

# Exploiting Term Importance Categories and Dependency Relations for Natural Language Search

**Keiji Shinzato**

Graduate School of Informatics,  
Kyoto University  
shinzato@i.kyoto-u.ac.jp

**Sadao Kurohashi**

Graduate School of Informatics,  
Kyoto University  
kuro@i.kyoto-u.ac.jp

## Abstract

In this paper, we propose a method that clearly separates terms (words and dependency relations) in a natural language query into important and other terms, and differently handles the terms according to their importance. The proposed method uses three types of term importance: necessary, optional, and unnecessary. The importance are detected using linguistic clues. We evaluated the proposed method using a test collection for Japanese information retrieval. Performance was resultantly improved by differently handling terms according to their importance.

## 1 Introduction

Currently, search engines that receive a couple of keywords reflecting users' information needs predominate. These keyword-based searches have been focused on evaluation conferences for information retrieval (IR) such as TREC and NTCIR. Search engines based on keywords, however, have a crucial problem that it is difficult for their users to represent complex needs, such as “*I want to know what Steve Jobs said about the iPod.*” A natural language sentence can more adeptly accommodate such information needs than a couple of keywords because users can straightforwardly present their needs. We call a query represented by a sentence a natural language query (NLQ).

The other advantage of NLQs is that search engines can leverage dependency relations between words in a given query. Dependency relations allow search engines to retrieve documents with a similar linguistic structure to that of the

query. Search performance improvement can be expected through the use of dependency relations.

For handling an NLQ, we can consider a conjunctive search (AND search) that retrieves documents that include all terms in the query, a simple methodology similar to real-world Web searches. This methodology, however, often leads to insufficient amounts of search results. In some instances, no documents match the query. This problem occurs because the amount of search results is inversely proportional to the number of terms used in a search; and an NLQ includes many terms. Hence, a conjunctive search simply using all terms in an NLQ is problematic.

Apart from this, we can consider conventional IR methodology. This approach performs a disjunctive search (OR search), and then ranks retrieved documents according to scores that are computed by term weights derived from retrieval models. The methodology attempts to use term weights to distinguish important terms and other items. However, a problem arises in that irrelevant documents are more highly ranked than relevant ones when giving NLQs. This is because an NLQ tends to contain some important terms and many noisy (redundant) terms and document relevancy is calculated from the combinations of these term weights.

Avoiding the above problems, we define three discrete categories of term importance: *necessary*, *optional*, and *unnecessary*, and propose a method that classifies words and dependency relations in an NLQ into term importance, and then, when performing document retrieval, differently handles the terms according to their importance. The necessary type includes expressions in Named Enti-

ties (NEs) and compound nouns, the optional includes redundant verbs and the unnecessary includes expressions that express inquiries such as “*I want to find.*” The process of IR consists of two steps: document collecting and document scoring. The proposed method uses only necessary terms for document collecting and necessary and optional terms for document scoring.

We evaluated the proposed method using the test collections built at the NTCIR-3 and NTCIR-4 conferences for evaluating Japanese IR. Search performance was resultantly improved by differently handling terms (words and dependency relations) according to their importance.

This paper is organized as follows. Section 2 shows related work, and section 3 describes how to leverage dependency relations in our retrieval method. Section 4 presents term importance categories, and section 5 gives methodology for detecting such categories. Experiment results are shown in section 6.

## 2 Related Work

A large amount of the IR methodology that has been proposed (Robertson et al., 1992; Ponte and Croft, 1998) depends on retrieval models such as probabilistic and language models. Bendersky and Croft (Bendersky and Croft, 2008), for instance, proposed a new language model in which important noun phrases can be considered.

IR methodology based on important term detection has also been proposed (Callan et al., 1995; Allan et al., 1997; Liu et al., 2004; Wei et al., 2007). These previous methods have commonly focused on noun phrases because the methods assumed that a document relates to a query if the two have common noun phrases. Liu et al. (Liu et al., 2004) classified noun phrases into four types: proper nouns, dictionary phrases (e.g., computer monitor), simple phrases, and complex phrases, and detected them from a *keyword-based* query by using named entity taggers, part-of-speech patterns, and dictionaries such as WordNet. The detected phrases were assigned different window sizes in a proximity operator according to their types. Wei et al. (Wei et al., 2007) extended Liu’s work for precisely detecting noun phrases. Their method used hit counts obtained from Google and

Wikipedia in addition to clues used in Liu’s work. The differences between the proposed method and these methods are (i) the proposed method focuses on an NLQ while the previous methods focus on a keyword-based query, (ii) the proposed method needs no dictionaries, and (iii) while the previous methods retrieve documents by proximity searches of words in phrases, the proposed method retrieves them by dependency relations in phrases. Therefore, the proposed method does not need to adjust window size, and naturally performs document retrieval based on noun phrases by using dependency relations.

