have the same *concept words*, their semantics are often different because of differences in their dependency relations. We therefore evaluate the similarity between the two triplets according to the following three levels.

Exact Match : The two *relation words* are the same.

Category Match : The two *relation words* are different but their categories are the same.

Wild Match : The two *relation words* and their categories are different.

We change the scoring factor of *ST* to reflect the above three levels of matching.

Second, we use the notion of importance of the triplet in the document set that is the target of retrieval. Because the importance of a triplet grows according to the importance of the two *concept words* in it, we adopt the product of their IDF scores as the importance of the triplet. In addition, considering the noise that might be caused by *general words*, we define the importance of a triplet as zero if any *general words* are included in it.

Using the above two matching criteria, the score of a triplet $Sd(TR)$ that has the left *concept word* C_l and the right C_r is shown in Eq. (4).

$$Sd(TR) = LD(TR)ID(C_l, C_r) \quad (4)$$

where the variables in the equation are

$$LD(TR) \ : \ \text{the weight for the matching level of the triplet } TR$$

$$= \begin{cases} we & \text{for Exact Match} \\ wc & \text{for Category Match} \\ ww & \text{for Wild Match} \end{cases},$$

$$ID(C_l, C_r) \ : \ \text{the importance of the triplet } TR \text{ in the document set}$$

$$= idf(C_l)idf(C_r)gw(C_l)gw(C_r) \ , \quad (5)$$

$$idf(C) = \log\left(\frac{N_{all}}{df(C)}\right) \ , \ \text{and}$$

$$gw(C) = \begin{cases} 0 & \text{if the concept word } C \text{ is a general word} \\ 1 & \text{otherwise} \end{cases}.$$

4.4.2 Scoring in *CO*

In the *CO* method, we calculate a score based on the ordered co-occurrence of two *concept words* in a sentence. We adopt the product of the IDF scores of two *concept words* as the importance measure of their co-occurrence. Consequently, the score in *CO* is equivalent to the function *ID*, which is defined as the importance of a triplet in Eq. (5).

4.4.3 Total score on relationships in *ST* and *CO*

We next define the similarity score between a query and a document element with regard to relationships between words, using the scores of all matched triplets or pairs in the document element. From our previous research, we obtained a method of scoring document elements that is effective for retrieval performance [10]. According to the method, the score on relationships of a document element b is calculated by the following Eq. (6).

$$SR_b = \sum_{j=1}^{m} \max\{Sd(TR) : TR \in Rel_j\} \quad (6)$$

where the variables in the equation are

Rel_j : j-th triplet or pair in the query , and
m : the number of triplets or pairs in the query .

The function max chooses the maximum score out of all scores of triplets or pairs matched in the document element for each triplet or pair in the query. This method prevents the repetition of scoring by a triplet or pair in a query and avoids the dropping of important triplets or pairs in scoring document elements.

5 Experiments and evaluation

We submitted ten official runs for the *J-J task*: from STIX1 to STIX10. In this section, we show the result of these ten official runs and several unofficial runs that we have refined after relevance judgements.

5.1 Conditions of experiments

Document collections used in *J-J task* are ‘ntc1-j1.mod’ from NTCIR–1 and ‘ntc2-j0g’ and ‘ntc2-j0k’ from NTCIR–2 (Preliminary version), and the search topic set is ‘topic-j101-150’. To apply our methods to the *J-J task* of NTCIR–2, we chose following three elements from ‘ntc1-j1.mod’ and ‘ntc2-j0g’: ‘ TITL TYPE=”kanji” ’ as the title, ‘ ABST TYPE=”kanji” ’ as the abstract, and ‘ KYWD TYPE=”kanji” ’ as the keyword. We also chose three elements from ‘ntc2-j0k’: ‘ PJNM TYPE=”kanji” ’ as the title, ‘ ABST TYPE=”kanji” ’ as the abstract, and ‘ KYWD TYPE=”kanji” ’ as the keyword. We used the ‘DESCRIPTION’ field of search topics as queries. Consequently, we must optimize six parameters in Eq. (1):

Table 2. The results of the official runs.

run ID	method	(xw, ww)	11-pt. ave. level 1 (S&A)	11-pt. ave. level 2 (S&A&B)
STIX1	*ST*	$(0.8, 0.6)$	**0.2309**	0.2073
STIX2	*CO*	$(0.8, 1.0)$	**0.2277**	**0.2042**
STIX3	*ST*	$(0.8, 0.7)$	0.2308	0.2077
STIX4	*ST*	$(0.8, 0.8)$	0.2306	0.2071
STIX5	*CO*	$(0.7, 1.0)$	0.2246	0.2027
STIX6	*ST*	$(0.7, 0.6)$	0.2301	**0.2097**
STIX7	*ST*	$(0.7, 0.7)$	**0.2309**	0.2082
STIX8	*ST*	$(0.7, 0.8)$	0.2301	0.2075
STIX9	*CO*	$(0.9, 1.0)$	0.2186	0.1959
STIX10	TF–IDF	$(1.0, 1.0)$	0.1770	0.1553

