Structured Index System at NTCIR1: Information Retrieval using Dependency Relationship between Words

Atsushi MATSUMURA, Atsuhiro TAKASU, Jun ADACHI
Research & Developmnt Department
National Center for Science Information Systems
{atsushi, takasu, adachi}@rd.nacsis.ac.jp

Abstract

It is difficult to improve retrieval effectiveness using only keyword-based retrieval, the major method in document retrieval, due to its high dependence on statistical word distribution. We therefore propose a method to enhance retrieval effectiveness using dependency relationships between words in a sentence. In our method, we create a Structured Index, represented by a binary tree through dependency analysis and compound nouns analysis based on a word bigram.

This paper describes our methodology and shows the result of the retrieval experiments on the IR test collection of NTCIR1.

1 Introduction

In recent years databases have become easily accessed through the Internet, as electronic documents become more common and high performance computers less expensive. As a consequence, it has become important for users to be able to easily retrieve the information they want from large databases by phrasing their search parameters in natural language[1].

However, the keyword-based retrieval system, which is common in document retrieval, does not meet these requirements. Instead, it requires users to write complex logical expressions for queries and presents the search output in a disordered manner. There are some search engines that arrange the search output using the vector space model or the Term Frequency Inverted Document Frequency (TF–IDF) model[2]. However, there are obvious limitations in retrieval effectiveness because it is difficult to search documents using only query words and their statistical characteristics.

On the other hand, much work has been done towards solving these problems using natural language processing methods[3]. Morphological analysis could be one practical method that can be useful in information retrieval. More advanced techniques of natural language processing are only effective in limited applications [4][5].

We have previously proposed an information retrieval method for Japanese language using the dependency of words in the text and in the query[6] and shown that our method is superior to TF–IDF [7] on a prototype system with a small collection containing one database field. However, the effectiveness is not enough in the retrieval experiments on a larger collection which covers more fields than in the previous study [8][9].

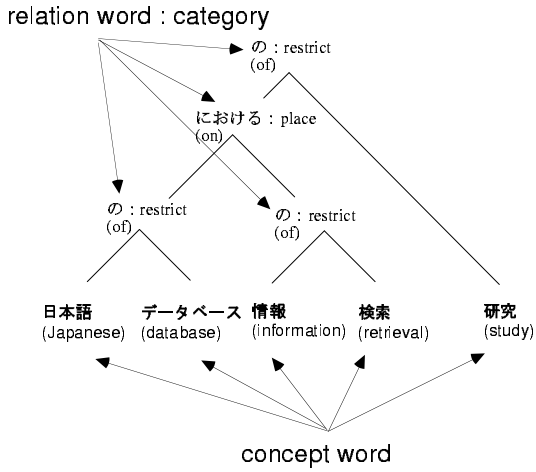

Figure 1: Structured Index

In this paper, we report the result of application of our method to IR test collection of NTCIR1, and consider the effect of the retrieval method using dependency between words.

2 Structured Indexing Method

To improve the effectiveness of retrieval, we propose an information retrieval method using the dependency relationship between words in a sentence. Since the ambiguity of words decreases by considering the relationships between them, the performance of the retrieval system is expected to be higher.

To utilize word dependency in information retrieval, we create a Structured Index represented by a binary tree, which shows the dependency between words. Figure 1 is an example of a Structured Index.

A Structured Index includes three elements. The first is

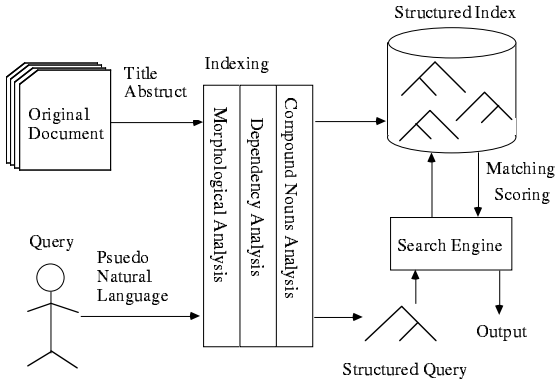

Figure 2: Processing low of our system

'concept words', which symbolize a concept and are placed on the leaf nodes in the Structured Index. The second is the 'relation word', which associates two concept words. The third element is the 'category' into which relation words are classified according to semantic similarity. Relation words and their category are placed on the internal nodes in the Structured Index.

