Paraphrasing Japanese noun phrases using character-based indexing

Tokunaga Takenobu **Tanaka Hozumi** **Kimura Kenji**
Department of Computer Science, Tokyo Institute of Technology
Tokyo Meguro Ôokayama 2-12-1, 152-8552 Japan
take@cl.cs.titech.ac.jp

Abstract

This paper proposes a novel method to extract paraphrases of Japanese noun phrases from a set of documents. The proposed method consists of three steps: (1) retrieving passages using character-based index terms given a noun phrase as an input query, (2) filtering the retrieved passages with syntactic and semantic constraints, and (3) ranking the passages and reformatting them into grammatical forms. Experiments were conducted to evaluate the method by using 53 noun phrases and three years worth of newspaper articles. The accuracy of the method needs to be further improved for fully automatic paraphrasing but the proposed method can extract novel paraphrases which past approaches could not.

1 Introduction

We can use various linguistic expressions to denote a concept by virtue of richness of natural language. However this richness becomes a crucial obstacle when processing natural language by computer. For example, mismatches of index terms cause failure of retrieving relevant documents in information retrieval systems, in which documents are retrieved on the basis of surface string matching. To remedy this problem, the current information retrieval system adopts query expansion techniques which replace a query term with a set of its synonyms (Baeza-Yates and Riberto-Neto, 1999). The query expansion works well for single-word index terms, but more sophisticated techniques are necessary for larger index units, such as phrases. The effectiveness of phrasal indexing has recently drawn researchers' attention (Lewis, 1992; Mitra et al., 1997; Tokunaga et al., 2002). However, query expansion of phrasal index terms has not been fully investigated yet (Jacquemin et al., 1997).

To deal with variations of linguistic expressions, paraphrasing has recently been studied for various applications of natural language processing, such as machine translation (Mitamura, 2001; Shimohata and Sumita, 2002), dialog systems (Ebert et al., 2001), QA systems (Katz, 1997) and information extraction (Shinyama et al., 2002). Paraphrasing is defined as a process of transforming an expression into another while keeping its meaning intact. However, it is difficult to define what "keeping its meaning intact" means, although it is the core of the definition. On what basis could we consider different linguistic expressions denoting the same meaning? This becomes a crucial question when finding paraphrases automatically.

In past research, various types of clues have been used to find paraphrases. For example, Shinyama et al. tried to find paraphrases assuming that two sentences sharing many Named Entities and a similar structure are likely to be paraphrases of each other (Shinyama et al., 2002). Barzilay and McKeown assume that two translations from the same original text contain paraphrases (Barzilay and McKeown, 2001). Torisawa used subcategorization information of verbs to paraphrase Japanese noun phrase construction "NP_1 *no* NP_2" into a noun phrase with a relative clause (Torisawa, 2001). Most of previous work on paraphrasing took corpus-based approach with notable exceptions of Jacquemin (Jacquemin et al., 1997; Jacquemin, 1999) and Katz (Katz, 1997). In particular, text alignment technique is generally used to find sentence level paraphrases (Shimohata and Sumita, 2002; Barzilay and Lee, 2002).

In this paper, we follow the corpus-based approach and propose a method to find paraphrases of a Japanese noun phrase in a large corpus using information retrieval techniques. The significant feature of our method is use of character-based indexing. Japanese uses four types of writing; *Kanzi* (Chinese characters), *Hiragana*, *Katakana*, and Roman alphabet. Among these, *Hiragana* and *Katakana* are phonographic, and *Kanzi* is an ideographic writing. Each *Kanzi* character itself has a certain meaning and provides a basis for rich word formation ability for Japanese. We use *Kanzi* characters as index terms to retrieve paraphrase candidates, assuming that noun phrases sharing the same *Kanzi* characters could be paraphrases of each other. For example, character-based indexing enables us to retrieve a paraphrase "通学する子供 (a commuting child)" for "学校に通う子供 (a child going to school)". Note that their head is the same, "子供 (child)", and their modifiers are different but sharing common characters "通 (commute)" and "学 (study)". As shown in this example, the paraphrases generated based on Japanese word formation rule cannot be classified in terms of the past paraphrase classification (Jacquemin et al., 1997).

The proposed method is summarized as follows. Given a Japanese noun phrase as input, the method finds its paraphrases in a set of documents. In this paper, we used a collection of newspaper articles as a set of documents, from which paraphrases are retrieved. The process is decomposed into following three steps:

1. retrieving paraphrase candidates,
2. filtering the retrieved candidates based on syntactic and semantic constraints, and
3. ranking the resulting candidates.

