Library and Information Science No. 34 1995

An Automatic Indexing of Compound Words based on Mutual Information for Korean Text Retrieval

韓国語テキスト検索のための相互情報に基づく複合語の自動索引

Pan Koo Kim

要　　旨

最近，自然語テキスト表現に対する索引語として単語と複合語が広く利用されてきている。本稿では，膠着語，特に韓国語に適した複合語の自動索引手法を提案している。まず，索引語としての複合語を組み立てるための条件を述べ，次に，テキスト全体から連続する名詞群に適用できる分解規則を示した。最後に，情報理論に基づき複合語における語の連合の程度を算出する，語の利用可能性を測定する一つの尺度として相互情報を提案した。この方法を当てはめた結果，複合語の語の精度は72%から87%に向上した。

Pan Koo Kim: Department of Computer Science, Chosun University, 375 Seosuk-Dong Dong-Ku KwangJu 501-759 Korea
E-mail: pkkim@mina.chosun.ac.kr, or pkkim@ssrnet.snu.ac.kr
1995年9月29日受付

I. Introduction

Information retrieval is useful in locating the relevant texts in a large text database easily. In addition, a user can build his own text database. A user-inserted document is not modified, but indexed and tagged automatically by the system to make later retrieval efficient. Automatic indexing is the process of algorithmically examining the documents to generate lists of index terms. The efficiency of an information retrieval system depends on how well the indexing scheme performs[1)~3), 8), 9)].

It is customary to use single and compound words as indexing terms to represent the contents of natural language text[1)~3), 9), 14), 15)]. Indexing method of single words is relatively simple, however, they tend to retrieve unnecessary texts. Normally single word cannot distinguish relevant from irrelevant documents. On the other hand, compound word which is consisting of sequences of related text words carries more specific meaning than single word included in the compound word. For example, "information storage" or "information retrieval" is more specific than "information". Compound words retrieve relatively fewer documents, however, most of the retrieved documents seem to be helpful to users[2), 13)].

The Korean language has some grammatical characteristics different from inflective languages, specifically the English language. Firstly, Korean has the SOV (Subject-Object-Verb) order. It is the same type of language as Japanese, Turkish, or Mongolian. Secondly, Korean is a postpositional language that attaches a postpositional word functioning as an auxiliary to the end of the substantives or declinable word-root. This is opposed to English that is a prepositional language. We call this postpositional language an agglutinative language. In agglutinative language, this postpositional word is attached to the end of the substantives and normally plays an important role in indicating the case of the substantives in a sentence. For example,

(Kim-un 10-il Seoul-ul tenatta.)
↓ ↓ ↓ ↓
〈Kim on the 10th Seoul left.〉
(*Kim left Seoul on the 10th.*), in English

Here, "*un*" and "*ul*" are postpositions. Other postpositions in Korean are "*eyse*", "*lo*", "*ka*", "*i*", "*nun*", "*lul*", etc., and in Japanese: "*no*", "*ha*", "*ka*", "*o*", etc. By using these postpositions we can determine the category of case in a sentence. That is, in the above example, "*un*" enables us to determine the case of "*Kim*", and it is usually used to represent the case of nominative (in Japanese, "*ha*"). And also, "*ul*" is for the case of accusative (in Japanese, "*o*"). Another characteristics of Korean is that the postposition representing the case of genitive, nominative, or accusative can be omitted. Because of these characteristics, there are some difficulties in applying indexing techniques already developed for inflective language, specifically English to agglutinative languages. So, we need some technique to extract compound words, different from English but suitable to Korean.