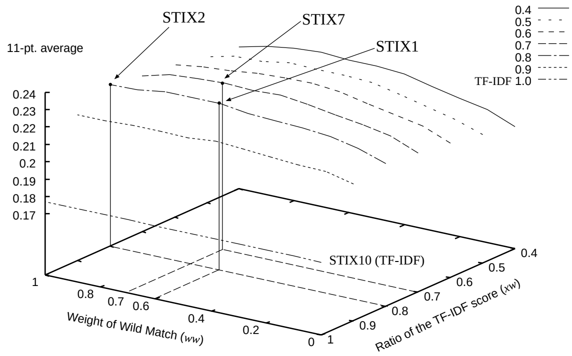

Figure 2. 11-point average precision in a 2D parametric space.

xw_t, xw_a, xw_k, xr_t, xr_a, xr_k, where subscripts t, a and k mean 'title', 'abstract' and 'keyword', respectively, and three parameters in Eq. (4): *we*, *wc*, *ww*.

In NTCIR–2, relevance judgements were done in four grade: Highly Relevant (rank S), Relevant (rank A), Partial Relevant (rank B), and Non-Relevant (rank C). We used two different levels of judgements: relevance level 1 (rank S and A are rated "relevant") and relevance level 2 (rank S, A and B are rated "relevant").

Table 3. Results of the optimization on the relevance level 1.

method	11-pt. ave.(gain)	(xw, ww)	run ID
TF–IDF	0.1770 (—)	$(1.0, 1.0)$	STIX10
ST	0.2309 (30.5%)	$(0.8, 0.6)$	STIX1
ST	0.2309 (30.5%)	$(0.7, 0.7)$	STIX7
ST	0.2309 (30.5%)	$(0.7, 0.5)$	unofficial
CO	0.2277 (28.6%)	$(0.8, 1.0)$	STIX2

5.2 Results of the official runs

In our official runs, we used abstracts as the target of retrieval, that is, the weight of the elements in Eq. (1) were 0 except for xw_a and xr_a. Here, we write xw and xr, omitting the subscripts, define $xr = 1 - xw$, and use $0 \leq xw \leq 1$ as a parameter for scoring documents. Our method, then, is equivalent to the TF–IDF method when $xw = 1$. We use this method as the baseline of our system. Next, we define the weights of *Exact Match* (*we*) and *Category Match* (*wc*) to be equal to 1 and use the weight of *Wild Match* ($0 \leq ww \leq 1$) as another parameter characterizing the system. When $ww = 1$, *ST* is equivalent to *CO*. Consequently, we can compare these three methods by changing only these two parameters.

Table 2 shows the 11-point average precisions of ten official runs that are calculated both by using relevance level 1 and 2. The method of STIX2, STIX5 and STIX9 is *CO*, that of STIX10 is TF–IDF (baseline) and that of the rest is *ST*. The parameters of these runs were decided by considering the results of experiments using NTCIR–1[11]. The maximum 11-point average precision using relevance level 1 was 0.2309 for STIX1 and STIX7 both of which used *ST* method. The maximum by using *CO* method was 0.2277 for STIX2. On the other hand, the best performance of the relevance level 2 was achieved by STIX6 (0.2097). STIX2 was also the best *CO* method in this case. These methods achieved about 30% superiority over the baseline STIX10.

Next we tuned xw and ww for *ST* and xw for *CO* to gain the maximum 11-point average precision using the relevance level 1 by investigating a two-dimensional parameter space (xw versus ww). Figure 2 shows the 11-point average precision versus the two parameters xw and ww. Each curve is for a fixed ratio of TF–IDF, which is given in the upper right corner of the figure. When $xw = 1$ (dash-dot-dot line in Figure 2), *ST* is equivalent to the TF–IDF method. Because of the optimization of two parameters for *ST*, the maximum 11-point average precision was 0.2309 for parameters $(xw, ww) = (0.8, 0.6)$, $(0.7, 0.7)$ and $(0.7, 0.5)$. We also optimized the parameter xw of *CO* ($ww = 1$). The maximum 11-point average precision of *CO* was 0.2277 at $xw = 0.8$. These results are summarized in Table 3. The highest precision for *CO* was given by the parameter $xw = 0.8$ that was optimized by using NTCIR–1. On the other hand, one of the highest average precision for *ST* was given by the parameters that were also optimized by using NTCIR–1.