Newspaper articles are segmented into passages at punctuation symbols, then the passages are indexed based on *Kanzi* characters and stored in a database. The database is searched with a query, an input noun phrase, to obtain a set of passages, which are paraphrase candidates. In general, using smaller index units, such as characters, results in gains in recall at the cost of precision. To remedy this, we introduce a filtering step after retrieving paraphrase candidates. Filtering is performed based on syntactic and semantic constraints. The resulting candidates are ranked and provided as paraphrases.

The following three sections 2, 3 and 4 describe each of three steps in detail. Section 5 describes experiments to evaluate the proposed method. Finally, section 6 concludes the paper and looks at the future work.

2 Retrieving paraphrase candidates

2.1 Indexing and term expansion

In conventional information retrieval, a query is given to the system to retrieve a list of documents which are arranged in descending order of relevance. Our aim is to obtain paraphrases given a noun phrase as a query, where retrieved objects should be smaller than documents. We divide a document into a set of passages at punctuation symbols. These passages are retrieved by a given query, a noun phrase.

The input noun phrase and the passages are segmented into words and they are assigned part of speech tags by a morphological analyzer. Among these tagged words, content words (nouns, verbs, adjectives, adverbs) and unknown words are selected. *Kanzi* characters contained in these words are extracted as index terms. In addition to *Kanzi* characters, words written in *Katakana* (most of them are imported words) and numbers are also used as index terms. Precisely speaking, different numbers should be considered to denote different meaning, but to avoid data sparseness problem, we abstract numbers into a special symbol $\langle num \rangle$.

As mentioned in section 1, the query expansion technique is often used in information retrieval to solve the surface notational difference between queries and documents. We also introduce query expansion for retrieving passage. Since we use *Kanzi* characters as index terms, we need linguistic knowledge defining groups of similar characters for query expansion. However this kind of knowledge is not available at hand. We obtain similarity of *Kanzi* characters from an ordinary thesaurus which defines similarity of words.

If a word t is not a *Katakana* word, we expand it to a set of *Kanzi* characters $E(t)$ which is defined by (1), where C_t is a semantic class including the word t, K_C is a set of *Kanzi* characters used in words of semantic class C, $fr(k, C)$ is a frequency of a *Kanzi* character k used in words of semantic class C, and K_t is a set of *Kanzi* characters in word t.

$$\begin{aligned} E(t) = \{k | k \in K_{C_t}, k' = \arg\max_{l \in K_t} fr(l, C_t), \\ fr(k, C_t) > fr(k', C_t)\} \cup K_t \cup \\ \{s | s \in C_t, s \text{ is a \textit{Katakana} word}\} \end{aligned} \quad (1)$$

$E(t)$ consists of *Kanzi* characters which is used in words of semantic class C_t more frequently, than the most frequent *Kanzi* character in the word t. If the word t is a *Katakana* word, it is not expanded.

Let us see an expansion example of word "温泉 (hot spring)". Here we have $t =$ "温泉" to expand, and we have two characters that make the word, i.e. $K_t = \{$ 温, 泉 $\}$. Suppose "温泉" belongs to a semantic class C_t in which we find a set of words { 温泉郷 (hot sprint place), ぬるま湯 (lukewarm water), 温水 (warm water), スパ (spa), オアシス (oasis), ...}. From this word set, we extract characters and count their occurence to obtain $K_{C_t} = \{$ 湯 (35), 泉 (22), 村 (20), 温 (8),...}, where a number in parentheses denotes the frequency of characters in the semantic class C_t. Since the most frequent character of K_t in K_{C_t} is "泉" in this case, more frequently used character "湯" is added to $E(t)$. In addition, *Katakana* words "スパ" and "オアシス" are added to $E(t)$ as well.

2.2 Term weighting

An index term is usually assigned a certain weight according to its importance in user's query and documents. There are many proposals of term weighting most of which are based on term frequency (Baeza-Yates and Riberto-Neto, 1999) in a query and documents. Term frequency-based weighting resides on Luhn's assumption (Luhn, 1957) that a repeatedly mentioned expression denotes an important concepts. However it is obvious that this assumption does not hold when retrieving paraphrase candidates from a set of documents. For term weighting, we use character frequency in a semantic class rather than that in a query and documents, assuming that a character frequently used in words of a semantic class represents the concept of that semantic class very well.