In this way, we are able to construct a hierarchical structure of concepts and their dependency relationships. What we especially emphasize is that because of its hierarchical nature, a Structured Index treats a sentence in more detail than an index that relies only upon the binary relations between words.

Secondly, we consider reducing the user's workload. Our system requires a query to be phrased in pseudo-natural language such as the expressions found in a document's title. In this way, users can easily express their requests in their queries. Another solution is to rank the retrieved documents according to the similarity of their meaning. Users can therefore locate the document they want more efficiently.

The whole system in which the above concepts are realized is shown in Figure 2. Here we will present an outline of our system.

The first step in indexing is morphological analysis, which separates the words in the sentence into concept words and relation words. The second step is dependency analysis and compound nouns analysis, which provides the dependency relationship between concept words in a sentence. Then, the system creates the Structured Index.

As well, the retrieval query is structured in the form of a binary tree through the same process as indexing. Our system compares the structured query and the Structured Index, scores them for similarity and presents the output to the users in order.

In the following two sections, we describe in detail the indexing method and the retrieval and scoring processes.

3 Indexing Method

In this section, we present the three processes of making a Structured Index: morphological analysis, dependency analysis and compound nouns analysis.

3.1 Morphological Analysis

To decide the dependency in sentences, we divided them into the concept words and the relation words using morphological analysis. In our definition, concept words include nouns, adjectives, adverbs and constituents of compound nouns, and relation words include a post-positional particles, auxiliary verbs and their combinations. We also classified relation words into 18 categories. Table 1 shows all categories and their typical elements.

3.2 Dependency Analysis

We used the order of relation words in the sentence, or 'title pattern', to define the dependency relationship between concept words. For example, a title such as '単語間の係受け情報を用いた文献検索手法' (A method of document retrieval using dependency relationships between words) belongs to the title pattern 'A 間の B を用いた C' (C using B between A), where A, B and C are concept words or their combinations. In order to apply 'title pattern' to natural language sentences, we translate all sentences into pseudo-natural languages, simply eliminating the last relation word.

We manually assigned the dependency relationship between words to any title pattern, which appeared more than three times in the titles of 3666 scientific and technical documents. It became clear that two relationship types existed for titles of at least three concept words: for the first type we could give only one dependency relationship, and for the second, more than two.

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 '提案' (proposal), which does not have a dependency relationship with a particular word in the title but with all the title that precedes it. We defined 53 words as general words, including the above, '評価' (evaluation), '実現' (implementation) and so on. For example, let us consider the title that has three words and belongs to the title pattern 'A における B の C'(C of B on A) (Figure 3). If the title has a general word at the end, its title pattern is on the left of Figure 3. Otherwise its title pattern is on the right.

If the dependency pattern is not decided by the title pattern, we use the 'extended title pattern' in which a relation word is replaced with its category name.

When the dependency pattern is not decided, even by the extended title pattern, we divide the sentence into small parts using heuristics, then give a dependency pattern to each part. This method is important for maintaining the effectiveness of our retrieval system, because the dependency pattern given by this method is correct locally in most cases, even if the total dependency pattern is incorrect.

For the dependency analysis, we define 105 title patterns (62 are of 3 concept words and 43 are of 4 concept words)

Table 1: All categories of relation words and their typical elements.

Category name	Typical elements
restrict	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)
way	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)、を想定した (for,in)
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)、を備えた (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 (*)、は (*)
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)

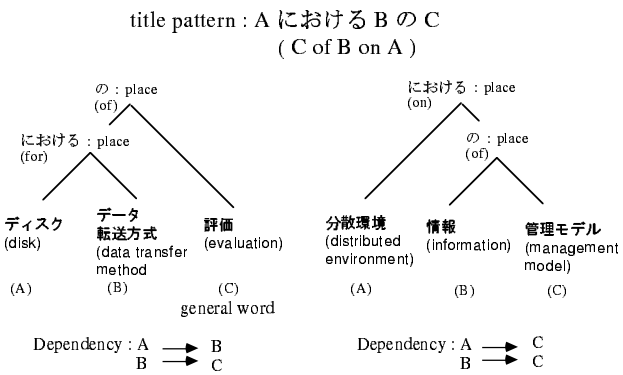

Figure 3: Ambiguity of the dependency.

and 73 extended title patterns (40 are of 3 concept words and 33 are of 4 concept words) .

