A Deterministic Method for Structural Analysis of Compound Words in Japanese

Dongli Han, Takeshi Ito and Teiji Furugori
Department of Computer Science, The University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan
{han, ito, furugori@phaeton.cs.uec.ac.jp}

Abstract

Structural analysis of compound words is necessary and an important process in natural language processing. Proposed here is a corpus- and statistics- based method for the structural analysis of compound words in Japanese. We determine the structure of a compound word by using Internet corpus and calculating the strength of word association among its constituent words. Experiments with 5, 6, 7, and 8 kanji compound words show that our method works well and its performance is better than those of other comparable studies.

1 Introduction

A sequence of words may assume a single syntactic function in English and the same is true in Japanese. *Machine translation system,* for instance, consists of three words *machine*, *translation* and *system*. Its equivalent in Japanese, 機械翻訳システム, is a compound word with three constituent words, 機械 (machine), 翻訳 (translation), and システム (system). It is a single concept and functions as a noun.

Both English and Japanese face problems in processing such a sequence of words. A problem involved in English is in identifying the sequence of words as a syntactic as well as semantic unit. It is easy to find a compound word in Japanese but we have difficulties in segmenting it into constituent words as well as determining its syntactic or semantic structure among the constituents.

Finding its constituents and determining the structure of a compound word is an important process in constructing practical natural language systems for machine translation, information retrieval, text summarization, etc. For the compound word 元日銀総裁 (the former president of Bank of Japan), for instance, there are six lexically possible segmentations: 元/日/銀/総/裁, 元/日/銀/総裁, 元/日銀/総/裁, 元/日銀/総裁, 元日/銀/総/裁 and 元日/銀/総裁, the correct segmentation here being 元(former)/日銀(Bank of Japan)/総裁(president).

Standing alone, the word segmentation is useful in many areas of natural language processing. However, we should know how the constituents in a compound word are combined to make the meaning, say, for its translation into foreign languages. 元/日銀/総裁 has two probable dependency structures: ((元)((日銀)(総裁))) and (((元)(日銀))(総裁)). The former leads to the meaning of *the former president of Bank of Japan* and the latter *the president of former Bank of Japan.* The problem is how can we arrive at the structure, ((元)((日銀)(総裁))).

In this paper we offer a method for analyzing the structure of compound words in Japanese. In Section 2 describes some previous work done for a compound word on the identification of constituents and determination of its structure. We propose a method and subsequently an algorithm in

Section 3 for analyzing the structure of compound words using the strength of association measures among the constituent words. In Section 4 conducts experiments for our method and others, and compares the results. Finally, we conclude the paper with some remarks in Section 5.

2 Some Work on the Analysis of Compound Words

We have two tasks in the analysis of a compound word: identification of its constituents and then the determination of its structure. Our present interest lies in the latter task.

There are numerous studies on the identification for the constituents of a compound word. To mention a few, Miyazaki et al. (1984) used a handcrafted, rule-based method and a technique of ambiguity resolution for segmenting a compound word. Takeda and Fujisaki (1987) segmented kanji compound words using Markov Model. Utiyama and Itahashi (1992) divided a compound noun expressed in hiragana characters into component nouns based on word co-occurrence relation. In recent ones, Shimohata et al. (1997) and Fujii et al. (1999) segmented a compound word into its constituents employing simple heuristics based mainly on the features of Japanese character types. Han et al. (2001a) devised a segmentation strategy using contextual information from a corpus.

Relatively few studies are available on the determination of dependency structure of a compound word. Nishino and Fujisaki (1988) attempted to analyze the structure of a kanji compound word using a probabilistic CFG. Miyazaki et al. (1993) used a large amount of handcrafted rules and resolved the ambiguities on semantic structure in compound words.

These studies are all rule-based and have obvious deficiencies. Obtaining rules is time-consuming process to begin with. Then rules are limited, fixed and not easily extendable.