A weight of a term k in a word t is calculated by (2).

$$w(k) = \begin{cases} 100 \\ \qquad \text{if } k \text{ is \textit{Katakana} word or } \langle num \rangle \\ 100 \times \dfrac{\log fr(k, C_t)}{\displaystyle\sum_{k' inE(t)} \log fr(k', C_t)} \\ \qquad \text{if } k \text{ is a \textit{Kanzi}} \end{cases} \tag{2}$$

Katakana words and numbers are assigned a constant value, 100, and a *Kanzi* character is assigned a weight according to its frequency in the semantic class C_t, where k is used in the word t.

In the previous example of “温泉”, we have obtained an expanded term set { 湯, 温, 泉, スパ, オアシス }. Among this set, “スパ” and “オアシス” are assigned weight 100 because these are *Katakana* words, and the rest three characters are assigned weight according to its frequency in the class. For example, “湯” is assigned weight $100 \times \frac{\log 35}{\log 35 + \log 22 + \log 8} = 40.7$.

2.3 Similarity

Similarity between an input noun phrase (I) and a passage (D) is calculated by summing up the weights of terms which are shared by I and D, as defined in (3). In the equation, k takes values over the index terms shared by I and D, $w(k)$ is its weight calculated as described in the previous section.

$$sim(I, D) = \sum_{k \in I \wedge k \in D} w(k) \tag{3}$$

Note that since we do not use term frequency in passages, we do not introduce normalization of passage length.

3 Syntactic and semantic filtering

The proposed method utilizes *Kanzi* characters as index terms. In general, making index terms smaller units increases exhaustivity to gain recall, but, at the same time, it decreases specificity to degrade precision (Sparck Jones, 1972). We aim to gain recall by using smaller units as index terms at the cost of precision. Even though *Kanzi* are ideograms and have more specificity than phonograms, they are still less specific than words. Therefore there would be many irrelevant passages retrieved due to coincidentally shared characters. In this section, we describe a process to filter out irrelevant passages based on the following two viewpoints.

Semantic constraints : Retrieved passages should contain all concepts mentioned in the input noun phrase.

Syntactic constraints : Retrieved passages should have a syntactically proper structure corresponding to the input noun phrase.

3.1 Semantic constraints

In the indexing phase, we have decomposed an input noun phrase and passages into a set of *Kanzi* characters for retrieval. In the filtering phase, from these characters, we reconstruct words denoting a concept and verify if concepts mentioned in the input noun phrase are also included in the retrieved passages.

To achieve this, a retrieved passage is syntactically analyzed and dependencies between *bunsetu* (word phrase) are identified. Then, the correspondence between words of the input noun phrase and *bunsetu* of the passage is verified. This matching is done on the basis of sharing the same *Kanzi* characters or the same *Katakana* words. Passages missing any of the concepts mentioned in the input noun phrase are discarded in this phase.

3.2 Syntactic constraints

Since passages are generated on the basis of punctuation symbols, each passage is not guaranteed to have a syntactically proper structure. In addition, a part of the passage tends to be a paraphrase of the input noun phrase rather than the whole passage. In such cases, it is necessary to extract a corresponding part from the retrieved passage and transform it into a proper syntactic structure.

By applying semantic constraints above, we have identified a set of *bunsetu* covering the concepts mentioned in the input noun phrase. We extract a minimum dependency structure which covers all the identified *bunsetu*.

Finally the extracted structure is transformed into a proper phrase or clause by changing the ending of the head (the right most element) and deleting unnecessary elements such as punctuation symbols, particles and so on.

Figure 1 illustrates the matching and transforming process described in this section. The input noun phrase is “電話 $_{w_1}$ 料金 $_{w_2}$ の $_{w_3}$ 引き下げ $_{w_4}$ (reduction of telephone rate)” which consists of four words $w_1 \ldots w_4$. Suppose a passage “同社が通話料金を値下げしたことで

(the company's telephone rate reduction caused. . . ” is retrieved. This passage is syntactically analyzed to give the dependency structure of four *bunsetu* $b_1 \ldots b_4$ as shown in Figure 1.