Linguistically motivated IR research pointed out that dependency relations did not contribute to significantly improving performance due to low accuracy and robustness of syntactic parsers (Jones, 1999). Current state-of-the-art parsers, however, can perform high accuracy for real-world sentences. Therefore, dependency relations are remarked in IR (Miyao et al., 2006; Shinzato et al., 2008b). For instance, Miyao et al. (Miyao et al., 2006) proposed an IR system for a biomedical domain that performs deep linguistic analysis on a query and each document. Their system represented relations between words by a predicate-argument structure, and used ontological databases for handling synonyms. Their experiments using a small number of *short* queries showed that their proposed system significantly improved search performance versus a system not performing deep linguistic analysis. Shinzato et al. (Shinzato et al., 2008b) proposed a Web search system that handles not only words but also dependency relations as terms; yet they did not discuss the effectiveness of dependency relations. This paper reveals the effectiveness of dependency relations through experiments using test collections for Japanese Web searches.

## 3 Exploitation of Dependency Relation

One of the advantages of an NLQ is leveraging dependency relations between words in the query. We can expect that search performance improves because the dependency relations allow systems to retrieve documents that have similar linguistic structure to that of the query. Therefore the proposed method exploits dependency relations for

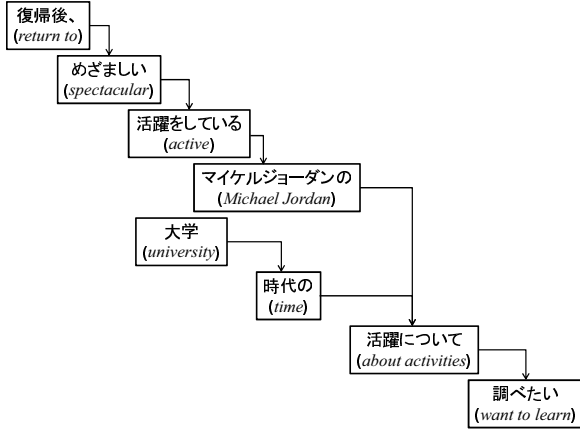


Figure 1: Parsing result of an NLQ.

retrieving documents. Though a dependency relation is generally a relation between two clauses, we regard a relation between two content words as a dependency relation. More precisely, we represent a dependency relation by a *directed* binary relation of content words, and discard the case marker between content words. Also, (compound) functional words such as “*について (about)*” and “*に従って (according to)*” are attached to the former content word. Figure 1 shows the parsing result of the query “*復帰後、めざましい活躍をしているマイケルジョーダンの大学時代の活躍について調べたい.*<sup>1</sup>” The pair of content words  $\langle$ 大学 (*university*), 時代 (*time*) $\rangle$  is extracted as a dependency relation from the parsing result. Note that the pair of content words  $\langle$ 時代 (*time*), 大学 (*university*) $\rangle$  is not extracted as a dependency relation because a dependency relation is represented by a directed binary relation.

We used Okapi BM25 (Robertson et al., 1992) for estimating relevancy between a query and a document, which is how it is used in most case, though we slightly extend this measure for estimating relevancy for dependency relations. We denote a set of words in a query  $q$  as  $T_{q_{word}}$ , and also denote a set of dependency relations in  $q$  as  $T_{q_{dpnd}}$ . The relevancy between query  $q$  and document  $d$  is as follows:

$$R(q, d) = (1 - \beta) \sum_{t \in T_{q_{word}}} BM(t, d) + \beta \sum_{t \in T_{q_{dpnd}}} BM(t, d),$$

where  $\beta$  is a parameter for adjusting the ratio of a

<sup>1</sup>This means that Michael Jordan’s performance has been spectacular since his return to NBA, and I want to learn about his activities when he was a university student.

score calculated from dependency relations. The score  $BM(t, d)$  is defined as:

$$BM(t, d) = w \times \frac{(k_1 + 1)F_{dt}}{K + F_{dt}} \times \frac{(k_3 + 1)F_{qt}}{k_3 + F_{qt}},$$

$$w = \log \frac{N - n + 0.5}{n + 0.5}, K = k_1((1 - b) + b \frac{l_d}{l_{ave}}).$$

Here,  $F_{dt}$  is the frequency with which  $t$  appears in document  $d$ ,  $F_{qt}$  is the frequency that  $t$  appears in  $q$ ,  $N$  is the number of documents being searched,  $n$  is the document frequency of  $t$ ,  $l_d$  is the length of document  $d$  (words), and  $l_{ave}$  is the average document length. Finally,  $k_1$ ,  $k_3$ , and  $b$ , are Okapi parameters, for which we use values  $k_1 = 1$ ,  $k_3 = 0$  and  $b = 0.6$ .

#### 4 Term Importance Category

Conventional IR methodology regards weights estimated by retrieval models, such as probabilistic and language models, as term importance. The methods depending on the term weights, however, cause a problem in that irrelevant documents are more highly ranked than relevant ones when an NLQ is given. This is because (i) NLQs tend to contain some important terms and a large quantity noise (redundant terms) and (ii) document relevancy is estimated by the combinations of these term weights.

Avoiding this problem, term importance is clearly separated, instead of representing by weights. We propose three term-importance categories and methodology that differently handles terms according to their importance categories. These categories are defined as follows:

**Necessary:** Terms that must be in retrieved documents. We can also consider a proximity constraint so that all retrieved documents must contain necessary terms within  $N$  words.

**Optional:** Terms preferable for inclusion in retrieved documents.

**Unnecessary:** Terms for which it does not matter if they are included in retrieved documents.

In this paper, terms in *necessary*, *optional* and *unnecessary* categories are referred to as *necessary terms*, *optional terms*, and *unnecessary terms*, respectively.

