Automatic Term Recognition based on Statistics of Compound Nouns

To be appeared on Terminology

Hiroshi Nakagawa

Information Technology Center, The University of Tokyo

7-3-1 Hongo, Bunkyo, Tokyo, 113-0033, Japan

The NTCIR1 TMREC group called for participation of the term recognition task which is a part of NTCIR1 held in 1999. As an activity of TMREC, they have provided us with the test collection of the term recognition task. The goal of this task is to automatically recognize and extract terms from the text corpus which consists of 1,870 abstracts gathered from the NACSIS Academic Conference Database. This paper describes the term extraction method we have proposed to extract terms consisting of simple and compound nouns and the experimental evaluation of the proposed method with this NTCIR TMREC test collection. The basic idea of scoring a simple noun :N of our term extraction method is to count how many nouns are conjoined with N to make compound nouns. Then we extend this score to measure the score of compound nouns because most of technical terms are compound nouns. Our method has a parameter to tune the degree of preference either for longer compound nouns or for shorter compound nouns. As for term candidates, in addition to noun sequences, we may add variations such as patterns of “A NO B” that roughly means “B of A” or “A's B” and/or “A NA B” where “A NA” is an adjective. Experimental results of our method are promising, namely recall of 0.83, precision of 0.46 and F-value of 0.59 for exactly matched extracted terms when we take into account top scoring 16,000 extracted terms.

1. Introduction

As widely known, automatic term extraction is definitely useful in various applications including (1) automatic extraction of back-of-book indices from a volume of text, (2) terminology extraction of one academic field from corpora of that academic field and (3) keywords extraction from documents for IR purposes. Especially (1) and (2) have so far been done by hand in practice at huge cost and/or intolerably long processing time. Therefore, an automatic term extraction technology would be a great help for these purposes. In this situation, the TMREC task which is a part of NTCIR1 really fits well the research into high accuracy term extraction methods. The task is not only calling for good term extraction methods but also for providing the researchers of this field with a test collection which cannot be built by an individual small research group. In fact, we applied our term extraction method to the test collection after the NTCIR1 conference and describe the results in this paper. Honestly speaking, it is a sort of afterthought. Nevertheless we present it here because we would like to focus on the nature of correct terms in the test collection. Another reason is that our method does not include parameters that can only be tuned up by machine learning with a test data set. Our experimentation shows our method to be potentially very powerful.

In this paper, section 2 gives the background of term extraction. Section 3 describes the proposed method of term extraction, including the term candidate extraction method in 3.1 and the term scoring scheme in 3.2. The experimental results are discussed in section 4.

2. Background

2.1. *Categories of Terms*

We have to think about at least three categories of terms, namely (1) keywords for IR, (2) index words, and (3) terminologies. (1) is discussed in IR, and recently every free keyword is given its own weight such as tf·idf. However, this category is out of our scope here. Category (3) is what we are about to discuss in this paper. While (2) has got little attention so far in NLP research, except for (Schutze 1998:101-104), the methodology developed for extracting and weighting terms of (1) or (3) might be and can be applied for terms of category (2). Presumably the purpose of TMREC is to develop the term extraction method for the purpose of (3) (Kageura 1999:411-440).

As for the purpose of (3), Kageura and Umino (1996:259-289) refer to two essential aspects of the nature of terms, namely unithood and termhood.

Unithood refers to the degree of strength or stability of syntagmatic combinations or collocations. For instance, a word has very solid unithood. Other linguistic units having strong unithood are compound words, collocations, and so forth.

Termhood refers to the degree that a linguistic unit is related to a domain-specific concept. Termhood is usually calculated based on term frequency and bias of frequency (so called Inverse Document Frequency). Even though these calculations give a good approximation of termhood, still they do not directly reflect termhood because these calculations are based on superficial statistics. This is not a meaning in a writer's mind but a meaning in use. Apparently, termhood is intended to reflect the former type of meaning.

These two notions are fundamental and essential guidelines for devising algorithms for extracting terms from corpora.

2.2. *Standard Procedure of Automatic Term Extraction*

In general an automatic term extraction procedure consists of two procedures in general. The first one is a procedure of extracting term candidates from corpora. The second procedure is to assign each term candidate extracted in the first procedure a score that indicates how likely the term candidate is a term we would like to extract. Once we assign a score to each term candidate, they are all ranked according to their scores. The remaining problem is how many terms we have to select from these ranked term candidates. It depends on various factors including how many terms we actually want to extract, who uses these extracted terms for what purpose, and so forth. Considering these factors, at this moment, this selection problem is not yet scientifically well defined. In other words the formalization of this problem is still open to us. Thus, in the rest of this subsection, we describe the background of a candidate extraction procedure and a scoring procedure respectively.