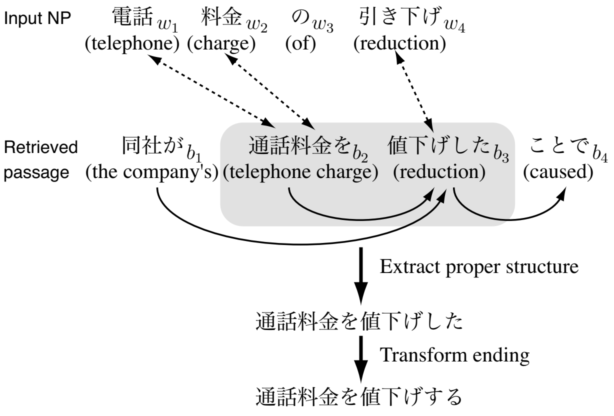

Figure 1: An example of matching and transformation

Correspondence between word w_1 and *bunsetu* b_2 is made bacause they share a common character “話”. Word w_2 corresponds to *bunsetu* b_2 as well due to characters “ 料” and “金”. And word w_4 corresponds to *bunsetu* b_3. Although there is no counterpart of word w_3, this passage is not discarded because word w_3 is a function word (postposition). After making correspondences, a minimum dependency structure, the shaded part in Figure 1, is extracted. Then the ending auxiliary verb is deleted and the verb is restored to the base form.

4 Ranking

Retrieved passages are ranked according to the similarity with an input noun phrase as described in section 2. However this ranking is not always suitable from the viewpoint of paraphrasing. Some of the retrieved passages are discarded and others are transformed through processes described in the previous section. In this section, we describe a process to rerank remaining passages according to their appropriateness as paraphrases of the input noun phrase. We take into account the following three factors for reranking.

- Similarity score of passage retrieval
- Distance between words
- Contextual information

The following subsections describe each of these factors.

4.1 Similarity score of retrieval

The similarity score used in passage retrieval is not sufficient for evaluating the quality of the paraphrases. However, it reflects relatedness between the input noun phrase and retrieved passages. Therefore, the similarity score calculated by (3) is taken into account when ranking paraphrase candidates.

4.2 Distance between words

In general, distance between words which have a dependency relation reflects the strength of their semantic closeness. We take into account the distance between two *bunsetu* which have a dependency relation and contain adjacent two words in the input noun phrase respectively. This factor is formalized as in (4), where t_i is the ith word in the input noun phrase, and $dist(s, t)$ is the distance between two *bunsetu* each of which contains s and t. A distance between two *bunsetu* is defined as the number of *bunsetu* between them. When two words are contained in the same *bunsetu*, the distance between them is defined as 0.

$$M_{distance} = \frac{1}{1 + \sum_i dist(t_i, t_{i+1})} \qquad (4)$$

4.3 Contextual information

We assume that phrases sharing the same *Kanzi* characters likely represent the same meaning. Therefore they could be paraphrases of each other. However, even though a *Kanzi* denotes a certain meaning, its meaning is often ambiguous. This problem is similar to word sense ambiguities, which have been studied for many years. To solve this problem, we adopt an idea *one sense per collocation* which was introduced in word sense disambiguation research (Yarowsky, 1995). Considering a newspaper article in which the retrieved passage and the input noun phrase is included as the context, the context similarity is taken into account for ranking paraphrase candidates. More concretely, context similarity is calculated by following procedure.

1. For each paraphrase candidate, a context vector is constructed from the newspaper article containing the passage from which the candidate is derived. The article is morphologically analyzed and content words are extracted to make the context vector. The $tf \cdot idf$ metric is used for term weighting.

2. Since the input is given in terms of a noun phrase, there is no corresponding newspaper article for the input. However there is a case where the retrieved passages include the input noun phrase. Such passages are not useful for finding paraphrases, but useful for constructing a context vector of the input noun phrase. The context vector of the input noun phrase is constructed in the same manner as that of paraphrase candidates, except that all newspaper articles including the noun phrase are used.

3. Context similarity $M_{context}$ is calculated by cosine measure of two context vectors as in (5), where $w_i(k)$ and $w_d(k)$ are the weight of the k-th term of the input context vector and the candidate context vector, respectively.

$$M_{context} = \frac{\sum_k w_i(k) w_d(k)}{\sqrt{\sum_k w_i^2(k)} \sqrt{\sum_k w_d^2(k)}} \quad (5)$$

4.4 Ranking paraphrase candidates

Paraphrase candidates are ranked in descending order of the product of three measures, $sim(I, D)$ (equation (3)), $M_{distance}$ (equation (4)) and $M_{context}$ (equation (5)).

5 Experiments

5.1 Data and preprocessing

As input noun phrases, we used 53 queries excerpted from Japanese IR test collection BMIR-J2[1] (Kitani et al., 1998) based on the following criteria.