We usually construct compound words by concatenating successive words (simple-type). However, if we construct only simple-type compound words, it is impossible (1) to represent text content thoroughly, and (2) to match the query terms with document terms, because we cannot recognize compound words (complex-type) that is different in form but are semantically similar enough. Therefore we need to extract keywords thoroughly to represent the text content. To do this, the normalization of compound words for English texts has been already done in Fagan[2)]. In Fagan[2)], normalization and decomposition techniques for generating compound words have been presented. Different from the methods for inflective language, this paper presents construction conditions and decomposition rules to construct all compound words from Korean texts, by performing not the full analysis but partial analysis of syntax as well as morphological analysis.

In order to select relevant keywords from candidate keywords, we can use the weighting

method, self information, based on information theoretic notion, which has been used to derive a measure of term usefulness for indexing purposes[13]. Because occurrences of compound words are less frequent than those of single words, it is not proper to estimate the weight of compound words with self information based on frequency of terms. This fact calls for some other weighting measures. In this paper, we newly propose another weighting method, mutual information, to measure the weight of compound words for selecting relevant compound words. This method is described in IV.

II. Automatic Indexing

A. General issues

An automatic indexing normally involves the process of algorithmically examining the documents to generate lists of index terms in many practical areas. Manual document indexing is labor-intensive and time consuming work, and has the drawbacks of lack of uniformity and indexer-user mismatch. In contrast, automatic indexing has advantage of bias-free uniformity and efficiency. Furthermore, studies show that automatic indexing for English texts has approximately the same retrieval performance as manual indexing[13]. For the case of Korean text, there are some studies about the effectiveness of several automatic indexing techniques[8), 9)].

Our method uses not the full syntactic analysis but the partial analysis, because it is difficult to analyze syntax structure correctly in a sentence by the current technology. So, we analyze the sentence to the extent that we can recognize whether the phrases are compound words or not. A full syntactic approach might be helpful to select keywords more precisely. However, syntactic analyses have several problems. Firstly, it is difficult to analyze sentence correctly, so the result may include errors. Next, because existing parsers are usually very complicated and large program, they are too bulky to adopt into the automatic indexing system. Another one is that syntactic parsers require large dictionaries, which are generally difficult to prepare and maintain[14]. On the contrary, our system which uses much simpler syntax analysis system can be executed on PC DOS without serious degrading of the performance.

B. Keyword candidate extraction

Automatic indexing is usually proceeded by the following steps[5]: (1) take document, (2) do morphological analysis for each words, (3) exclude stopwords and function words, (4) extract single words as candidate keywords, (5) after checking if conditions for compound word are satisfied, construct and decompose it, (6) eliminate domain-specific stopwords from the candidate keywords, and (7) compute the self and mutual information values which are information theoretic notion, and select keywords based on these values. Our system was implemented on a SUN UNIX workstation as well as on PC DOS using the C language. Morphological analysis divides words into the morpheme and function word. This is based on Kang and Kim[4].

1. Extracting single words

Results after analyzing each words consists of morpheme, a part of speech such as noun, verb, etc., and postpositions which functions as an auxiliary to the end of the substantives or declinable word-root. Morpheme selected as a single word is : (1) that a part of speech is a noun, (2) that a part of speech is a verb and postposition attached to it is "*hata*" or "*toyta*", or (3) that is a indeclinable adjectives.

2. Stopword elimination

For the English IR systems, standard stopword lists have been presented in Fox[7]. A stopword list contains very frequently (mostly functional) words such as articles, pronouns, etc. On the contrary, in Korea, there is no standard stopword list yet. So we have constructed Korean stopword list from document collections. In our system, there are two stopword lists, general stopword list and domain-specific stopword list[6]. General stopword list contains nouns which are useless or worthless as a keyword. It mainly consists of some less discriminating one-syllable nouns, numeral

Table 1. Conditions for constructing compound words

Post-position	Meaning	Construction
ui	"of"	n_1-*ui* n_2 => $n_1 n_2$
eyui	"of" for direction	n_1-*eyui* n_2 => $n_1 n_2$
eyseui	"of" for location	n_1-*eyseui* n_2 => $n_1 n_2$
uloui	"of" for direction	n_1-*uloui* n_2 => $n_1 n_2$
uloseui	"of" for qualification	n_1-*uloseui* n_2 => $n_1 n_2$