2.2.1. *Candidates Extraction*

Whichever aspect, namely unithood or termhood, we may rely on to extract terms, the first thing to do is to extract *term candidates* from the given text corpus. There are two major types of term candidates in terms of linguistic structure. One is an N-gram of characters such as (Fujii and Croft 1993:237-246) in Japanese or (Lam et al. 1997:68-80) in Chinese. However, an N-gram apparently does not match a term that is the target of the TMREC task. The other is a structure based on words, more precisely a simple word or a compound word, which is exactly the target of the TMREC task. Thus, in the rest of this paper we pay attention to the latter.

Term candidates that consist of words are nouns or compound nouns. To extract compound nouns as promising term candidates and at the same time to exclude undesirable strings such as "*is a*" or "*of the*" the most frequently used method is to filter out the words that are in the so called *stop-word-list*. In addition to a stop-word-list, more complex structures like noun phrases, collocations consisting of noun, verb, preposition, determiner and so on, become focused on (Smadja and McKeown 1990:252-259; Frantzi and Ananiadou 1996:41-46, Zhai and Evans 1996:17-23; Hisamitsu and Nitta 1996:550-555, Shimohata et al. 1997:476-481). All of these are good term candidates in a document or a corpus of a specific domain because all of them have a strong unithood. Needless to say, for terms of complex structure like compound words or collocations, we make the following basic assumption:

Assumption *Terms of complex structure are to be made of existing simple terms.*

The structure of complex terms is another important factor for automatic term extraction. It is expressed syntactically or semantically. As a syntactic structure, dependency structures that are the result of parsing a noun phrase are focused on in many works. Of course, we need heuristics or theories based on statistics to select plausible dependency as described in (Zhai and Evans 1996:17-23). Since we focus on these complex structures, the first task in extracting term candidates is morphological analysis including part of speech (POS) tagging. In English, POS tagging has been one of the main issues of natural language processing, i.e. (Brill 1994a: 722-727), and high quality POS taggers such as (Brill 1994b) have already been widely used. In Japanese, which is an agglutinative language, morphological analysis segments out words from a sentence and does POS tagging simultaneously (Matsumoto et al. 1996).

After POS tagging, the complex structure mentioned above is extracted as a term candidate. Previous studies have proposed many promising ways for this purpose. Zhai and Evans (1996:17-23) focus on noun phrases. Ananiadou (1994:1034-1038) proposes a method to extract word compounds as terms. Hisamitsu and Nitta (1996:550-555) and Nakagawa (1997:598-611) concentrate their efforts on compound nouns. Smadja and McKeown (1990:252-259), Daille et al. (1994:515-521), Frantzi and Ananiadou (1996:41-46) and Shimohata et al. (1997:476-481) try to treat more general structures like collocations. As described in the later section, we focus on four types of extended compound noun that may include Japanese NO particles and/or Japanese NA adjectives in our experimentation.

2.2.2. *Scoring*

Once term candidates have been extracted from a text corpus, the next thing we should do is to assign a certain score to each term candidate in order to rank them in descending order of termhood. The question here is what sort of term candidates should be ranked high. According to unithood and termhood described in 2.1, we have a dichotomy of scoring methods, namely scoring based on unithood and that based on termhood. Obviously, terms with high termhood should get a high score. However, to directly measure termhood of the given term candidate is extremely difficult because only the writer of a document knows which terms are qualified as necessary and sufficient terms in the given documents. Many researchers have tried to work out the definition of term candidate’s score which approximates termhood. In fact, many of those proposals are calculated on the basis of surface statistics like tf·idf. Ananiadou et al. proposed C-value (Frantzi and Ananiadou 1996:41-46) and NC-value (Frantzi and Ananiadou 1999:145-179) which count how independently the given compound noun is used in the given corpus. In this sense, their proposal is basically a unithood-based method. Hisamitsu and Niwa (1999: 475-484) and Hisamitsu (2000: 320-326) propose a way to measure termhood which counts how far the given term is different from the average of term distribution. Basically their measures use word co-occurrence, log-likelihood ratio of distribution. Kageura et al. (2000: 397-403) use the frequency with which a Japanese word and its English counterpart co-occur in a bi-lingual document set. All of them try to capture how important and independent a writer regards and uses individual terms in a corpus. In our experiment, we basically employ the method proposed in (Nakagawa 1997:598-611) with some improvements as described in 3.2.