- A query has two or more index terms.
  It is less likely to retrieve proper paraphrases with only one index term, since we adopt character-based indexing.

- A query does not contain proper names.
  It is generally difficult to paraphrase proper names. We do not deal with proper name paraphrasing.

- A query contains at most one *Katakana* word or number.
  The proposed method utilize characteristics of *Kanzi* characters, ideograms. It is obvious that the method does not work well for *Kanzi* -poor expressions.

We searched paraphrases in three years worth of newspaper articles (Mainichi Shimbun) from 1991 to 1993. As described in section 2, each article is segmented into passages at punctuation marks and symbols. These passages are assigned a unique identifier and indexed, then stored in the GETA retrieval engine (IPA, 2003). We used the JUMAN morphological analyzer (Kurohashi and Nagao, 1998) for indexing the passages. As a result of preprocessing described above, we obtained 6,589,537 passages to retrieve. The average number of indexes of a passage was 12.

5.2 Qualitative evaluation

Out of 53 input noun phrases, no paraphrase was obtained for 7 cases. Output paraphrases could be classified into the following categories.

(1) The paraphrase has the same meaning as that of the input noun phrase.
e.g. 冷夏の被害 (damage by cool summer) → 冷害 (cool summer damage)[2]
Note that this example is hardly obtained by the existing approaches such as syntactic transformation and word substitution with thesaurus.

(2) The paraphrase does not have exactly the same meaning but has related meaning. This category is further divided into three subcategories.

(2-a) The meaning of the paraphrase is more specific than that of the input noun phrase.
e.g. 農薬 (agricultural chemicals)→ 殺虫・除草剤 (insecticide and herbicide)

(2-b) The meaning of the paraphrase is more general than that of the input noun phrase.
e.g. 株価動向 (stock movement) → 株価、為替相場の動向 (movement of stock and exchange rate)

(2-c) The paraphrase has related meaning to the input but is not categorized into above two.
e.g. 飲料品 (drinks) → 国際食品飲料展 (international drink exhibition)

(3) There is no relation between the paraphrase and the input noun phrase.

Among these categories, (1) and (2-a) are useful from a viewpoint of information retrieval. By adding the paraphrase of these classes to a query, we can expect the effective phrase expansion in queries.

Since the paraphrase of (2-b) generalizes the concept denoted by the input, using these paraphrases for query expansion might degrade precision of the retrieval. However, they might be useful for the recall-oriented retrieval. The paraphrases of (2-c) have the similar property, since *relatedness* includes various viewpoints.

The main reason of retrieval failure and irrelevant retrieval (3) are summarized as follows:

- The system cannot generate a paraphrase, when there is no proper paraphrase for the input. In particular, this tends to be the case for single-word inputs, such as “液晶 (liquid crystal)” and “映画 (movie)”. But this does not imply the proposed method does not work well for single-words inputs. We had several interesting paraphrases for single-word inputs, such as “農園芸用薬剤 (chemicals for agriculture and gardening)” for “農薬 (agricultural chemicals)”.

- We used only three years worth of newspaper articles due to the limitation of computational resoruces. Sometimes, the system could not generate

[1] BMIR-2 contains 60 queries.

[2] The left-hand side of the arrow is the input and the right-hand side is its paraphrase.

the paraphrase of the input because of the limited size of the corpus.

5.3 Quantitative evaluation

Since there is no test collection available to evaluate paraphrasing, we asked three judges to evaluate the output of the system subjectively. The judges classified the outputs into the categories introduced in 5.2. The evaluation was done on the 46 inputs which gave at least one output.

Table 1 shows the results of judgments. Column "Q" denotes the query identifier, "Len." denotes its length in morphemes, "#Para." denotes the number of outputs and the columns (1) through (3) denote the number of outputs which are classified into each category by three judges. Therefore, the sum of these columns makes a triple of the number of outputs. The decimal numbers in the parentheses denote the generalized raw agreement indices of each category, which are calculated as given in (6) (Uebersax, 2001), where K is the number of judged cases, C is the number of categories, n_{jk} is the number of times category j is applied to case k, and n_k is calculated by summing up over categories on case k; $n_k = \sum_{j=1}^{C} n_{jk}$.

$$p_s(j) = \frac{\sum_{k=1}^{K} n_{jk}(n_{jk} - 1)}{\sum_{k=1}^{K} n_k - 1} \qquad (6)$$

In our case, K is the number of outputs (column "#Para."), n_k is the number of judges, 3, and j moves over (1) through (3).