3.3 Compound Nouns Analysis

It is known that a compound noun can be translated into a sentence by supplementing it with suitable function words between its constituent words [10][11]. We use this principle to give a dependency relationship to compound nouns and to place them within the framework of a Structured Index.

The central problem of this method is determining which relation words can be supplemented between concept words. We supplement 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].

When the supplementary relation words are decided, we analyze the dependency between the concept words in the compound nouns. For the dependency analysis we propose to use word bigram statistics, which are statistics of the frequency with which the two words appear next to each other in compound nouns. The word bigram can be considered as the strength of the relationship between the two concept words, if enough samples are used. In our system, for the dependency analysis of the compound nouns we use 814 bigram data, which appear more than 10 times in a 60507 bigram picked up from 13615 titles.

These processes are done automatically. All the sentences are given their dependency pattern and structured in the form of a Structured Index.

4 Retrieval and Scoring

In our system, retrieval queries are in the form of pseudo-natural language, like article titles. They are therefore structured in the form of a binary tree through the same processes explained in section 3. As a result, retrieval in our system is a question of matching a binary tree of query and binary trees of documents.

In this section, we define the similarity score of the query and the document, the method of matching and scoring on words and their dependency.

4.1 Total Score

A document has several elements, such as title or abstract, which has its own characteristics of information. To reflect their differences to scoring documents, each element is given

both word score and dependency score respectively. Total score of document d is the linear combination of the word score and the dependency score of each elements, which is shown in Equation (1).

$$S_d = \sum_b (xw_b \times SW_b + xd_b \times SD_b) \quad (1)$$

where SW_b and SD_b are word score and dependency score given to the document element b, and xw_b and xd_b are the weights of them, respectively.

4.2 Scoring on Words

Our system scores the matched words in the sentence using TF–IDF. Then, the score of matched words is defined as Equation (2).

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

where the variable in the equation is

SW_b : word score of element b

w_j : j-th word in the query

n : the total number of words in the query

δ_j : $\begin{cases} 1 & \text{when word } w_j \text{ is matched} \\ 0 & \text{otherwise} \end{cases}$

$$tfidf(w_j) = \log(tf(w_j)+1)\log(\frac{N_{all}}{df(w_j)}) \quad (3)$$

$tf(w_j)$: frequency of word w_j in the element

$df(w_j)$: the number of documents which contain the word w_j

N_{all} : the total number of the documents

4.3 Scoring on Dependency

While some elements, such as abstract, have several sentences, we must define the semantic similarity between a binary tree of the query and a set of binary trees of the element. First, we measure the similarity between a binary tree of the query and all the binary trees in the element. In this stage, there are several definition of the similarity of the query and the element.

- calculating the total of the similarity scores
- calculating the average of the similarity scores
- selecting the maximum score out of the similarity scores

In this paper, we select the maximum score as the similarity between the query and the element.

The similarity score between binary trees of query and each binary tree in the element is calculated by the sum of all scores of dependency relationships which are calculated according to the following two matching criteria.

Table 2: Weight of the scoring

parameter	value
xw_{title}	1
$xw_{abstruct}$	1
xd_{title}	0.1
$xd_{abstruct}$	0

First, we consider the level of matching between the two dependencies. Even if the two dependencies have the same words, their semantics are often different because of the difference of their dependency relation. Then, we evaluate the similarity of the two dependencies according to the following three levels.

Exact Match Two relation words are the same.

Category Match Two relation words are different but their categories are the same.

Wild Match Both two relation words and their categories are different.

Second, we use the notion of importance in the collection. Since the importance of the dependency grows according to the importance of modifier and modificand, we adopt the product of TF–IDF scores of modifier and modificand calculated by Equation (3) as the importance of the dependency.