(1) When there exist postpositions representing the genitive case

Case	Copula ending	Meaning	Construction
accusative *"ul[lul]"*	*"hata"*	"do"	n_1-*ul[lul]* n_2-*hata* => $n_1 n_2$
nominative *"ka"*	*"toyta"*	"become"	n_1-*ka* n_2-*toyta* => $n_1 n_2$

(2) When there exist omissible postpositions

Post-position	Indeclinable adjective	Meaning	Construction
ey	"koanhan"	"be related to"	n_1-*ey koanhan* n_2 => $n_1 n_2$
	"tayhan"	"on, about"	n_1-*ey tayhan* n_2 => $n_1 n_2$
	"uihan"	"by"	n_1-*ey uihan* n_2 => $n_1 n_2$
	"sokhan"	"belong to"	n_1-*ey sokhan* n_2 => $n_1 n_2$
	"innun"	"existing in"	n_1-*ey innun* n_2 => $n_1 n_2$
lo	"toyn"	"be made of"	n_1-*lo toyn* n_2 => $n_1 n_2$
	"mantun"	"be made of"	n_1-*lo mantun* n_2 => $n_1 n_2$
	"inhan"	"be caused by"	n_1-*lo inhan* n_2 => $n_1 n_2$
ul[lul]	"uehan"	"for"	n_1-*ul uehan* n_2 => $n_1 n_2$
	"han"	"done"	n_1-*han* n_2 => $n_2 n_1$
	"hal"	"will do"	n_1-*hal* n_2 => $n_1 n_2$

(3) In the case of omissible indeclinable adjectives

nouns, incomplete nouns, highly frequently nouns and very general or less discriminating nouns. Domain-specific stopword list contains nouns which are unselectable in specific domain and can be used in composing compound words. We have built one domain-specific stopwords for economy and business in our system.

3. Extracting compound words

In Korean, postpositions tend to clarify the case of their prepositional word as mentioned in section I. In our system, the construction conditions for compound words make use of this characteristics and try to find the compound words in a sentence quickly. Construction conditions are mentioned in III.

III. Normalizing compound words

When we identify useful compound words in a document, it is desirable to recognize groups of phrases that differ in form but similar enough semantically and represent them by a single compound word. In other words, if phrases which have the same meaning are presented syntactically different, they can be indexed by one compound word. For example, "information retrieval" can be represented by different syntactic structure, such as "retrieval of information" and "retrieving information" in English. These can be represented by the same compound word, "information retrieval". In Korean, by considering the characteristics of agglutinative language, we have derived several conditions to build the compound word index as Table 1. In Table 1, n_i is i'th noun.

IV. Keyword selection method

In information theory, the predictable terms in a text —those exhibiting the least occurrence probabilities— carry the greatest information value. In particular, the information value of a text word x with occurrence probability p called SI (Self Information) is given as follows[11],

$$SI(x) = -Log_2\, p(x) \qquad (1)$$

The information value of equation (1) has been

used to estimate the term usefulness for indexing purposes[13]. When $p(x)$ are measured by counting the number of occurrences of x, there can be problem in applying $SI(x)$ to measure the information value of compound words because they have relatively the less occurrences than single words do. Therefore we need other methods to measure the information value for compound words.