3. Term Extraction Method

3.1. *Candidates Extraction*

In our experiment, a process of extracting term candidates is carried out as follows. We use the corpus morphologically analyzed and POS tagged by the NTCIR1 TMREC group. From this POS tagged text, we extract term candidates. In our experimentation, we extract and compare the following four types of term candidates.

Noun sequence: NS

This is an uninterrupted sequence of nouns, which henceforth we call NS. For instance, YOUGO ZISHO (terminology dictionary) is of this type. In Japanese we usually and consistently create new terms by combining already known nouns, sometimes ending up with very long terms like NITI-EI TAIYAKU YOUGO ZISHO TYUUSHUTU HYOUKA HOUHOU (Japanese-English translation dictionary extraction evaluation method). This type of noun sequence, or more precisely compound noun, constitutes the majority of Japanese technical terms that are focused on in the TMREC Task. Despite the strong expressive power of this type of noun sequence structure, it is still too simple to express very complicated meanings with complex dependency relations among the component words of a compound noun. Thus we often introduce the following structures.

Noun sequence including NO particles: NS+NO

Although the English counterpart of the Japanese particle NO is generally said to be “of”, NO has a wider range of meanings. For instance, BYOUKI NO INU (a dog with disease) or HARE NO HI (a fine day) cannot be expressed with “of” like “a dog of disease” or “a day of fine.” Since we expect NO to be widely used to make terms of compound nouns because of its broad coverage of meaning, we extract noun sequences that may include NO particles as term candidates. Hereafter we call this type NS+NO.

Noun sequence including NA adjectives: NS+NA

In English, complicated notions are often expressed as “adjective + noun” e.g. “finite automata.” This is, to some extent, similar to Japanese if we focus on NA adjectives. A NA adjective is a kind of adjective whose postfix is NA . Actually its root form is XXX-DA, and XXX- NA is one of its inflectional forms to be followed by a noun or a noun phrase. For example, KEIKENTEKI-NA TISHIKI (empirical knowledge) could be used as a term. Then this type, which we call NS+NA, consists of a noun sequence that may include NA adjectives.

Noun sequence including NO particles and NA adjectives: NS+NO+NA

Obviously, the broadest type so far is a noun sequence that may include NO particles and/or NA adjectives. We call this type NS+NO+NA.

The problem here is how widely and generally types NS+NO, NS+NA and NS+NO+NA contribute to make technical terms. From another point of view, we are very interested in the nature of the correct terms registered in the NTCIR TMREC test collection, namely how many terms of these three types are included in it. But it is an empirical issue and will be discussed later.

3.2. *Scoring*

In this section, we focus on the scoring method we adopt for assigning a score to each term candidate extracted by the methods described in 3.1. Obviously, the relation between a simple term and complex terms which include this simple term is very important. Nevertheless, to my knowledge, this relation has not been paid enough attention to so far. Nakagawa (1997:598-611) shows a new direction because it proposed a term scoring method which utilizes this type of relation.

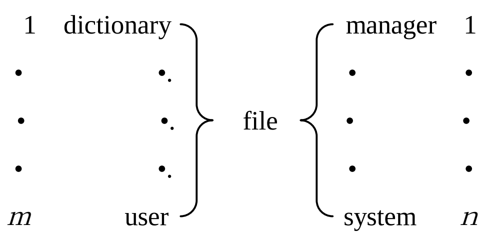
Here we focus on compound nouns among the various types of complex terms. In technical documents, the majority of domain-specific term are noun phrases or compound nouns. In spite of the huge number of technical terms consisting of compound nouns, not many simple nouns contribute to make these compound nouns. This is a basic observation we rely on. Considering this observation, we propose a new scoring method that measures the importance of each simple noun. In a nutshell, this scoring method for a simple noun measures how many distinct compound nouns contain a particular simple noun as their part in a given document or a set of documents. More precisely, we introduce *Pre*(simple noun) and *Post*(simple noun) defined as follows:

Before stating the definitions, we have to note how we treat NO particles and NA adjectives in NS+NO, NS+NA and NS+NO+NA compounds. In NS+NO and NS+NO+NA, NOs are just ignored in calculating *Pre* and *Post*. In NS+NA and NS+NO+NA, XXX-NA is treated as XXX that can always be regarded as a noun. By regarding NO and NA in this way, what we have to consider is only superficially a noun sequence. Now we state the definition of *Pre* and *Post*.