Now we calculate the score of the dependency by measuring its similarity with the above two matching criteria. The score is shown in Equation (4).

$$SD_b = \sum_{j=1}^{m} LD(d_j)IW(wl_j, wr_j) \quad (4)$$

where the variable in the equation is

$LD(d_j)$: the weight for the matching level of the dependency d_j (Exact, Category, Wild)

$IW(wl_j, wr_j)$: the weight of importance of dependency derived from words wl_j and wr_j

$= tfidf(wl_j)tfidf(wr_j)$

5 Experiments and Evaluation

We show the result of experimental retrieval, using IR test collection of NTCIR1. Our system uses the 'DESCRIPTION' field of IR topic as queries and the 'TITL TYPE-"kanji"' and the 'ABST TYPE-"kanji"' field of data as target elements of retrieval. The weight parameters in the Equation (1) are shown in Table 2. The weights for the matching level of the dependency in the Equation (4) are shown in Table 3.

Table 3: Parameter for the best performance in these experiments

parameter	value
$LD(d_j)$	Exact Match 1.0 Category Match 0.9 Wild Match 0.1

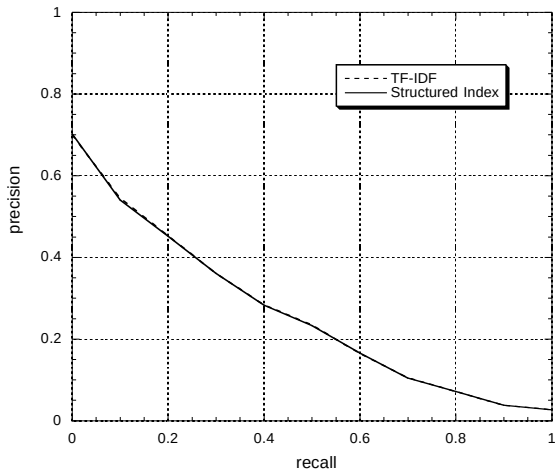

Figure 4: 11-point recall-precision curves

Figure 4 is 11point recall-precision curves of the TF–IDF method and Structured Index method for the average of 53 topics. Table 4 shows normalized precision. These results are gained when the relevant article set is the sum of the articles in ranks A and B.

In this result of average of 53 queries, it seems that the difference between the TF–IDF method and Structured Index method is small. However, the result varies depending on queries. For some queries, the use of dependency relations improves retrieval, but gives worse results for others. A preliminary analysis of the individual result shows that several factors influence the retrieval performance.

Matching method of dependency relationship The system permits for now semantically incorrect dependency in scoring the document. For example, the correct dependency relationship in a query 'WWW トラヒックの分析'(analysis of the traffic of WWW) is 'WWW トラヒック'(the traffic of WWW) and 'トラヒックの分析'(analysis of the traffic). Our system uses not only these two dependencies but also the dependency relationship 'WWW の分析'(analysis of WWW), which is incorrect semantically, in matching and scoring of the sentence. For the title retrieval, using semantically incorrect dependencies is effective in retrieving relevant documents. For the abstract retrieval, however, much more non-relevant documents are retrieved than relevant documents by using those incorrect dependencies. It is because the information content in an abstract is larger than that in a title.

Table 4: normalized precision

method	normalized precision
TF–IDF Method	0.2475
Structured Index method	0.2469

Length of a sentence Because an unimportant dependency relation is matched many times in a long sentence, the sentence is often given too heigh score. In a query '分散ネットワーク環境におけるメディア同期問題の解決' (Solution of the media synchronization problem on the distributed network environment), the dependency relationship '問題解決'(Solution of the problem) is less important than 'メディア同期'(media synchronization). However, a long sentence in which the former dependency occurs five times is given higher score than a sentence in which the latter dependency occurs only one time. As the result, the rank of the relevant document matched by the important dependency relationship is lower than that of the non-relevant document matched many times by the unimportant dependency.

Importance of the dependency The dependency score is strongly influenced by the TF of modifier or modificand because it contains the product of TF–IDFs of the modifier and the modificand.

In the case of a query 'ATM 網を用いた TCP/IP 通信のスループット特性' (Throughput characteristic of TCP/IP communication using ATM network), a non-relevant document, in which 'TCP/IP' and 'IP 通信'(IP communication) occur is given higher score of dependency than a relevant document, in which 'TCP/IP' and 'ATM 網'(ATM network) occur. One of the reason of this incorrect scoring is that the TF of 'TCP' and 'IP' of the former document is much larger than that of the latter document.

similarity between the query and the element It sometimes has a bad influence on the retrieval performance that the system uses the maximal one out of all dependency scores of sentences in the element as the similarity between the query and the element.