Recently the researchers have turned to corpus and statistical means to analyzing the structure of compound words. Along this line, Kobayasi et al. (1994) built dependency trees for a compound word, calculated the likelihood of each tree structure using collocational information on its constituents extracted from a corpus, and selected the one with the highest likelihood value as the correct structure of the compound word. In this process, they used a thesaurus and got the collocational data through its classes to which each constituent of a compound word belongs.

Using a similar method, Hisamitsu and Nitta (1996), in an analysis of compound words extracted from news articles, tried to overcome a deficiency in the study of Kobayasi et al. that it could not deal with abbreviated words or, for that matter, unknown words in the corpus used. However, the method he adapted seems restricted only to the analysis of newspaper articles.

Our method of analyzing the structure of a compound word is corpus- and statistics- based, too. But it is different from others in a few points. Methodologically, it uses a deterministic process. Computationally, we calculate the strength of association, for the constituent words in a compound word using mutual information-like measures and determine its correct structure by discarding less probable ones along the way. Data-wise, we use the Internet corpus to coping with data sparseness problem and getting reliable statistical information.

3 A Structural Analysis of Compound Word

Our idea for analyzing the structure of compound words is similar to those used in various disambiguation tasks (e.g., Alves, 1996; Wu & Furugori, 1996; Karov et al., 1998). We all use a kind of

corpora and statistical means and then eliminate the more unlikely and select the more likely to get the desired result. However, the difference is that we use a deterministic process and keep discarding less likely candidates along the way we go, rather than collecting all the candidates first and then selecting the most probable one from them.

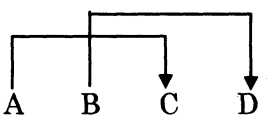
3.1 Analytical Processes

A compound word consists of a sequence of words. Each word in the compound word, except the rightmost one or headword, relates to, or depends semantically on, one of other words in its right more strongly than to any other words. For example, 関西 (Kansai) relates more strongly to 空港 (airport) than to 国際(international) in 関西/国際/空港 (Kansai International Airport).

We know for a Japanese compound word that the semantic dependency relations among its constituents has the following characteristics:

- The dependency relation that holds between two constituents is unique, i.e., no constituent relates to more than one constituent.
- An element, except the headword, depends always on a constituent to its right.
- No dependency relations cross each other.

The last characteristic is to mean that we never have dependency relations something like:



We may be able to find the structure of a compound word with its constituents, $w_1 w_2 \ldots\ldots w_i \ldots\ldots w_n$, using the dependency relations among its constituent words. An algorithm for this is:

Step1: For any word $w_i\,(1 \le i \le n-1)$ in the constituent sequence, S, of n words, $w_1 w_2 \ldots\ldots w_n$, find the word w_j to which w_i has the dependency relation $w_i \rightarrow w_j$ $(i < j)$. Call the set of relations the rules, R.

Step2: Repeat the following

2.1 Find the handle, h, or the leftmost dependency relation in S with the minimum inter-constituent distance (=1); Remove the handle from R.

2.2 If h takes a form of $w_i \rightarrow w_j$, amalgamate w_i and w_j in S with $C_{i,j}$ where C is an amalgamated constituent.

2.3 If h takes a form of $w_i \rightarrow C_{i+1,j}$ or $C_{i,j-1} \rightarrow w_j$, amalgamate w_i and $C_{i+1,j}$ or $C_{i,j-1}$ and w_j in S with $C_{i,j}$.

2.4 If h takes a form of $C_{i,i+x} \rightarrow C_{i+x+1,j}$ $(x>0)$, amalgamate $C_{i,i+x}$ and $C_{i+x+1,j}$ in S with $C_{i,j}$.

2.5 Replace the expression in R that is the same as the right side of expression in h with $C_{i,j}$.

until no amalgamation becomes possible.