Definition

In the given text corpus, *Pre (N)*, where *N* is a simple noun occurring in the document, is the number of distinct nouns that *N* adjoins and make compound nouns with *N*, and *Post(N)* is the number of distinct nouns that adjoin *N* and make compound nouns with *N*.

The key point of this definition is that *Pre (N)* and *Post(N)* count not the frequencies of words that are adjacent to *N*, but the number of distinct words that adjoin *N* or that *N* adjoins. That means that *Pre (N)* and *Post(N)* measure not the surface statistics of compound nouns containing *N*, but do measure how the writer of the technical document interprets *N* and its meaning, and uses it in the document. If a certain word, say *W*, expresses the key concept of a certain academic or technical area that the document treats, the writer of the document must use *W* not only many times but also in various ways. For instance, he/she composes quite a few compound nouns that include *W* and uses them in documents he/she writes. Since this kind of usage really reflects the termhood of *W*, *Pre (W)* and *Post(W)* measure directly *W*'s termhood. Figure 1 shows an example of *Pre* and *Post*.



Pre (“file”) = m and *Post*(“file”) = n

Figure 1: An example of *Pre* and *Post*

With these *Pre* and *Post* we assign each term candidate the score which indicates how likely each term candidate is a term we want to extract. However, remember the observation that the majority of terms in academic or technical fields are not simple nouns but compound nouns. Thus, what we should do next is to extend this scoring method to be able to treat compound nouns. For the given compound noun $N_1N_2\ldots N_k$ where N_is are simple nouns, the scores of importance of $N_1N_2\ldots N_k$, which is called $Imp(N_1N_2\ldots N_k)$, might have an infinite number of definitions, even though we define *Imp* with *Pre* and *Post*. Here we use the following definitions because they are very flexible, nonetheless they are very simple.

$$Imp_1(N_1 N_2 \ldots N_k) = \left(\prod_{i=1}^{k} ((Pre(N_i)+1)(Post(N_i)+1))\right)^{\frac{1}{k^a}}$$

$$Imp_2(N_1 N_2 \ldots N_k) = \frac{1}{k^a}\sum_{i=1}^{k}(Pre(N_i)+Post(N_i))$$

Apparently *k* is regarded as the length of a compound noun. A parameter *a* is used to tune how heavily Imp_1 and Imp_2 depend on *k*. If $a=0$, Imp_1 is a product of *Pre* and *Post* of every component simple word, Imp_2 is a sum of *Pre* and *Post* of every component simple word. If $a=1$, then Imp_1 and Imp_2 do not depend on *k* because they are averages of *Pre* and *Post* of every component simple word. If $a>1$ and *k* is greater than one, the bigger the parameter *a* is, the less the value of Imp_1 and Imp_2 becomes. In sum, The smaller *a* becomes, the larger Imp_1 and Imp_2 become for a longer compound noun. That means that the preference for longer or shorter compound nouns can be accomplished by tuning the parameter *a*. In our experiment, which we will describe in the next section, we vary *a* to know *a*'s effect on the quality of our term extraction system.

4. Experimental Evaluation

4.1. *Experimentation System*

Test Collection

As a test collection, we use 1,870 Japanese abstracts of several academic conferences those of which are POS tagged, and 8,843 correct terms in Artificial Intelligence, extracted by hand. This test collection was built by the NTCIR TMREC group at the NTCIR1 Workshop.

The combinations of Term candidates and Scoring

As term candidates, we extract four types of term candidates described above, namely NS, NS+NO, NS+NA and NS+NO+NA, automatically from the 1,870 tagged abstracts. As a scoring method, we use Imp_1 and Imp_2. Then, we combine these four types of term candidates and two scoring methods, ending up with eight combinations. As for the parameter *a*, we calculated the F-measure of each of the above combinations for $a = 0.5, 1, 2, 3, 4, 5, 6$ respectively. Here, F-measure is defined as follows.

$$F - measure = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

Recall, Precision and F-measure are calculated for a certain number of top scored term candidates where a scoring method is either Imp_1 or Imp_2.