IR methodology consists of two steps: document collecting and document scoring. In the proposed method, document collecting is performed using only necessary terms, document scoring is performed using both necessary and optional terms, and neither step uses unnecessary terms.

As mentioned, the proposed method retrieves documents exploiting not only words but also dependency relations. Though a conjunctive search with words and dependency relations can be considered, the proposed method basically only uses words. In short, words are handled as *necessary* terms, while dependency relations are handled as *optional* terms. This is because the number of documents that include all dependency relations tends to be small. Importance of words and dependency relations is, however, revised depending on whether they can be regarded as important expressions. The revision methodology is described in the next section.

## 5 Revision of Term Importance

The proposed method basically deals with words and dependency relations as necessary terms and optional terms, respectively. However, the term importance of the following words and dependency relations are revised.

1. Dependency relations in NEs and strongly connected compound nouns.
2. Redundant verbs, verbs whose meaning can be inferred from surrounding nouns.
3. Words and dependency relations in inquiry expressions and functional expressions.

This section describes how to recognize the above expressions and revise the term importance of the recognized expressions.

### 5.1 Named Entity and Strongly Connected Compound Noun

The term importance of all dependency relations in Named Entities (NEs) is revised to a *necessary* category. We believe that a user entering a search engine query including an NE expects to obtain documents that include the NE. For instance, if a user’s query includes “*American Bank*,” the user prefers documents that include “*American Bank*”

to those with the individual words “*American*” and “*Bank*.” That is why the proposed method revises the term importance of all dependency relations in an NE to a necessary category. This revision guarantees that search engine users will obtain documents including the NEs in a query.

In addition to NEs, for some compound nouns a search engine user prefers to obtain documents that include the compound noun rather than the individual words in the compound noun. We refer to this as a **Strongly Connected Compound Noun** (SCCN). An example of an SCCN is “*information science*.” In the same way as “*American Bank*,” a user whose search engine query contains “*information science*” expects to obtain documents that include “*information science*” rather than with the individual words “*information*” and “*science*.”

On the other hand, there are also compound nouns, such as “*Kyoto sightseeing*”, that do not need to be included in retrieved documents as a *single phrase*. For these, a user approves of retrieved documents that include “*Kyoto*” and “*sightseeing*” separately. We therefore need criteria for distinguishing such compound nouns and SCCNs.

The problem is how to compute the connection strength of words in a compound noun  $N$  (i.e.,  $w_1, \dots, w_{|N|}$ ). For computing the connection strength among words in  $N$ , we assumed that words in an SCCN are unlikely to occur in documents as “ $w_i \circ w_{i+1}$  ( $w_{i+1}$  of  $w_i$ )”.

This assumption reflects the observation that “*Kyoto sightseeing*” is likely to be expressed as “*sightseeing of Kyoto*” and that “*information science*” is unlikely to be expressed by “*science of information*.” In line with this assumption, the connection strength is calculated as follows:

$$Score_{strength}(N) = \frac{1}{|N| - 1} \sum_{i=1}^{|N|-1} \frac{DF(w_i \ w_{i+1})}{DF(w_{i+1} \circ w_i)}.$$

Here,  $DF(X)$  is the document frequency of  $X$  computed from hundreds of millions Japanese Web pages (Shinzato et al., 2008a). The proposed method regards a compound noun  $N$  as an SCCN if the value of  $Score_{strength}(N)$  exceeds a threshold  $T_p$ . We used the value of 300 as the threshold. In addition to dependency relations in NEs,

the term importance of dependency relations in an SCCN is also revised from an *optional* category to a *necessary* category.

## 5.2 Redundant Verb

The proposed method deals with a verb whose meaning is inferable from the surrounding nouns as an optional term. We refer to such a verb a *redundant verb*.

Consider the following two expressions:

- (A) 作家 (*author*) の (*of*) 書いた (*wrote*) 本 (*book*)  
(A book written by an author)
- (B) 作家 (*author*) の (*of*) 本 (*book*)  
(A book of an author)

The expression (A) is often paraphrased as the expression (B) which omits the verb “write.” However, we can recognize that (A) is equivalent to (B). This is because the meaning of the verb “write” can be inferred from the noun “author.” In other words, the noun “author” can be considered to imply the meaning of the verb “write.” According to this observation, we assumed that a verb whose meaning is inferable from the surrounding nouns does not need to be included in retrieved documents.

For computing redundancy of verbs, we made the assumption that a noun  $n$  implies the meaning of a verb  $v$  if a syntactic dependency relation between a noun  $n$  and a verb  $v$  frequently occurs in corpora. We defined the following score function according to the assumption.

$$Score_{cooc}(n, v) = P(n, v) \cdot \log_2 \frac{P(n, v)}{P(n) \cdot P(v)},$$

where  $P(n)$  and  $P(v)$  indicate the probabilities of a noun  $n$  and a verb  $v$  respectively.  $P(n, v)$  is the probability of a dependency relation between a noun  $n$  and a verb  $v$ . These probabilities were estimated from 1.6 billion Japanese sentences extracted from the hundreds of millions of Japanese pages used for computing  $DF(X)$  in the previous section.