As discussed in 5.2, from the viewpoint of information retrieval, paraphrases of category (1) and (2-a) are useful for query expansion of phrasal index terms. Column "Acc." denotes the ratio of paraphrases of category (1) and (2-a) to the total outputs. Column "Prec." denotes non-interpolated average precision. Since the precision differs depending on the judge, the column is showing the average of the precisions given by three judges.

We could obtain 45 paraphrases on average for each input. But the average accuracy is quite low, 10%, which means only one tenth of output is useful. Even though considering that all paraphrases not being in category (3) are useful, the accuracy only doubled. This means filtering conditions should be more rigid. However, looking at the agreement indices, we see that category (3) ranks very high. Therefore, we expect finding the paraphrases in category (3) is easy for a human. From all this, we conclude that the proposed method need to be improved in accuracy to be used for automatic query expansion in information retrieval, but it is usable to help users to modify their queries by suggesting possible paraphrases.

Seeing the column "Len.", we find that the proposed method does not work for complex noun phrases. The average length of input noun phrase is 4.5 morphemes. The longer input often results in less useful paraphrases. The number of outputs also decreases for longer inputs. We require all concepts mentioned in the input to have their counterparts in its paraphrases as described in 3.1. This condition seems to be strict for longer inputs. In addition, we need to take into account syntactic variations of longer inputs. Integrating syntactic transformation into the proposed method is one of the possible extensions to explore when dealing with longer inputs (Yoshikane et al., 2002).

6 Conclusions and future work

This paper proposed a novel approach to extract paraphrases of a Japanese noun phrase from a corpus. The proposed method adopts both information retrieval techniques and natural language processing techniques. Unlike past research, the proposed method uses *Kanzi* (ideograms) characters as index terms and retrieves paraphrase candidates in a set of passages. The retrieved candidates are then filtered out based on syntactic and semantic constraints.

The method was evaluated by a test set of 53 noun phrases, and paraphrases were extracted for 46 cases. These paraphrases were evaluated subjectively by three independent judges. The quantitative evaluation suggests that the performance needs to be further improved for fully automatic query expansion in information retrieval, but is usable to help users modify their queries by suggesting possible paraphrases.

From a qualitative point of view, the proposed method could extract paraphrases which cannot be obtained by previous approaches such as syntactic transformation and word substitution. Considering characteristics of Japanese word formation by using character-based indexing enables us to obtain novel paraphrases.

The performance of the current system needs to be improved for fully automatic paraphrasing. One direction is introducing more precise filtering criteria. The current system adopts only dependency analysis of *bunsetu*. We need case analysis as well, to capture relations among the *bunsetu*. Integrating syntactic transformation into the proposed method is another research direction to explore.

In this paper, we evaluated output paraphrases subjectively. Task oriented evaluation should be also conducted. For example, effectiveness of phrase expansion in information retrieval systems should be investigated.