For the same query in the previous item, while the score of dependency of another relevant document is that of the sentence in which the dependency 'スループット特性'(throughput characteristics) occurs, the dependency 'ATM 網'(ATM network) and 'TCP/IP' also occur in another sentence of this document. On the other hand, only two dependency 'TCP/IP' and 'IP 通信'(IP communication) occur in the previous non-relevant document. However the score of document of the latter non-relevant document is higher than that of the former relevant document.

6 Conclusion

In this paper, we proposed a new method for information retrieval, a retrieval system using the dependency between words. We performed experimental evaluations of our system using IR test collection of NTCIR1. The result varies depending on queries. For some queries, using dependency relations improves retrieval, but gives worse results for others.

Our method may not be effective enough, since there are still some problems in dependency analysis and matching and scoring method.

We will improve the matching algorithm based on a thorough analysis of the difference between non-relevant document and relevant document in the retrieval outputs. Also we will determine the scoring method depending on the length of sentences and the appropriate definition of the importance of the dependency.

The present system uses the maximum score out of all dependency scores of sentence in the element as the similarity between the query and the element. Although it is not clear the influence of this definition of similarity on the retrieval performance, we will do further experiments and define similarity between the query and the element which reflects dependency relationships in all sentences of the element.

The influence of the above factors on the retrieval performance seems to vary depending on queries. We should optimize retrieval performance of our system for each query and make clear the relation between above factors and the characteristics of queries. We hope the result will help us to achieve a better retrieval system.

Acknowledgements

This research is a part of the research project "A Study on Ubiquitous Information System for Utilization of Highly Distributed Information Resources", granted by the Japan Society for the Promotion of Science.

We participate in the NTCIR Workshop, and use the NACSIS Test Collection 1. This collection is constructed by the NACSIS R&D Department using a part of "Database of Academic Conference Papers" provided by academic societies[1].

References

[1] Masato Haibara : "Intellectual Access to Document Database in Japanese", *The Journal of the IEICE*, Vol.72, No.7, pp.797–806, 1989. (in Japanese)

[2] Donna Harman : "Ranking Algorithms" in *Information Retrieval*, Chapter 14. Prentice Hall, 1992.

[3] Alan F. Smeaton : "Progress in the application of natural language processing to information retrieval tasks",*The Computer Journal*, Vol.35, No.3, 1992.

[4] Tetsuya Nasukawa : "Retrieving Japanese texts based on phrases", *The 53rd Annual Convention IPSJ*, 5T-1, 1996. (in Japanese)

[5] C.Berrut : "Indexing Medical Reports : the RIME Approach", *Information Processing and Management*, Vol.26, No.1, pp.93–110, 1990.

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

[7] Atsushi Matsumura, Kazuyuki Ikeda, Atsuhiro Takasu and Jun Adachi : "An information retrieval system using Structured Index", *Proceeding of Advanced Database Symposium '97*, pp.151–158, 1997. (in Japanese)

[8] Atsushi Matsumura, Atsuhiro Takasu, Jun Adachi : "Evaluation of Information Retrieval Method using Structured Index", Technical report of IEICE, DE98-2, 1998, pp.7–14.

[9] Atsushi Matsumura, Atsuhiro Takasu, Jun Adachi : "Information Retrieval Method using Structured Index for Japanese Text", *Proceedings of the 3rd International Workshop on Information Retrieval with Asian Languages(IRAL'98)*, 15-16 October 1998, p.109-115.

[10] Mashiro Miyazaki : "Automatic segmentation method for compound words using semantic dependent relationships between words", *Transactions of IPSJ*, Vol.25, No.6, pp.970–979, 1984. (in Japanese)

[11] Masato Ishizaki : "Relationships between words in Japanese noun compounds", *The 37th Annual Convention IPSJ*, 3C-2, 1988. (in Japanese)

[1] http://www.rd.nacsis.ac.jp/~ntcadm/acknowledge/thanks1-ja.html