For each noun  $n$  that is the parent-of or child-of dependency relation of a verb  $v$ , the above score is calculated. We consider that the meaning of a verb  $v$  can be inferred from a noun  $n$  if the value

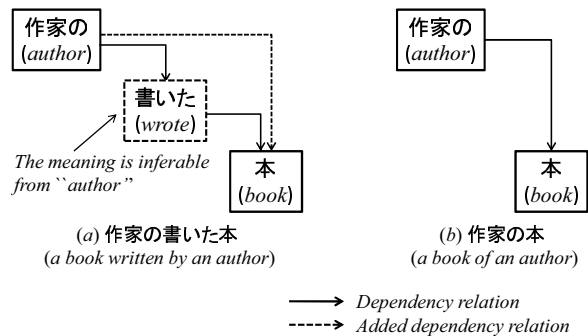


Figure 2: Structural difference between “作家の書いた本 (a book written by an author)” and “作家の本 (a book of an author)”.

of  $Score_{cooc}(n, v)$  exceeds a threshold  $T_v$ . The value of the threshold is used  $1 \times 10^{-6}$  which was decided empirically. For instance, the nouns *author* and *book* in Figure 2 (a) are used for computing the above score with respect to the verb *wrote*, and then *wrote* is regarded as a *redundant verb* if either one exceeds the threshold.

When a verb  $v$  is regarded as an optional term (i.e.,  $v$  is a redundant verb), the proposed method appends a new dependency relation consisting of the parent-of and child-of dependency relation of the redundant verb  $v$ . Figure 2 (a) shows the parsing result of the expression (A). A new dependency relation between “author” and “book” is depicted by a *dashed arrow*. Figure 2 (b) shows the parsing result of the expression (B). Though there is a structural gap between the expressions (A) and (B), this gap is bridged by the new dependency relation because the dependency relation (author, book) is contained in the both expressions.

## 5.3 Inquiry Expressions and Functional Words

An NLQ tends to contain expressions, such as “I want to find” and “I want to know,” and such expressions almost never relate to users’ information needs. Therefore we regard words and dependency relations in these expressions as unnecessary terms. To do so, we crafted the *inquiry pattern* shown in Figure 3. The importance of words and dependency relations in the matched expressions is revised to an *unnecessary* category if expressions in a query matched the pattern. The spelling variations of words, such as “探す (find)”

|  |
|--|
| <p><b>INQUIRY PATTERN:</b><br/> &lt;EPITHEM&gt;?&lt;EXPOSITION&gt;? &lt;DOC&gt;?(こつ<br/> いて (about))?&lt;PREDICATE&gt;;</p> <p><b>&lt;EPITHEM&gt;:</b> [ 詳しい (in detail)   詳細だ (in detail) ];</p> <p><b>&lt;EXPOSITION&gt;:</b> [ 説明 (explain)   書く (write)  <br/> 記述 (describe)   記載 (mention)   記す (write<br/> down)   述べる (express) ][ する (do) ]? [ (いる (be)<br/>   ある (be)   れる (reru)   られる (rareru) ]?;</p> <p><b>&lt;DOC&gt;:</b> [ ウェブ (Web)   WEB (Web) ]? [ 文書 (docu-<br/> ment)   ページ (page)   HP (homepage)   情報 (in-<br/> formation)   文章 (sentences)   テキスト (text) ];</p> <p><b>&lt;PREDICATE&gt;:</b> [ 知る (know)   探す (look for)  <br/> 調べる (find)   見る (watch)   見つける (find out)  <br/> 読む (read) ][ たい (tai)   いる (iru) ];</p> |
|--|

Figure 3: Inquiry patterns. The notation [A | B] indicates A or B and the symbol ‘?’ indicates that an expression in front of the symbol may be omitted. The words *reru*, *rareru*, *tai* and *iru* are Japanese functional words.

and “さがす (find)” are properly handled when matching an inquiry pattern.

In addition to the inquiry expressions, we can consider that content words that play a role like functional words, such as ある (*be*), なる (*become*), and 使う (*use*), are unnecessary for retrieving documents. To detect these words we constructed an unnecessary content word list.

## 6 Experiments

### 6.1 Settings

We evaluated the proposed method by using the test collections built at the NTCIR-3 (Eguchi et al., 2003) and NTCIR-4 (Eguchi et al., 2004) conferences. These share a target document set, which consists of 11,038,720 Japanese Web pages. For the evaluation, we used 127 informational topics defined in the test collections (47 from NTCIR-3 and 80 from NTCIR-4). An example of the informational topic definition is shown in Figure 4. <DESC> includes a sentence reflecting the user’s information needs; the sentence can be regarded as an NLQ. Therefore, we used only <DESC> as a query in the experiments. The relevance of each document with respect to a topic was judged as *highly relevant*, *relevant*, *partially relevant*, *irrelevant* or *unjudged*. We regarded the highly relevant, relevant, and partially relevant documents as *correct* answers.

The process of IR consists of two steps: doc-

|  |
|--|
| <pre>&lt;TOPIC&gt;&lt;NUM&gt; 0008 &lt;/NUM&gt;&lt;TITLE&gt; Salsa,<br/>learn, methods &lt;/TITLE&gt;&lt;DESC&gt; I want to<br/>find out about methods for learning how<br/>to dance the salsa &lt;/DESC&gt; .. &lt;/TOPIC&gt;</pre> |
|--|

Figure 4: Example of a search topic.

ument collecting and document scoring. In both steps, the proposed method considered synonyms automatically extracted from ordinary dictionaries and Web pages (Shibata et al., 2008). For calculating the scores, we selected the value of 0.2 as the parameter  $\beta$ . This value was estimated using the dry-run data set of NTCIR-3.