4.2. *Results*

Table 1 shows the results of each combination. Each of these results is the result of the best F-measure among the results of various values of *a*. In addition, these results are of the case where we take into account the top scoring 13,000 terms. In Table 1 and 2, the meanings of two symbol characters in the first column are the followings:

The first character	Meaning
A	The value of the parameter *a* with which the data of each row are calculated.
T	Total.
F	Fully matched terms.
I	Result includes term candidates.
P	Result is a part of term candidates.
A	Result is including both I & P

The second character	Meaning
N	Number of extracted terms. Then the third column's TN means the total number of high scored extracted terms.
R	Recall
P	Precision
F	F-measure

Table 1 Experimental Results of each combination

	a	TN	FN	IN	PN	FR	FP	FF	AR	AP	AF
NS,Imp_1	1	13000	6362	4825	429	0.719	0.489	0.582	0.817	0.860	0.838
NS,Imp_2	3	13000	6337	4441	742	0.716	0.487	0.580	0.775	0.829	0.801
NS+NO,Imp_1	5	13000	4767	5805	682	0.539	0.366	0.436	0.594	0.813	0.686
NS+NO,Imp_2	6	13000	4854	5463	699	0.548	0.373	0.444	0.597	0.793	0.681
NS+NA,Imp_1	1	13000	6010	5400	333	0.679	0.462	0.550	0.796	0.877	0.835
NS+NA,Imp_2	3	13000	5977	4931	650	0.675	0.459	0.547	0.747	0.839	0.790
NS+NO+NA,Imp_1	5	13000	4705	5835	666	0.532	0.361	0.430	0.590	0.810	0.683
NS+NO+NA,Imp_2	6	13000	4748	5503	679	0.536	0.365	0.434	0.595	0.788	0.678

The first thing we notice in Table 1 is that Imp_1 and Imp_2 do not show noticeable differences in performance. On the contrary, the effects of NO particles and NA adjectives are more diverse. The results where NO particles are included as term candidates, namely NS+NO+NA and NS+NO, show the poor performance compared to other cases. NS shows the best performance in every aspect that corresponds to each column. We previously thought that a set of terms selected by hand would include a good number of compound nouns including NO or NA. But these results have proved our expectation be false. Let us reflect on NO and NA. As is well known, a NO particle has a variety of, or even too many, meanings. Thus we end up with quite a few undesirable term candidates included in the first selection. A NA adjective seems to be less guilty, but we usually omit NA from “NA adjective + noun” even if NA adjective is possible. For instance, KEIKENTEKI TISIKI (empirical knowledge) is preferred to KEIKENTEKI-NA TISHIKI; nevertheless both of them have exactly the same meaning. As a result, we empirically recognize NS is a good set of term candidates at this level of granularity of term extraction.

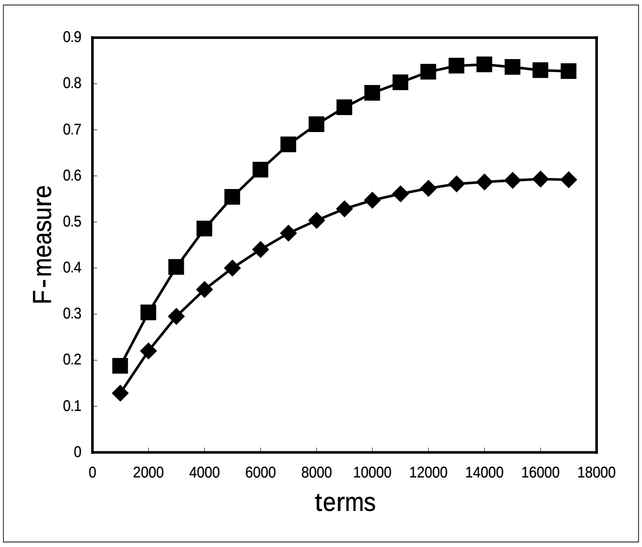
Table 2 shows the comparison of our best result with other NTCIR1 TMREC participants' results. Our result of NS is the best where *a=1* if we take into account the top scoring 16,000 term candidates. The N system is also our result submitted to the NTCIR1 workshop. Since our system N adopted term candidates including the NO particles at the NTCIR1 workshop, the result of N is almost the same as NS+NO in this paper except for term selection process where we used a window system (Nakagawa and

Mori 1998:64-70). However, as known from these results, the window method that is our unique proposal for term selection does not work well in the NTCIR1 TMREC Task. The number of top scored extracted terms of the other two systems that is to say Z and W, are 16,112 that is near to ours and 23,270 respectively. Apparently our method only using NS shows the best performance of fully matched terms at 16,000 extracted terms. In the W system, the number of correct extracted term is the highest because the number of extracted terms is much larger than others. In these comparisons, our method using NS is best in the F-measure that indicates an overall performance. Actually, Z and W showed the two highest performances among the NTCIR1 TMREC Task participants. Therefore, our scoring method per se indicates its great flexibility and potential for term extraction tasks.