Table 4. Optimized 11-point average precisions with the new baseline.

method	11-pt. ave.	gain		(xr_t, xr_a, xr_k, ww)
old baseline	0.1770	(—)	(—)	—
new baseline	0.2187	(23.6%)	(—)	—
ST	0.2558	(44.5%)	(17.0%)	$(0.04, 0.04, 0.0, 0.6)$
CO	0.2537	(43.3%)	(16.0%)	$(0.04, 0.04, 0.0, 1.0)$

Table 5. Statistical significance tests.

method pair	t-test	sign test	Wilcoxon test
ST vs. new baseline	$t_0(48) = 3.047, p < 0.01$	$Z_0 = 3.280, p < 0.01$	$Z_0 = 3.578, p < 0.01$
CO vs. new baseline	$t_0(48) = 2.980, p < 0.01$	$Z_0 = 3.578, p < 0.01$	$Z_0 = 3.640, p < 0.01$
ST vs. *CO*	$t_0(48) = 0.934, p > 0.05$	$Z_0 = 0.894, p > 0.05$	$Z_0 = 0.079, p > 0.05$

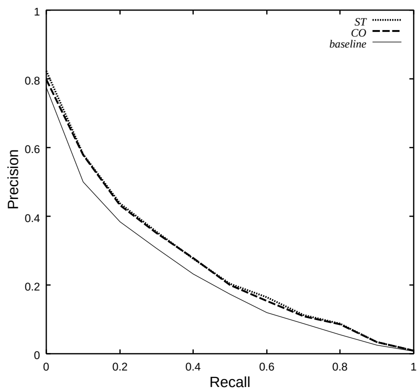

Figure 3. Recall versus precision figures for *ST* and *CO* with new baseline.

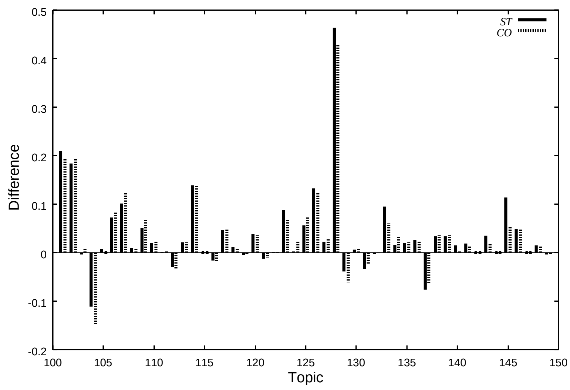

Figure 4. Difference from new baseline in average precision per topic.

5.3 Improving the baseline

Since our scoring method of the baseline was primitive, and its 11-point average precision was too low, we refined it and tuned *ST* and *CO* using the new baseline. As the scoring method of the new baseline, we employed the scoring method of Kanazawa et al.[8] that was developed using NTCIR–1. The scoring function modified to be used by *ST* and *CO* is shown in Eq. (7).

$$NSB_d = \sum_{j=1}^{n} \left(\frac{1}{\pi} \arctan \left(tf_d(C_j) \right) + 0.5 \right) \times \frac{2}{\pi} \arctan \left(\frac{N_{all}}{df(C_j)} \right) \quad (7)$$

where the variables in the equation are

$tf_d(C_j)$: the frequency of the *concept word* C_j in the document d ,

$df(C_j)$: same as in Eq. (3) .

Because this scoring function does not distinguish the difference of document elements, the number of parameters to be tuned were reduced from seven to four: xr_t, xr_a, xr_k and ww. Table 4 shows the results of optimization of these four parameters. We achieved a great improvement on retrieval effectiveness with the new baseline. The 11-point average precision of the new baseline was 0.2187, which is 23.6% higher than that of the old baseline (0.1770). Similarly, those of *ST* and *CO* were 0.2558 and 0.2537, respectively, which are 44.5% and 43.3% higher than that of the old baseline, and 17.0% and 16.0% higher than that of the new baseline.

Figure 3 illustrates the recall versus precision figures for the new baseline, optimized *ST* and optimized *CO*. This figure shows that *ST* and *CO* improve the precision at almost all recall levels, and that the difference between *ST* and *CO* is small.

Figure 4 shows the difference from the new baseline in average precision per topic. The result varies depending on the topics, but extensive improvement of the average precision is achieved on many topics by our two methods. For some topics, the dropping of average precision of *ST* is smaller than that of *CO*.