For each topic, we retrieved 1,000 documents and then assessed search performance according to MRR, P@10, R-prec, MAP, DCG<sub>N</sub> (Jarvelin and Kekalainen, 2002), and Q-Measure (QM) (Sakai, 2004). We calculated these scores for each topic then averaged them. Note that *unjudged* documents were treated as *irrelevant* when computing the scores. As the graded relevance for DCG<sub>N</sub> and QM, we mapped *highly relevant*, *relevant* and *partially relevant* to 3, 2 and 1, respectively.

The proposed method often leads to an insufficient number of search results because the method performs a conjunctive search using necessary terms. Therefore, evaluation measures, such as QM, which utilize low-ranked search results for computing their scores, give low scores in the proposed method. To avoid this problem we combine the proposed method with an OR (dpnd) search, which is described in the next section. More precisely, let  $R(d)$  denote the rank given by the proposed method for a document  $d$ , and  $R_{OR}(d)$  denote the rank given by the OR(dpnd) search. The final score for a document  $d$  is defined as:

$$S(d) = \frac{1}{R(d)} + \frac{1}{R_{OR}(d)}$$

The documents collected by the proposed method and the OR(dpnd) search are sorted according to values of  $S(d)$ , and then the top 1,000 of the sorted documents are regarded as the search result of the proposed method. Note that the search result of the OR(dpnd) search is dealt with fusing the proposed method when the number of search results of the proposed method is zero.

All NLQs extracted from <DESC> were an-

Table 1: Comparison between the proposed method and alternative methods.

| Methods             | AND   |      | OR     |      | OR (dpnd) |      |       | AND <sub>prox</sub> +<br>OR (dpnd) |        |        | Proposed method |        |        |        |        |
|---------------------|-------|------|--------|------|-----------|------|-------|------------------------------------|--------|--------|-----------------|--------|--------|--------|--------|
|                     | Prox. | Word | Prox.  | Word | Prox.     | Word | Dpnd. | Prox.                              | Word   | Dpnd.  | Prox.           | Word   |        | Dpnd.  |        |
| Search conditions   | No    | ○    | No     | △    | No        | △    | △     | Yes<br>No                          | ○<br>△ | △<br>△ | Yes<br>No       | ○<br>△ | △<br>△ | △<br>△ | ○<br>△ |
| MRR                 | 0.533 |      | 0.538  |      | 0.503     |      |       | <b>0.547</b>                       |        |        | 0.537           |        |        |        |        |
| P@10                | 0.328 |      | 0.337  |      | 0.352     |      |       | 0.352                              |        |        | <b>0.357</b>    |        |        |        |        |
| DCG <sub>10</sub>   | 3.469 |      | 3.497  |      | 3.583     |      |       | 3.634                              |        |        | <b>3.713</b>    |        |        |        |        |
| DCG <sub>100</sub>  | 7.191 |      | 8.898  |      | 9.167     |      |       | 9.045                              |        |        | <b>9.280</b>    |        |        |        |        |
| DCG <sub>1000</sub> | 8.956 |      | 16.221 |      | 16.553    |      |       | 16.678                             |        |        | <b>16.866</b>   |        |        |        |        |
| R-prec              | 0.174 |      | 0.207  |      | 0.212     |      |       | 0.217                              |        |        | <b>0.221</b>    |        |        |        |        |
| MAP                 | 0.120 |      | 0.151  |      | 0.158     |      |       | 0.161                              |        |        | <b>0.164</b>    |        |        |        |        |
| QM                  | 0.095 |      | 0.168  |      | 0.175     |      |       | 0.179                              |        |        | <b>0.183</b>    |        |        |        |        |

Prox: Proximity, Dpnd: Dependency relation, RV: Redundant verb.

alyzed by the JUMAN<sup>2</sup>, Japanese morphological analyzer and KNP<sup>3</sup>, Japanese syntactic parser which implemented the named entity recognition feature proposed by Sasano and Kurohashi (Sasano and Kurohashi, 2008). All documents were also analyzed by JUMAN and KNP, and then words and dependency relations in the documents were indexed as index terms. For instance, the dependency relation (university, time) shown in Figure 1 is indexed as *university* → *time*.

## 6.2 Comparison with Alternative Searches

We first investigated the effectiveness of clear boundaries of term importance and differently handling of terms according to their importance. We compared the proposed method with the following alternative search methods (see Table 1):

**AND:** Conjunctive search only using words. We do nothing even if the number of retrieved documents is less than 1,000. Retrieved documents are ranked according to Okapi BM25 scores. This is the same equation when the parameter  $\beta$  is regarded as zero in  $R(q, d)$ . The Prox. column in Table 1 indicates whether a proximity operator is imposed. The symbol ○ in the Word column means that words in a query are handled as necessary terms.

**OR:** Disjunctive search only using words. Retrieved documents are ranked according to Okapi BM25 scores. The symbol △ in the Word column means that words in a query are handled as optional terms.

**OR (dpnd):** Disjunctive search using both words and dependency relations. Retrieved documents are ranked according to scores of  $R(q, d)$ . We used the value of 0.2 as the parameter  $\beta$ .