We newly propose an alternative measure, mutual information, for estimating the degree of word association of compound words, based on the information theoretic notion. If two words, x and y, have probabilities $p(x)$ and $p(y)$, then their mutual information, $MI(x\,;y)$, is defined to be[10), 11)]

$$MI(x\,;y)=Log_2\frac{p(x,y)}{p(x)*p(y)} \qquad (2)$$

Word probabilities $p(x)$ and $p(y)$ are estimated by counting the number of occurrences of x and y in document collections. And probability $p(x,y)$ is estimated by counting the number of occurrences of compound word xy. Informally, mutual information compares the probability of observing compound word xy together with the probabilities of observing x and y independently. If there is a genuine association between x and y, then the probability $p(x,y)$ will be much larger than $p(x)*p(y)$, and consequently $MI(x\,;y)\gg 0$. If there is no interesting relationship between x and y, then $p(x,y)$ is closer to $p(x)*p(y)$, and thus, $MI(x\,;y)\approx 0$. If x and y are in complementary distribution, then $p(x,y)$ will be much less than $p(x)*p(y)$, forcing $MI(x\,;y)\ll 0$. Mutual information is symmetric in the sense that $MI(x\,;y)=MI(y\,;x)$. However, we do not allow this symmetry since $p(x,y)$ denotes the number of times that word x appears before y in the compound word, not the number of times the two words appear in either order.

Now if we are given three words x, y, z, we also propose the mutual information $MI(x,y\,;z)$ like follows[11)],

$$MI(x,y\,;z)=Log_2\frac{p(x,y,z)}{p(x,y)*p(z)} \qquad (3)$$

$$MI(x,z\,;y)=Log_2\frac{p(x,y,z)}{p(x,z)*p(y)} \qquad (4)$$

$$MI(y,z\,;x)=Log_2\frac{p(x,y,z)}{p(y,z)*p(x)} \qquad (5)$$

Probability $p(x,y,z)$ is estimated by counting the number of occurrences of compound word xyz. Probabilities $p(x,y)$ and $p(z)$ are estimated by counting the number of occurrences of compound word xy and a word z in document collections respectively. Because the equation (2), (3), (4), and (5) will measure the degree of word association between words, the information value of these equations can be used to derive a measure of term usefulness of compound words for indexing purposes. We have used equation (1) to select single words and equations (2), (3), (4), and (5) for compound words.

V. Decomposing compound words

Compound word consists of several consecutive nouns. We may have to decompose the compound word consists of three or more words to satisfy the exhaustivity of indexing as mentioned in[12)]. In English, the phrases "automatic text analysis" is easily recognized by syntactic analysis, that "analysis" is head word and "automatic" and "text" are modifier. Therefore "automatic analysis" and "text analysis" can be generated as keywords. In Korean, it is difficult to analyze compound word that consists of several nouns as $n_1\ n_2\ n_3 \ldots\ n_n$ where ni is a noun because it is not easy to discern if the word is a head word or modifier morphologically in agglutinative language. For example, "chutayk / kunsul / keyhoyk" 〈housing construction plan〉 is one of such compound words. Considering this compound word, it is obvious that "housing" is adjective and modifier. On the contrary, in Korean, we cannot recognize whether the word "chutayk" is a modifier or not. We can just understand that the part of speech of it is a noun. Therefore, we cannot know how this word should be decomposed.

If we are given a compound word which has component words x, y, and z, we can decompose it as xy, xz, and yz. But some of these could be very implausible. We introduce mutual information mentioned in section IV to guide the decomposition of compound words.

Table 2. Summary of Korean test collections

Test collections	378 documents
Collection size	402 K byte
The number of words	48,891 words
The number of compound words	1,653 words

In above case, information value of each case can be computed by mutual information equations as followings. Then, we can combine each words according to this value. The combinations which have higher value than threshold value will be selected as compound index words.