Table 2 Comparison of the result of our method with other NTCIR participants' results

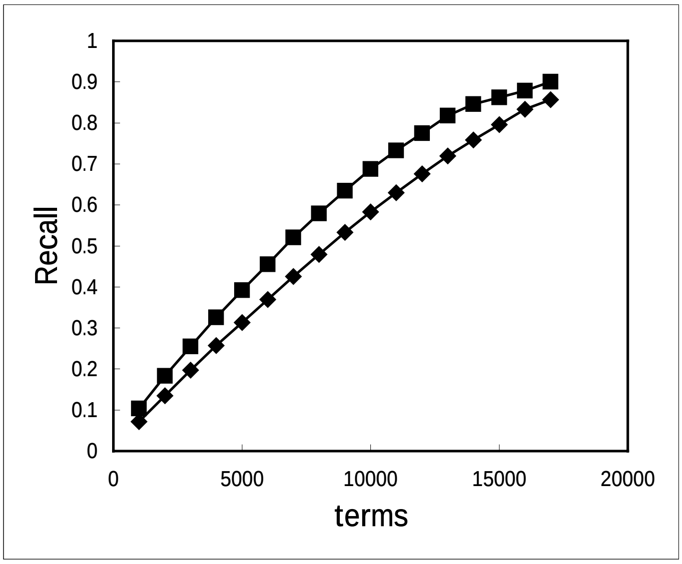
	Total.	FN	IN	PN	FR	FP	FF
Our system	16000	7367	5197	923	0.833	0.460	0.593
N	18608	5554	10484	355	0.629	0.299	0.405
Z	16112	6536	5464	690	0.739	0.405	0.524
W	23270	7944	9432	879	0.899	0.341	0.494

In Figure 1, 2 and 3 show F-measure, recall and precision respectively for the cases of the number of extracted top scoring terms being 1,000 to 17,000 by step of 1,000. The experimental condition of these results is NS, Imp_1 and $a=1$ which shows the best result among every combination appeared in Table 1.



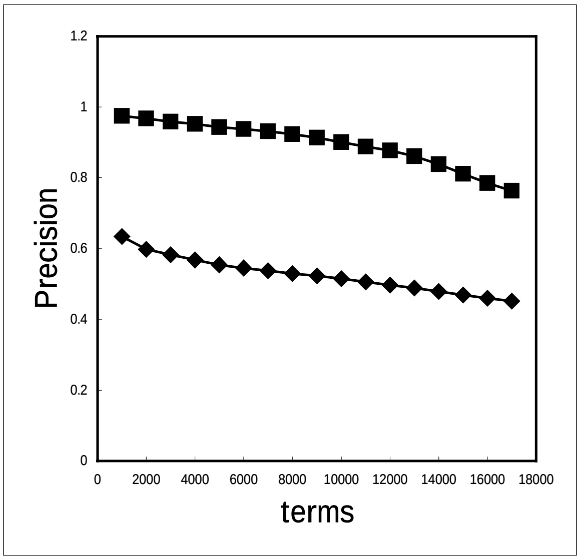
◆: F-measure of fully matched terms, ■: F-measure of fully matched terms and extracted terms including correct terms

Figure 1: F-measures of NS, *Imp*$_1$ and *a=1*



◆: Recall of fully matched terms, ■: Recall of fully matched terms and extracted terms including correct terms

Figure 2: Recall of NS, *Imp*$_1$ and *a=1*

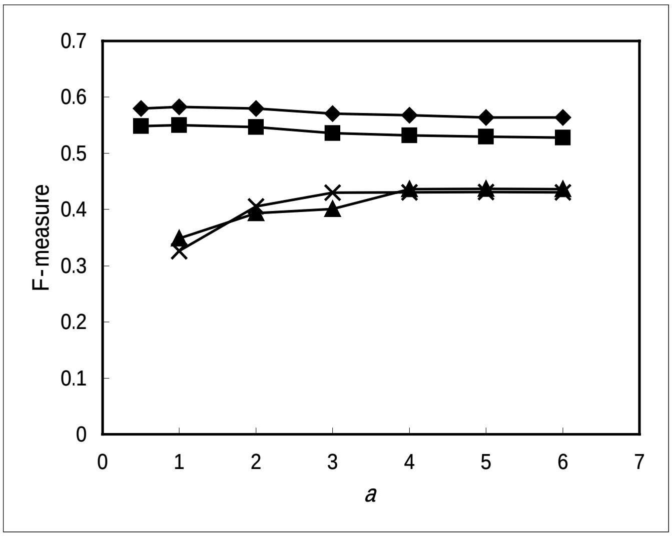


◆: Precision of fully matched terms, ■: Precision of fully matched terms and extracted terms including correct terms