**AND<sub>prox</sub>+OR(dpnd):** In the same way as the proposed method, this search consists of conjunctive search and OR search. The conjunctive search uses only words with a proximity operator. Retrieved documents must contain words in a search query within 75 words (regardless of order). The parameter value was decided by the results of pilot studies. Retrieved documents are ranked according to Okapi BM25 scores. These scores are calculated by both words and dependency relations. On the other hand, the OR(dpnd) search described above is used as an OR search. Let  $R_{prox}(d)$  denote the rank given by the conjunctive search, and  $R_{OR}(d)$  denote the rank given by the OR(dpnd) search, and the final score for a document  $d$  is defined as:

$$S(d) = \frac{1}{R_{prox}(d)} + \frac{1}{R_{OR}(d)}.$$

The documents collected by the conjunctive and OR(dpnd) searches are sorted according to the above values, then the top 1,000 documents are regarded as the search result of this search.

In the above methods, the unnecessary expressions described in Section 5.3 are not used.

The proposed method exploits dependency relations in NEs and SCCNs as necessary terms, and the other dependency relations are handled as optional terms. Redundant verbs are handled as optional terms and the others are necessary terms. The proposed method imposes the same proximity operator as the AND<sub>prox</sub>+OR (dpnd) search.

<sup>2</sup><http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman.html>

<sup>3</sup><http://nlp.kuee.kyoto-u.ac.jp/nl-resource/KNP.html>

Table 2: Comparison with systems in NTCIR3  
(a) For MRR and P@10. (b) For R-prec and MAP.

| System | MRR          | P@10         | System | R-prec       | MAP          |
|--------|--------------|--------------|--------|--------------|--------------|
| GRACE  | <b>0.502</b> | <b>0.330</b> | GRACE  | <b>0.230</b> | <b>0.208</b> |
| UAIFI5 | 0.383        | 0.289        | OKSAT  | 0.156        | 0.190        |
| NAICR  | 0.468        | 0.249        | NAICR  | 0.115        | 0.180        |
| Ours   | 0.431        | 0.313        | Ours   | 0.208        | 0.156        |

Table 3: Comparison with systems in NTCIR4.

| System | MRR          | P@10         | R-prec       | MAP          |
|--------|--------------|--------------|--------------|--------------|
| GRACE  | <b>0.645</b> | <b>0.501</b> | <b>0.278</b> | <b>0.216</b> |
| DBLAB  | 0.613        | 0.435        | 0.254        | 0.212        |
| SSTUT  | 0.562        | 0.370        | 0.189        | 0.132        |
| Ours   | 0.600        | 0.383        | 0.229        | 0.169        |

Table 1 shows performance of the proposed method and alternative methods. We can see that the proposed method outperforms not only AND and OR searches which are simple and conventional methodology but also the  $AND_{prox}+OR(dpnd)$  search. A small number of documents is returned by the AND search since the documents must include all necessary terms in a query. Because of this, the AND search indicates the worst performance in almost all evaluation measures. Though the proposed method also retrieves documents that must include all necessary terms in a query, the method achieves high performance because of its combination with the OR(dpnd) search.

From the difference between the OR and OR (dpnd) searches, we can see that dependency relations improve the performance of the OR search.

### 6.3 Comparison with Systems in NTCIR

Next we compared the search performance of the proposed method and that of systems participated in NTCIR 3 and NTCIR 4. In NTCIR 3, the measures MRR and P@10 and measures MAP and R-prec were used in different tasks. Therefore we selected the top three systems for each evaluation measure. In NTCIR 4, we selected the top three systems according to MAP.

Tables 2 and 3 show the comparison results for NTCIR3 and 4. Note that although GRACE, DBLAB and SSTUT in the tables used pseudo-relevance feedback, the proposed method did not. Tables 2 (a) and (b) show that the proposed method achieves the close performance of GRACE, the best system in NTCIR 3, in terms of

P@10 and R-prec.

On the other hand, Table 3 shows that the proposed method outperforms SSTUT, the third system in NTCIR 4. The difference between the performance of the proposed method and that of GRACE and DBLAB is derived from pseudo-relevance feedback. We expect that the proposed method achieves similar performance to GRACE and DBLAB if it utilizes pseudo-relevance feedback. Usage of of pseudo-relevance feedback is our future work.

### 6.4 Effectiveness of Dependency Relation in Document Scoring

We investigated the optimized value of the parameter  $\beta$  used to regulate the extent to which dependency relations are used in the document scoring. For estimating the value, we investigated the performance when changing the value of  $\beta$  from 0.0 to 0.9 at increments of 0.1.

The performance is shown in Table 4. The “0.0” row means that document scoring is performed without using dependency relations. We can see that dependency relations contribute to improved search performance. In particular, maximum values of most evaluation measure are indicated when the value of  $\beta$  is 0.2.

### 6.5 Influence of Redundant Verb

Next we classified all verbs in queries into redundant verbs and other verbs, then examined the search performance when changing their term importance. The result is shown in Table 5. The proposed method deals with redundant verbs as optional terms, and the others as necessary terms (Normal: ○, Redundant: △ in the table). The proposed method outperforms methods that handle all verbs as necessary terms (Normal: ○, Redundant: ○).