(1) When the number of component words is 3 (this is, *x*, *y*, and *z*), there are 3 cases to evaluate as follows,
MI(*x* ; *y*), *MI*(*x* ; *z*), *MI*(*y* ; *z*)

(2) When the number of component words is 4 (this is, *w*, *x*, *y*, and *z*), there are 10 cases to evaluate as follows,
MI(*w* ; *x*), *MI*(*w* ; *y*)*MI*(*w* ; *z*),
MI(*x* ; *y*), *MI*(*x* ; *z*), *MI*(*y* ; *z*).
MI(*w*, *x* ; *y*), *MI*(*w*, *x* ; *z*),
MI(*w*, *y* ; *z*), *MI*(*x*, *y* ; *z*)

(3) When the number of component words is 5 (this is, *v*, *w*, *x*, *y*, and *z*), there are many cases to evaluate as follows,
MI(*v* ; *w*), *MI*(*v* ; *x*), *MI*(*v* ; *y*),
MI(*w*, *x*), *MI*(*w*, *y*), *MI*(*w*, *z*),...
MI(*v*, *w* ; *x*), *MI*(*v*, *w* ; *y*), *MI*(*v*, *w* ; *z*),
MI(*v*, *x* ; *y*), *MI*(*v*, *x* ; *z*), *MI*(*v*, *y* ; *z*)

As you can see, there are many combinations for each compound word. This enforces us to estimate information value for all cases although most of all belonging to implausible case practically. Therefore, we try to deduce several general rules that can be applied when we decompose the compound words. This rule for deduction is based on the statistical analysis of our test documents. We extract the compound words from the documents first and decompose each compound word manually. To do this, we extracted the compound words from the 378 articles of Korean newspaper. Each compound word is manually decomposed into some words in order to create new related terms. As a result of above procedure, we could have the statistics as shown in Table 2 and Table 3. In Table 3, n_i is i'th noun and *j* is the position of omitted postposition. The figures on occurrence are the number of relevant and related words decomposed manually. For example, the word $n_1n_2n_3$ in type 1 can be decomposed into three cases n_1n_2, n_2n_3, n_1n_3. Among the 956 n_1n_2 words, 850 words are relevant, and 920 and 482 words are relevant for n_2n_3 and n_1n_3 respectively. These decomposed words have the relevance rates about 89%, 96% and 50% respectively. We can see that most of them are relevant when $n_1n_2n_3$ is only decomposed into n_1n_2 and n_2n_3. In the case of type 2, most of the decomposed words are relevant when $n_1jn_2n_3$ is decomposed into two words, n_1n_2 and n_1n_3.

To generate many relevant words, it is necessary for $n_1n_2n_3$ to be decomposed into n_1n_2 and n_2n_3, excluding n_1n_3 in the case of type 1. In this case, we can have about 93% relevant words and in type 2, about 81% relevant words. Based on these statistics, we have made the decomposition rules for all types as follows. That is, we have selected the decomposible forms have higher value about 70% relevance rate. In each rule, the figure in the parentheses means the average of relevance rate that is the ratio of the number of compound words decomposed relevantly to the number of compound words decomposed by each rules.

□Rule 1
Type 1 is decomposed into n_1n_2 and n_2n_3. (93%)

□Rule 2
Type 4 is decomposed into n_1n_2, $n_1n_2n_4$, and n_3n_4. (81%)

□Rule 3
Type 8 is decomposed into n_2n_3, $n_1n_2n_3$, and n_4n_5. (79%)

□Rule 4
i) Type 3 => n_1n_2, n_2n_3. (98%)
ii) Type 6 => n_1n_2, $n_1n_2n_4$, n_2n_4, n_3n_4. (92%)
iii) Type 7 => n_1n_2, n_2n_3, $n_1n_2n_3$, $n_2n_3n_4$, n_3n_4. (97%)
iv) Type 10 => n_1n_2, n_2n_3, $n_1n_2n_3$, n_4n_5. (93%)
v) Type 11 => n_1n_2, $n_1n_2n_3$, $n_1n_2n_4$,

Table 3. Statistics about modifying patterns

Type number	Type /occurrences	The decomposible forms / occurrences(relevance rate:%)		
		n_1n_2	n_2n_3	n_1n_3
1	$n_1n_2n_3$ / 956	850(89)	920(96)	482(50)
2	$n_1jn_2n_3$ / 14	5(36)	14(100)	14(100)
3	$n_1n_2jn_3$ / 131	131(100)	125(95)	52(40)