Figure 3: Precision of NS, *Imp*$_1$ and *a=1*

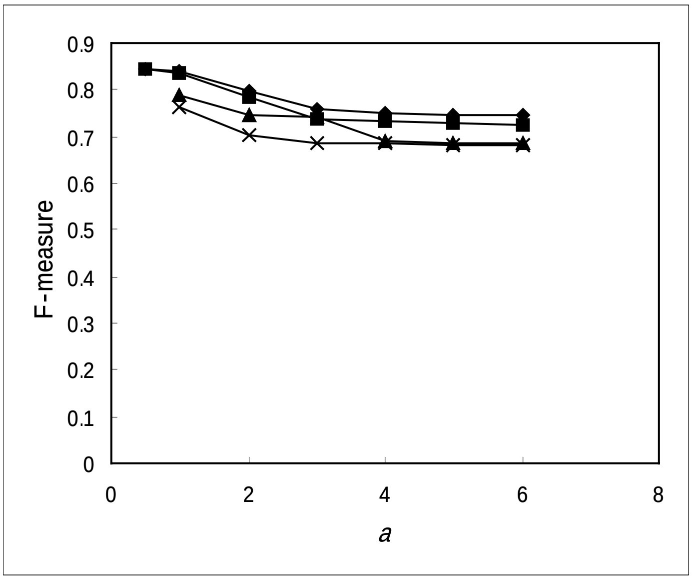
As known from Figure 1, the best F-measure of fully matched terms, which we call F-case, is 0.593 at 16,000 extracted terms, and the F-measure of fully matched terms, including correct terms, which we call

B-case, is 0.892 at 14,000 extracted terms. The F-measure of F-case is almost saturating at 14,000 extracted terms and is very slightly improving after that point. The F-measure of B-case is even decreasing after the highest point of 14,000 extracted terms. These are the results of the combination of the degree of increasing recall shown in Figure 2 and the degree of decreasing precision shown in Figure 3. Especially B-case shows that after the point of 14,000 extracted terms, the rate of extracted B-case terms declines. Figure 4 and 5 show the relation between F-measure and *a* of F-case and B-case respectively.



◆:NS ■:NS+NA ▲:NS+NO X:NS+NO+NA

Figure 4: Relation between *a* and F-measure of F-case



◆:NS ■:NS+NA ▲:NS+NO X:NS+NO+NA

Figure 5: Relation between *a* and F-measure of B-case

What we know from Figure 4 is the following: F-measures of NS+NO and NS+NO+NA of F-case increase as *a* increases. In other words, as for NS+NO and NS+NO+NA type extracted terms, shorter compound nouns are preferred as stated in 3.2. It is obvious that shorter compound nouns tend not to include NO particles. Thus, the correct terms in the NTCIR1 TMREC test collection do not include many compound nouns that contain NO particles. This is our new finding about the characterization of terms extracted by hand in this test collection. In fact, NO particles have too many meanings, some of which are not suitable for technical terms like TISHIKI NO HUSOKU (shortage of knowledge). NS and NS+NA show the opposite tendency even though this tendency is slight. That means that if we only focus on NS or NS+NA, the longer compound nouns are more or at least equally preferred to shorter ones. Thus we infer that very long compound nouns including NA can more frequently be technical terms than compound nouns including NO particles. Looking more carefully at the correct terms in the test collection, very few compound nouns that contain NO and/or NA are correct terms of the test collection. Actually they are approximately about 400 terms. Our results for the four types of term candidates are consistent with this character of the test collection. If we focus on B-case, longer compound nouns are preferred because parts of NS+NO, NS+NA and NS+NO+NA are apparently NS type terms. In other words, small *a* is expected to show a better F-measure. This expectation is exactly what we see in Figure 5.

Finally, our method of the best performance does not extract 875 correct terms. To introduce new ideas into the whole extracting system for extracting these 875 terms without extracting non-correct terms is to be our future problem.

5. Conclusions

As a scoring method for extracting technical terms in the form of compound nouns, we propose Imp_1 and Imp_2 functions that are based on *Pre(N)* and *Post(N)* scores of simple nouns. These scores count in how many compound nouns the simple noun *N* is included as a component. This type of score is sure to reflect how important the writer of a text regards *N* in the specific academic or technical area. In this sense, in our method we pay attention to a way of human thinking and human attitude towards terminology. This score shows quite good performance for the NTCIR1 TMREC Task. We also found that the majority of terms extracted by hand by the NTCIR1 TMREC group are compound nouns containing neither NO particles nor NA adjectives.