An example of a query that includes a redundant verb and contributes to improved search performance is “*I want to find shops that make bread with natural yeast.*” In this query, the proposed method found a document that describes “... *is a well-known bakery. Bread with natural yeast is a popular item.*” Though this document did not include the verb “*make,*” we were able to find it because the redundant verb detection procedure de-



Table 4: Changes in search performance, when varying the parameter  $\beta$  in document scoring.

| $\beta$ | MRR          | P@10         | DCG <sub>10</sub> | DCG <sub>100</sub> | DCG <sub>1000</sub> | R-prec       | MAP          | QM           |
|---------|--------------|--------------|-------------------|--------------------|---------------------|--------------|--------------|--------------|
| 0.0     | <b>0.548</b> | 0.341        | 3.528             | 9.108              | 17.209              | 0.208        | 0.151        | 0.170        |
| 0.1     | 0.529        | 0.350        | 3.619             | 9.265              | <b>17.454</b>       | 0.214        | 0.155        | 0.173        |
| 0.2     | 0.537        | <b>0.357</b> | <b>3.713</b>      | <b>9.280</b>       | 16.866              | <b>0.221</b> | <b>0.164</b> | <b>0.183</b> |
| 0.3     | 0.497        | 0.338        | 3.446             | 9.174              | 17.418              | 0.209        | 0.152        | 0.171        |
| 0.4     | 0.507        | 0.339        | 3.335             | 8.791              | 17.038              | 0.199        | 0.145        | 0.164        |
| 0.5     | 0.486        | 0.320        | 3.150             | 8.307              | 16.482              | 0.191        | 0.136        | 0.154        |
| 0.6     | 0.467        | 0.303        | 2.988             | 7.793              | 15.645              | 0.174        | 0.126        | 0.143        |
| 0.7     | 0.458        | 0.292        | 2.873             | 7.384              | 14.777              | 0.166        | 0.118        | 0.133        |
| 0.8     | 0.456        | 0.278        | 2.790             | 7.059              | 14.216              | 0.157        | 0.110        | 0.124        |
| 0.9     | 0.447        | 0.263        | 2.646             | 6.681              | 13.569              | 0.148        | 0.104        | 0.117        |

scribed in Section 5.2 judged that the meaning of “make” is inferable from “bread.”

The highest performance, however, was achieved when regarding all verbs as *optional* terms (Normal:  $\Delta$ , Redundant:  $\Delta$ ). In this setting, the example of a query that contributes to improved search performance is “I want to find out how the heliocentric theory of Copernicus was accepted by Christian society.” The redundant verb detection procedure judged that the meaning of “accept” is not inferable from “society.” Consequently, the verb “accept” is handled as a necessary term. Though this judgement is correct, the handling of verbs as necessary terms means that the possibility of the same event being expressed by different expressions such as synonyms is discarded. In general, a verb has multiple synonyms, and multiple expressions can be considered for describing the identical event. The handling of verbs as necessary terms can thereby be a cause of decreased search performance. We cope with the side effect of verbs by expanding synonym databases.

## 6.6 Influence of Dependency Relation Usage

Finally we investigated search performance when changing importance of dependency relations.

Table 6 shows that scores of all evaluation measures are close to each other when we simply used all dependency relations as *necessary*, *optional* or *unnecessary* terms. On the other hand, the proposed method handles dependency relations in NEs and SCCNs as necessary terms, and handles the other dependency relations as optional terms. This setting achieves relatively higher performance than the other settings. This means that the different handling of dependency relations according to their categories improves search performance.

## 7 Conclusion

In this paper, we defined three term importance categories: *necessary*; *optional* and *unnecessary*, and proposed a method that classifies terms in an NLQ into a category. The term importance is detected by word co-occurrence frequencies estimated from large-scale Web documents and NE recognition. The proposed method also handles dependency relations in a query as terms for achieving high performance.

We evaluated the proposed method using the NTCIR-3 and NTCIR-4 test collections for Japanese information retrieval. The search performance resultantly improved by regarding terms (words and dependency relations) in the named entities and compound nouns as *necessary* terms. Moreover, the performance was partially improved by regarding redundant verbs as *optional*.

## References

- Allan, James, Jamie Callan, W. Bruce Croft, Lisa Ballesteros, John Broglio, Jinxi Xu, and Hongmin Shu. 1997. Inquiry at trec-5. In *NIST*, pages 119–132.
- Bendersky, Michael and W. Bruce Croft. 2008. Discovering key concepts in verbose queries query. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval 2008*, pages 491–498.
- Callan, James P., W. Bruce Croft, and John Broglio. 1995. Trec and tipster experiments with inquiry. *Inf. Process. Manage.*, 31(3):327–343.
- Eguchi, Koji, Keizo Oyama, Emi Ishida, Noriko Kando, and Kazuko Kuriyama. 2003. The web retrieval task and its evaluation in the third ntcir workshop. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Table 5: Changes in search performance, when varying term importance of verbs.