(1) When the number of component words is 3

Type #	Types / occurrences	The decomposible forms / occurrences(relevance rate:%)								
		n_1n_2	n_1n_3	n_1n_4	$n_1n_2n_3$	n_2n_3	$n_1n_2n_4$	n_2n_4	n_3n_4	$n_2n_3n_4$
4	$n_1n_2n_3n_4$ / 301	219 (72)	80 (27)	73 (24)	210 (69)	181 (60)	223 (74)	93 (31)	286 (95)	130 (43)
5	$n_1jn_2n_3n_4$ / 9	0 (0)	4 (44)	9 (100)	7 (78)	8 (89)	1 (11)	3 (33)	9 (100)	9 (100)
6	$n_1n_2jn_3n_4$ / 38	36 (95)	0 (0)	4 (11)	4 (11)	6 (16)	35 (92)	31 (82)	38 (100)	15 (40)
7	$n_1n_2n_3jn_4$ /108	92 (85)	35 (32)	2 (2)	108 (100)	106 (98)	2 (2)	1 (1)	108 (100)	108 (100)

(2) When the number of component words is 4

Type #	Type / Occurrences	The decomposible forms / occurrences(relevance rate:%)					
8	$n_1n_2n_3n_4n_5$ / 47	n_1n_2	n_2n_3	$n_1n_2n_3$	n_4n_5	$n_2n_3n_4$	$n_2n_3n_5$
		26(55)	37(79)	35(75)	39(83)	29(61)	26(55)
9	$n_1n_2jn_3n_4n_5$ / 9	n_1n_2	n_3n_4	n_4n_5	$n_3n_4n_5$	$n_1n_2n_4$	$n_1n_2n_5$
		9(100)	6(67)	8(89)	9(100)	4(44)	5(56)
10	$n_1n_2n_3jn_4n_5$ / 18	n_1n_2	n_2n_3	$n_1n_2n_3$	n_4n_5	$n_2n_3n_4$	$n_2n_3n_5$
		16(89)	15(83)	18(100)	18(100)	7(39)	7(39)
11	$n_1n_2n_3n_4jn_5$ / 22	n_1n_2	$n_1n_2n_3$	$n_1n_2n_4$	$n_2n_3n_4$	n_3n_4	n_4n_5
		19(86)	18(82)	13(59)	13(59)	16(72)	15(68)

(3) When the number of component words is 5

$n_2n_3n_4$, n_3n_4, n_4n_5. (70%)

□Rule 5

i) Type 2 => n_2n_3, n_1n_3. (100%)

ii) Type 5 => n_1n_4, $n_1n_2n_3$, $n_2n_3n_4$, n_2n_3, n_3n_4. (93%)

iii) Type 9 => n_1n_2, $n_3n_4n_5$, n_3n_4n n_4n_5. (89%)

Rule 1, 2, and 3 are the ones that decompose the compound words generated without omitting postpositions, and rules 4 and 5 are the ones decomposing compound words generated by omitting postpositions and equivalent postpositions. This rule has good relevance rate. Average rate is about 90%. Table 4 shows the rate of relevance for the decomposition rules applied in our test documents. These rules decompose about 89% of all decomposable compound words. Using this rules, we can generate the plausible words easily in relative. Among these generated words, we select these have higher value as keywords, by estimating *MI*(*x* ; *y*), *MI*(*x*, *y*; *z*), *MI*(*x*, *z* ; *y*), or *MI*(*y*, *z* ; *x*).

VI. Evaluation

A. Collections

To estimate the indexing performance, it is necessary to use a standard test collection for the experiment. However, unfortunately, a standard test collection in Korean, such as CACM collection in English, is not established yet. Therefore, we used a test collection of about 25 articles, which are selected from the popular daily newspapers of Korea.