The scoring method we propose is quite simple and powerful. Thus there is a room to improve this scoring method. One of the directions is to tune up our term candidate extraction method and scoring method for corpora of a specific academic domain because characteristics of texts might change from one academic to another.

References

Ananiadou, S. 1994. "A methodology for automatic term recognition". In *Proceedings of 15th International Conference on Computational Linguistics*, 1034-1038.

Brill, E. 1994a. "Some advances in transformation-based part-of-speech tagging". In *Proceedings of 11th National Conference on Artificial Intelligence*, 722-727.

Brill, E. 1994b. "Supervised part of speech tagger". *http://www.cs.jhu.edu/~brill/*.

Daille, B., Gaussier, E. and Lange, J.M. 1994. "Towards automatic extraction of monolingual and bilingual terminology". In *Proceedings of 15th International Conference on Computational Linguistics*, 515-521.

Dunning, T. 1993. “ Accurate methods for the statistics of surprise and coincidence”. *Computational Linguistics* 19(1), 62-74.

Frantzi, T.K. and Ananiadou, S. 1996. “Extracting nested collocations”. In *Proceedings of 16th International Conference on Computational Linguistics*, 41-46.

Frantzi, T.K. and Ananiadou, S. 1999. “The c-value/nc-value method for atr”. *Journal of Natural Language Processing* 6(3), 145-179.

Fujii, H. and Croft, W.B. 1993. “A comparison of indexing techniques for Japanese text retrieval”. In *Proceedings of 16th International Conference on Research and Development in Information Retrieval*, 237-246.

Hisamitsu, T. and Nitta, Y. 1996. “Analysis of Japanese compound nouns by direct text scanning”. In *Proceedings of 15th International Conference on Computational Linguistics*, 550-555.

Hisamitsu, T. and Niwa, Y. 1999. “Term Extraction Using a New Measure of Term Representativeness”. In *Proceedings of the First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, 475-484.

Hisamitsu, T, 2000. “A Method of Measuring Term Representativeness”. In *Proceedings of 18th International Conference on Computational Linguistics*, 320-326.

Kageura, K. and Umino, B. 1996. “Methods of automatic term recognition: a review”. *Terminology* 3(2), 259-289.

Kageura, K, Tsuji, K. and Aizawa, A, 2000. “Automatic Thesaurus Generation through Multiple Filtering”. In *Proceedings of 18th International Conference on Computational Linguistics*, 397-403.

Kageura, K. 1999. “TMREC Task: Overview and Evaluation”. In *Proceedings of the First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, 411-440.

Lam, W., Wong, C.Y. and Wong, K.F. 1997. “Performance evaluation of character, word and n-gram-based indexing for Chinese text retrieval”. In *Proceedings of the Second International Workshop on Information Retrieval With Asian Languages*, 68-80.

Matsumoto, Y., Kurohashi, S., Yamaji, O., Taeki, H. and Nagao, M. 1996. *Instruction Manual of Japanese Morphological Analyzer JUMAN3.1*. Nagao Lab. at Kyoto University.

Nakagawa, H. 1997. "Extraction of index words from manuals". In *Conference Proceedings of Computer-Assisted Information Searching on Internet*, 598-611.

Nakagawa, H. and Mori, T. 1998. "Nested Collocation and Compound Noun for Term Recognition". In *Proceedings of the First Workshop on Computational Terminology COMPTERM'98*, 64-70.

Schtuze, H. 1998. "The Hypertext Concordance: A Better Back-of-the-Book Index". In *Proceedings of the First Workshop on Computational Terminology COMPTERM'98*, 101-104.

Shimohata, S., Sugio, T. and Nagata, J. 1997. "Retrieving collocations by co-occurrences and word order constraints". In *Proceedings of 35th Annual Meetings of the Association for Computational Linguistics*, 476-481.

Smadja, F.A. and Mckeown, K.R. 1990. "Automatically extracting and representing collocations for language generation". In *Proceedings of the 28th Annual Meetings of the Association for Computational Linguistics*, 252-259.

Zhai, C. and Evans, D.A. 1996. "Noun-phrase analysis in unrestricted text for information retrieval". In *Proceedings of 34th Annual Meetings of the Association for Computational Linguistics*, 17-23.