Table 5 shows statistical significance tests for our two methods and the new baseline. The difference between *ST* and the new baseline and between *CO* and the new baseline are both significant. However, the difference between *ST* and *CO* is not statistically significant.

6 Conclusion

We have proposed two IR methods using relationships between words. One method uses dependency relationships between words (*ST*) and the other is an approximation to *ST*, by using the ordered co-occurrence information about words (*CO*). We performed experimental evaluations on our two methods comparing them to the TF–IDF based baseline using the Japanese test collection for IR systems NTCIR–2 (Preliminary Version).

The results showed *ST* and *CO* outperformed the baseline based on the primitive TF–IDF scoring method. Also great performance improvements was achieved by improving the scoring function of the TF–IDF based baseline. Although the contribution of *ST* and of *CO* were reduced by the improvement of the baseline, they are clearly superior to the new baseline.

However, the difference between *ST* and *CO* is not statistically significant. The effect of *ST* deserves more discussion. In our preliminary experiments, we found that the accuracy of extracting dependency relationships was critical to the retrieval effectiveness of *ST*. Further improvements in extracting dependency relationships may improve the superiority of *ST* over *CO* and the baseline.

We must also analyse the retrieval output of each topic and make the system more general. We must optimize the average precision for each topic and clarify the relation between relationships of words and retrieval performance.

Acknowledgements

This research is supported by the 'Research for the Future Program', JSPS-RFTF96P00602 of the Japan Society for the Promotion of Science.

We participate in NTCIR Workshop 2, and used NII Test Collection 2 (Preliminary Version). This collection was constructed by the NII using a part of the "NACSIS Academic Conference Papers Database" provided by academic societies [6], and "NACSIS Grant-in-Aid Scientific Research Database".

[6] http://research.nii.ac.jp/ntcir/acknowledge/thanks1-en.html

References

[1] J. Farradane. Relational indexing. part I. *Journal of Information Science*, 1(5):267–276, 1990.

[2] J. Farradane. Relational indexing. part II. *Journal of Information Science*, 1(5):313–324, 1990.

[3] D. Harman. Ranking algorithms. In *Information Retrieval*, chapter 14. Prentice Hall, 1992.

[4] Y. Hyoudo, K. Niimi, and T. Ikeda. Comparison between proximity operation and dependency operation in Japanese full-text retrieval. In *Proceedings of ACM SIGIR'98*, pages 341–342, August 24–28 1998.

[5] Y. Hyoudo, K. Niimi, and T. Ikeda. The report of evaluation on IREX. In *Proceedings of the IREX Workshop*, pages 41–44, September 1–3 1999. (in Japanese).

[6] K. Ikeda and J. Adachi. A method of document retrieval using dependency relationship between words. In *The 54rd Annual Convention IPSJ*, 1997. 4K–2 (in Japanese).

[7] M. Ishizaki. Relationships between words in Japanese noun compounds. In *The 37th Annual Convention IPSJ*, 1988. 3C–2 (in Japanese).

[8] T. Kanazawa, A. Takasu, and J. Adachi. A relevance-based superimposition model for effective information retrieval. *IEICE Transactions on Information and Systems*, E83-D(12):2152–2160, 2000.

[9] X. Lu. Document retrieval: A structural approach. *Information Processing & Management*, 26(2):209–218, 1990.

[10] A. Matsumura, A. Takasu, and J. Adachi. Structured Index system at NTCIR1 : Information retrieval using dependency relationship between words. In *Proceedings of the First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition (NTCIR Workshop 1)*, pages 117–122, August 30–September 1 1999.

[11] A. Matsumura, A. Takasu, and J. Adachi. The effect of information retrieval method using dependency relationship between words. In *Proceedings of the RIAO'2000 Conference*, pages 1043–1058, April 12–14 2000.

[12] M. Mitra, C. Buckley, A. Singhal, and C. Cardie. An analysis of statistical and syntactic phrases. In *Proceedings of the RIAO'97 Conference*, pages 200–214, June 1997.

[13] M. Miyazaki. Automatic segmentation method for compound words using semantic dependent relationships between words. *Transactions of IPSJ*, 25(6):970–979, 1984.

[14] S. H. Myaeng, C. Khoo, and M. Li. Linguistic processing of text for a large-scale conceptual information retrieval system. In *Lecture Notes in Artificial Intelligence 835, Proceedings of 2nd International Conference on Conceptual Structures*, pages 69–83. Springer–Verlag, 1994.

[15] A. F. Smeaton and F. Kelledy. User-chosen phrases in interactive query formulation for information retrieval. In *Proceedings of the 20th BCS-IRSG Colloquium*, pages 200–214, April 1998.