Table 4. The proportion of decomposition of compound words

	EN	PN	TD	FD	TD/PN(%)
Rule 1	274	596	510	38	90%
Rule 2	41	141	117	6	83%
Rule 3	3	18	15	3	83%
Rule 4	146	498	445	35	89%
Rule 5	31	107	94	8	89%
Total	458	1333	1181	90	89%

EN : *the number of compound words extracted from documents*
PN : *the number of compound words generated by manual decomposition*
TD : *the number of relevant compound words generated by our rules*
FD : *the number of irrelevant compound words generated by our rules*

B. Some experiments

Usually the effectiveness of automatic indexing or information retrieval is measured by the recall and precision rates[13), 16)]. For evaluating automatic indexing results in our system, they are given by Figure 1. We have only evaluated the precision factor of our system because the term-recall can be influenced by the manual indexers and we have not standard test collection for experiment. To do this, we first extract terms from each documents as candidate keywords automatically. In selecting keywords, we calculate the $SI(x)$ for single words and the $MI(x\,;y)$ for compound words. For computing $MI(x\,;y)$, probability $p(x)$ is firstly estimated by counting the number of occurrences of term x in a document. Probability $p(x, y)$ is estimated by counting the number of occurrences of compound word xy in a document. For example, if the total number of extracted terms is 100, 20 and 40 for term x and y respectively, and 15 for the compound word xy, $MI(x\,;y)$ is computed like this;

$$Term-recall=\frac{The\ number\ of\ extracted\ words\ relevant}{The\ total\ number\ of\ relevant\ words}$$

$$Term-precision=\frac{The\ number\ of\ extracted\ words\ relevant}{The\ total\ number\ of\ extracted\ words}$$

Figure 1. The rates of the term-recall and term precision

$$MI(x\,;y)=Log_2\frac{15/100}{20/100*40/100}$$

Therefore, for each document, all words have their weighting values to be selected as keywords.

Mutual information $MI(x\,;y)$ is satisfied as like;

$0\leq MI(x\,;y)\leq Log(N)$,

$Log(N)$ is the greatest mutual information in one document, and N is the total number of terms extracted. In our system, we take the threshold to select keywords as following;

if $(Log(N)-2)\leq MI(x;\,y)\leq Log(N)$, *select them if not, not select*

To evaluate the performance of indexing in our system, three intelligent persons choose terms from the set of terms as keywords, which have only properness and relatedness for that document in meaning when selecting single and compound words. And then we intersect the relevant terms selected by three persons and finally take the set of terms as keywords for one document.

We have performed the three types of experiments as followings

Experiment 1: no normalization, no decomposition, and applying the equation (1).

Experiment 2: normalization, decomposition, and applying the equation (1).

Table 5. The rate of term-precision of indexing terms

Experiment / Articles	Experiment 1		Experiment 2		Experiment 3	
	A	B	A	B	A	B
Article(5)	83.5	75.2	80.0	66.8	90.0	85.9
Article(5)	81.8	75.7	80.5	78.4	89.6	88.1
Article(5)	79.8	74.6	79.0	72.3	89.5	93.8
Article(5)	82.1	63.4	80.4	67.3	89.2	80.9
Article(5)	84.6	70.5	82.7	66.9	91.6	85.6
Average	82.7	71.9	80.5	69.6	88.7	86.8

A : *term-precision rate for single and compound words*
B : *term-precision rate for compound words only*

Experiment 3: normalization, decomposition, and applying the equations (2), (3), (4), and (5).

C. Evaluation

The average precision figures for the documents are shown in Table 5. Comparing with Experiment 1 and 2, we can observe that, although we have used construction conditions and decomposition rules, our system has good precision rate. Also, when we compare Experiment 2 with 3, we have very interesting results. We can see that mutual information value is very useful in estimating the weight of compound words. Our system has raised the term-precision level from 72% to 87% using the evaluation method of mutual information value. Our system also enhances the exhaustivity of indexing and the specificity of terms by applying construction conditions and decomposition rules.