Q	Len.	#Para.	(1)	(2-a)	(2-b)	(2-c)	(3)	Acc.	Prec.
3	1	17	0 (0.00)	7 (0.86)	0 (0.00)	15 (0.60)	29 (0.83)	0.14	0.33
4	1	60	1 (0.00)	61 (0.74)	2 (0.50)	38 (0.47)	78 (0.69)	0.34	0.33
5	1	68	4 (0.75)	8 (0.62)	16 (0.00)	56 (0.14)	120 (0.62)	0.06	0.13
6	1	81	0 (0.00)	0 (0.00)	3 (0.33)	2 (0.00)	238 (0.99)	0.00	0.00
7	2	61	5 (0.60)	20 (0.70)	44 (0.45)	58 (0.66)	56 (0.73)	0.14	0.24
8	1	93	3 (0.00)	22 (0.68)	11 (0.64)	24 (0.42)	218 (0.91)	0.09	0.21
9	2	64	4 (0.75)	6 (0.67)	2 (0.50)	3 (0.33)	177 (0.99)	0.05	0.07
10	3	68	24 (0.42)	37 (0.22)	14 (0.50)	83 (0.41)	45 (0.29)	0.30	0.29
11	2	68	0 (0.00)	12 (0.08)	9 (0.44)	20 (0.25)	163 (0.83)	0.06	0.08
12	2	53	7 (0.14)	54 (0.76)	1 (0.00)	60 (0.37)	37 (0.19)	0.38	0.38
13	2	89	22 (0.32)	23 (0.30)	3 (1.00)	9 (0.11)	210 (0.98)	0.17	0.24
14	3	62	13 (0.85)	0 (0.00)	16 (0.44)	8 (0.12)	149 (0.92)	0.07	0.06
15	3	77	41 (0.49)	18 (0.44)	7 (0.57)	32 (0.38)	133 (0.89)	0.26	0.29
18	2	76	13 (0.08)	18 (0.28)	9 (0.56)	55 (0.42)	133 (0.80)	0.14	0.21
20	3	51	11 (0.82)	19 (0.95)	14 (0.71)	29 (0.62)	80 (0.82)	0.20	0.20
21	2	50	0 (0.00)	4 (0.75)	3 (0.33)	0 (0.00)	143 (0.98)	0.03	0.04
22	3	70	18 (0.72)	7 (0.00)	2 (0.50)	14 (0.36)	169 (0.94)	0.12	0.16
24	3	64	8 (0.88)	1 (0.00)	3 (1.00)	1 (0.00)	179 (0.99)	0.05	0.04
26	4	58	2 (0.50)	22 (0.18)	1 (0.00)	22 (0.27)	127 (0.78)	0.14	0.13
27	6	13	1 (0.00)	7 (0.00)	0 (0.00)	0 (0.00)	31 (0.77)	0.21	0.30
28	4	56	20 (0.25)	8 (0.38)	3 (0.33)	53 (0.30)	83 (0.54)	0.17	0.22
29	6	34	0 (0.00)	3 (1.00)	0 (0.00)	1 (0.00)	97 (0.98)	0.03	0.25
30	4	16	0 (0.00)	12 (0.33)	1 (0.00)	7 (0.14)	28 (0.64)	0.25	0.27
31	6	4	0 (0.00)	0 (0.00)	0 (0.00)	0 (0.00)	12 (1.00)	0.00	0.00
32	4	60	15 (0.80)	19 (0.58)	4 (0.00)	31 (0.39)	111 (0.84)	0.19	0.24
33	4	67	15 (0.60)	58 (0.83)	2 (0.50)	20 (0.65)	105 (0.94)	0.36	0.51
34	4	54	1 (0.00)	12 (0.67)	0 (0.00)	7 (0.57)	142 (0.99)	0.08	0.19
36	7	13	0 (0.00)	1 (0.00)	0 (0.00)	1 (0.00)	37 (0.97)	0.03	0.06
37	5	7	1 (0.00)	1 (0.00)	0 (0.00)	1 (0.00)	18 (0.89)	0.10	0.22
38	5	64	2 (0.50)	1 (0.00)	6 (1.00)	8 (0.38)	175 (0.97)	0.02	1.00
39	4	59	2 (0.50)	4 (0.00)	0 (0.00)	9 (0.56)	162 (0.97)	0.03	0.04
40	4	54	0 (0.00)	11 (0.55)	30 (0.10)	2 (0.50)	119 (0.76)	0.07	0.09
41	5	51	0 (0.00)	4 (0.50)	4 (0.00)	2 (0.00)	143 (0.97)	0.03	0.07
43	5	65	1 (0.00)	1 (0.00)	4 (0.00)	5 (0.20)	184 (0.95)	0.01	0.01
44	7	54	3 (1.00)	0 (0.00)	34 (0.35)	3 (0.00)	122 (0.81)	0.02	0.03
45	6	7	0 (0.00)	0 (0.00)	0 (0.00)	0 (0.00)	21 (1.00)	0.00	0.00
46	7	1	0 (0.00)	0 (0.00)	0 (0.00)	0 (0.00)	3 (1.00)	0.00	0.00
47	9	5	0 (0.00)	0 (0.00)	0 (0.00)	1 (0.00)	14 (0.93)	0.00	0.00
48	7	10	1 (0.00)	1 (0.00)	3 (0.00)	3 (0.00)	22 (0.86)	0.07	0.21
49	8	1	0 (0.00)	0 (0.00)	0 (0.00)	0 (0.00)	3 (1.00)	0.00	0.00
50	8	58	1 (0.00)	1 (0.00)	2 (0.00)	3 (0.00)	167 (0.97)	0.01	0.06
51	6	18	1 (0.00)	13 (0.92)	1 (0.00)	9 (0.78)	30 (1.00)	0.26	0.33
52	7	21	4 (0.00)	1 (0.00)	1 (0.00)	1 (0.00)	56 (0.95)	0.08	0.13
55	7	26	2 (0.00)	1 (0.00)	0 (0.00)	4 (0.00)	71 (0.96)	0.04	0.03
59	10	21	0 (0.00)	0 (0.00)	0 (0.00)	4 (0.50)	59 (0.97)	0.00	0.00
60	12	2	0 (0.00)	0 (0.00)	0 (0.00)	0 (0.00)	6 (1.00)	0.00	0.00
Ave.	4.5	45	5.35 (0.24)	10.8 (0.30)	5.54 (0.23)	15.3 (0.24)	97.9 (0.87)	0.10	0.17