| Verbs  |           | MRR          | P@10         | DCG <sub>10</sub> | DCG <sub>100</sub> | DCG <sub>1000</sub> | R-prec       | MAP          | QM           |
|--------|-----------|--------------|--------------|-------------------|--------------------|---------------------|--------------|--------------|--------------|
| Normal | Redundant |              |              |                   |                    |                     |              |              |              |
| ○      | ○         | 0.525        | 0.352        | 3.640             | 9.110              | 16.734              | 0.217        | 0.161        | 0.180        |
| ○      | △         | <b>0.537</b> | 0.357        | 3.713             | 9.280              | 16.866              | <b>0.221</b> | 0.164        | 0.183        |
| ○      | ×         | 0.534        | 0.354        | 3.664             | 9.273              | 16.832              | <b>0.221</b> | 0.164        | 0.183        |
| △      | △         | <b>0.537</b> | <b>0.360</b> | <b>3.755</b>      | <b>9.404</b>       | <b>17.053</b>       | <b>0.221</b> | <b>0.165</b> | <b>0.184</b> |
| △      | ×         | 0.534        | 0.357        | 3.709             | 9.399              | 17.019              | <b>0.221</b> | <b>0.165</b> | <b>0.184</b> |
| ×      | ×         | 0.533        | 0.356        | 3.703             | 9.401              | 17.018              | <b>0.221</b> | <b>0.165</b> | <b>0.184</b> |

Table 6: Changes in search performance, when varying the importance of dependency relations.

| Dependency relations   |                       | MRR          | P@10         | DCG <sub>10</sub> | DCG <sub>100</sub> | DCG <sub>1000</sub> | R-prec       | MAP          | QM           |
|------------------------|-----------------------|--------------|--------------|-------------------|--------------------|---------------------|--------------|--------------|--------------|
| Outside of NEs & SCCNs | Inside of NEs & SCCNs |              |              |                   |                    |                     |              |              |              |
| ○                      | ○                     | 0.513        | 0.338        | 3.474             | 8.987              | 16.650              | 0.211        | 0.155        | 0.174        |
| △                      | ○                     | 0.537        | 0.357        | 3.713             | <b>9.280</b>       | <b>16.866</b>       | <b>0.221</b> | <b>0.164</b> | <b>0.183</b> |
| ×                      | ○                     | <b>0.561</b> | 0.349        | 3.642             | 9.072              | 16.547              | 0.213        | 0.159        | 0.177        |
| △                      | △                     | 0.552        | 0.347        | 3.647             | 9.073              | 16.565              | 0.215        | 0.159        | 0.177        |
| ×                      | △                     | 0.539        | <b>0.359</b> | <b>3.725</b>      | 9.223              | 16.827              | <b>0.221</b> | <b>0.164</b> | 0.182        |
| ×                      | ×                     | <b>0.561</b> | 0.344        | 3.655             | 9.059              | 16.545              | 0.214        | 0.159        | 0.177        |

Eguchi, Koji, Keizo Oyama, Akiko Aizawa, and Haruko Ishikawa. 2004. Overview of web task at the fourth ntcir workshop. In *Proceedings of the Fourth NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization*.

Jarvelin, Kalervo and Jaana Kekalainen. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20:422–446.

Jones, Karen Sparck. 1999. What is the role of nlp in text retrieval? In Strzalkowski, T., editor, *Natural language information retrieval*, pages 1–24. Kluwer Academic Publishers.

Liu, Shuang, Fang Liu, Clement Yu, and Weiyi Meng. 2004. An effective approach to document retrieval via utilizing wordnet and recognizing phrases. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 266–272.

Miyao, Yusuke, Tomoko Ohta, Katsuya Masuda, Yoshimasa Tsuruoka, Kazuhiro Yoshida, Takashi Ninomiya, and Jun’ichi Tsujii. 2006. Semantic retrieval for the accurate identification of relational concepts in massive textbases. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, pages 1017–1024.

Ponte, Jay M. and W. Bruce Croft. 1998. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281.

Robertson, Stephen E., Steve Walker, Micheline Hancock-Beaulieu, Aaron Gull, and Marianna Lau.

1992. Okapi at TREC. In *Text REtrieval Conference*, pages 21–30.

Sakai, Tetsuya. 2004. New performance metrics based on multigrade relevance: Their application to question answering. In *Proceedings of the Fourth NTCIR Workshop Meeting*.

Sasano, Ryohei and Sadao Kurohashi. 2008. Japanese named entity recognition using structural natural language processing. In *Proceedings of Third International Joint Conference on Natural Language Processing*, pages 607–612.

Shibata, Tomohide, Michitaka Odani, Jun Harashima, Takashi Oonishi, and Sadao Kurohashi. 2008. SYNGRAPH: A flexible matching method based on synonymous expression extraction from an ordinary dictionary and a web corpus. In *Proc. of IJCNLP2008*, pages 787–792.

Shinzato, Keiji, Daisuke Kawahara, Chikara Hashimoto, and Sadao Kurohashi. 2008a. A large-scale web data collection as a natural language processing infrastructure. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC08)*.

Shinzato, Keiji, Tomohide Shibata, Daisuke Kawahara, Chikara Hashimoto, and Sadao Kurohashi. 2008b. TSUBAKI: An open search engine infrastructure for developing new information access methodology. In *Proc. of IJCNLP2008*, pages 189–196.

Wei, Zhang, Liu Shuang, Yu Clement, Sun Chaojing, Liu Fang, and Meng Weiyi. 2007. Recognition and classification of noun phrases in queries for effective retrieval. In *Proceedings of the sixteenth ACM conference on information and knowledge management*, pages 711–720.