VII. Conclusion

This paper presents an automatic indexing technique for compound words suitable to an agglutinative language, specifically Korean. We have presented some of the construction conditions for compound words. Also we have presented the decomposing rules for compound words to enhance the exhaustivity of indexing. We have shown that, using construction conditions and decomposition rules, our system enhanced the exhaustivity of indexing and the specificity of terms. And by using a mutual information to select relevant compound words, we have raised the term-precision level from 72% to 87%. The construction conditions and decomposition rules presented in this paper may be used in multilingual information retrieval systems to transform the index terms of the specific-language into those of needed language.

References

1) Dillon, Martin; Gray, Ann S. "FASIT: A Fully Automatic Syntactically Based Indexing System". Journal of the American society for Information Science. Vol. 34, p. 99–108 (1983)
2) Fagan, Joel L. Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-Syntactic Methods. Ph.D. Thesis, Technical Report 87–868, Cornell University, Computer Science Department, 1987.
3) Salton, Gerard; Buckley, Chris. A Comparison between Statistically and Syntactically Generated Term Phrases. Technical Report 89–1027, Cornell University, Computer Science Department, 1989.
4) Kang, S. S.; Kim, Y. T. "A Statistical Approach to Syllable-based Morphological Analysis". Proceedings of the international Conference on Computer Processing of Chinese and Oriental Languages, 1992.
5) Kim, P. K.; Cho, Y. K. "A Design of the Korean Information Retrieval System". Proceedings of the 20 th KISS Spring Conference. in Korean. Vol. 17, No. 2, p. 791–794 (1990)
6) Kim, P. K.; Cho, Y. K. "Construction and Application of Stopword List for Korean Information Retrieval". Proceedings of the 20 th KISS Spring Conference. in Korean. Vol. 20, No. 1, 1993.
7) Fox, C. "A Stop List for General Text". SIGIR FORUM. Vol. 24, No. 1–2, p. 19–35 (1990)
8) Park, H. R.; Chung, K.; Choi, G. S. "Syntactic Information Based Automatic Indexing for Korean Texts". Natural Language Processing Pacific Rim Symposium. 1991.
9) Kim, M. J.; Kwon, H. C. "An Automatic Indexing Method using the Characteristics of Korean". Proceedings of the 19th KISS Spring Conference. in Korean. Vol. 19, No. 2 (1992)
10) Church, K. W. "Word Association Norms, Mutual Information, and Lexicography". Computational Linguistics. Vol.16, No. 1, p. 22–29 (1990)
11) McEliece, R. J. The Theory of Information and Coding. Addison Wesley, 1977.
12) ISO 5963, Documentation-Methods for examining documents, determining their subjects, and selecting indexing terms. 1985.
13) Salton, G. Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison Wesley, 1989.
14) Ogawa Y.; Bessho A.; Hirose M. "Simple Word Strings as Compound Keywords: An Indexing and Ranking Method for Japanese Texts". Proceedings of the 16th ACM SIGIR Conference on R & D in IR. 1993.
15) Fujii H.; Croft, W. B. "A Comparison of Index-

ing Techniques for Japanese Text Retrieval". Proceedings of the 16th ACM SIGIR Conference on R & D in IR. 1993.

16) Frakes, W. B. Information Retrieval-Data Structures and Algorithms. Prentice-Hall, NJ, 1992.

17) Croft, W. B.; Turtle, H. R. "The Use of Phrases and Structured Queries in Information Retrieval," ACM SIGIR Conference on R & D in IR. p. 32–45 (1991)

18) Kim, P. K. "Automatic Indexing of Compound Words Based on Mutual Information for the Korean Information Retrieval". in Korean. Ph. D. Thesis, Seoul National University, Department of Computer Engineering, 1994.