Table 1: Summary of judgment

References

R. Baeza-Yates and B. Riberto-Neto. 1999. *Modern Information Retrieval*. Addison Wesley.

R. Barzilay and L. Lee. 2002. Bootstrapping lexical choice via multiple-sequence alignment. In *Proceedings of 2002 Conference on Empirical Methods in Natural Language Processing*, pages 164–171.

R. Barzilay and K. R. McKeown. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 50–57.

C. Ebert, L. Shalom, G. Howard, and N. Nicolas. 2001. Generating full paraphrases of fragments in a dialogue interpretation. In *Proceedings of the 2nd SIGdial Workshop on Discourse and Dialouge*.

IPA. 2003. GETA: Generic Engine for Transposable Association. `http://geta.ex.nii.ac.jp`.

C. Jacquemin, J. L. Klavans, and E. Tzoukermann. 1997. Expansion of multi-word terms for indexing and retrieval using morphology and syntax. In *Proceedings of 35th Annual Meeting of the Assosiation for Computational Linguistics*.

C. Jacquemin. 1999. Syntagmatic and paradigmatic representation of term variation. In *Proceedings of 37th Annual Meeting of the Assosiation for Computational Linguistics*, pages 341–348.

B. Katz. 1997. Annotating the world wide web using natural language. In *Proceedings of "Computer-assisted information searching on Internet" (RIAO '97)*, pages 136–155.

T. Kitani, Y. Ogawa, T. Ishikawa, H. Kimoto, I. Keshi, J. Toyoura, T. Fukushima, K. Matsui, Y. Ueda, T. Sakai, T. Tokunaga, H. Tsuruoka, H. Nakawatase, and T. Agata. 1998. Lessons from BMIR-J2: A test collection for Japanese IR systems. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 345–346.

S. Kurohashi and M. Nagao. 1998. Building a Japanese parsed corpus while improving the parsing system. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, pages 719–724.

D. D. Lewis. 1992. An evaluation of phrasal and clustered representations of a text categorization task. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 37–50.

H. P. Luhn. 1957. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1(4):390–317.

T. Mitamura. 2001. Automatic rewriting for controlled language translation. In *The Sixth Natural Language Processing Pacific Rim Symposium (NLPRS2001) Post-Conference Workshop, Automatic Paraphrasing: Theories and Applications*, pages ?–?

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

M. Shimohata and E. Sumita. 2002. Automatic paraphrasing based on parallel corpus for normalization. In *Third International Conference on Language Resources and Evaluation*, pages 453–457.

Y. Shinyama, S. Sekine, K. Sudo, and R. Grishman. 2002. Automatic paraphrase acquisition from news articles. In *Proceedings of Human Language Technology Conference (HLT2002)*, pages 40–46.

K. Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

T. Tokunaga, K. Kenji, H. Ogibayashi, and H. Tanaka. 2002. Selecting effective index terms using a decision tree. *Natural Language Engineering*, 8(2-3):193–207.

K. Torisawa. 2001. A nearly unsupervised learning method for automatic paraphrasing of japanese noun phrase. In *The Sixth Natural Language Processing Pacific Rim Symposium (NLPRS2001) Post-Conference Workshop, Automatic Paraphrasing: Theories and Applications*, pages 63–72.

J. Uebersax. 2001. Statistical methods for rater agreement. `http://ourworld.compuserve.com/homepages/jsuebersax/agree.htm`.

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of 33rd Annual Meeting of the Assosiation for Computational Linguistics*, pages 189–196.

F. Yoshikane, K. Tsuji, K. Kageura, , and C. Jacquemin. 2002. Detecting Japanese term variation in textual corpus. In *Proceedings of 4th International Workshop on Information Retrieval with Asian Languages (IRAL'99)*, pages 164–171.