Let us see how the algorithm works. Suppose the compound word to be analyzed has five constituents, $w_1 w_2 w_3 w_4 w_5$. Suppose also that we have found the following rules in Step1.

$$w_1 \rightarrow w_3,$$
$$w_2 \rightarrow w_3,$$

$$w_3 \rightarrow w_5,$$
$$w_4 \rightarrow w_5$$

We then find h to be $w_2 \rightarrow w_3$ in Step 2.1. So, we amalgamate w_2 and w_3 with $C_{2,3}$ in Step 2.2 and get new S of $w_1\, C_{2,3}\, w_4\, w_5$. In Step 2.5, R is changed to:

$$w_1 \rightarrow C_{2,3},$$
$$C_{2,3} \rightarrow w_5,$$
$$w_4 \rightarrow w_5$$

With the new S and R, we go back to Step 2.1 and repeat the processes in Step 2. This time, we find h to be $w_1 \rightarrow C_{2,3}$ in Step 2.1, amalgamate w_1 and $C_{2,3}$ with $C_{1,3}$ in Step 2.3 and get $C_{1,3}\, w_4\, w_5$, and change R in Step 2.5 to:

$$C_{1,3} \rightarrow w_5,$$
$$w_4 \rightarrow w_5$$

In the next time around, h is $w_4 \rightarrow w_5$ and applying the Steps 2.2 and 2.5, we get new S and R of $C_{1,3}\, C_{4,5}$ and $C_{1,3} \rightarrow C_{4,5}$. Repeating Step 2 again, we get h of $C_{1,3} \rightarrow C_{4,5}$, amalgamate $C_{1,3}$ and $C_{4,5}$ with $C_{1,5}$ in Step 2.4, and finally stop processing as no further amalgamation becomes possible.

Table 1 shows the above processes step by step and Figure 1 the structure of the compound word analyzed.

Table 1: Processes of Structural Analysis

Steps	Constituents	Rules	Handle and Amalgamation
2.1, 2.2	$w_1 C_{2,3} w_4 w_5$		$w_2 \rightarrow w_3 \Rightarrow C_{2,3}$
2.5		$w_1 \rightarrow C_{2,3}$ $C_{2,3} \rightarrow w_5$ $w_4 \rightarrow w_5$	
2.1, 2.3	$C_{1,3} w_4 w_5$		$w_1 \rightarrow C_{2,3} \Rightarrow C_{1,3}$
2.5		$C_{1,3} \rightarrow w_5,$ $w_4 \rightarrow w_5$	
2.1, 2.2	$C_{1,3} C_{4,5}$		$w_4 \rightarrow w_5 \Rightarrow C_{4,5}$
2.5		$C_{1,3} \rightarrow C_{4,5}$	
2.1, 2.4	$C_{1,5}$		$C_{1,3} \rightarrow C_{4,5} \Rightarrow C_{1,5}$
2.5		(no rules left)	

We left Step1 unexplained, but this is the most crucial step in the algorithm. We base our analysis in this paper on a search in which we try to find each relation of $x \rightarrow y$ in a compound word always in relation to the rightmost constituent or the headword. We describe this process in detail in the next sub-section.

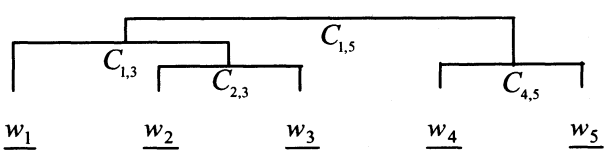


Figure 1: Structure of Compound Word Analyzed

3.2 Measurement of the strength of association

We try to determine the dependency relation, $x \rightarrow y$, using some mutual information-like metrics. As is well-known, mutual information (MI) is a standard way of estimating the strength of lexical association between any two words (Church, 1990). It is defined as:

$$I(w_1, w_2) = \log\left(\frac{N \times f(w_1, w_2)}{f(w_1) \times f(w_2)}\right) \quad (1)$$

where N is the size of the corpus used in the estimation, $f(w_1, w_2)$ is the frequency of co-occurrences of w_1 and w_2, and $f(w_1)$ and $f(w_2)$ is the frequency of each word.

Naturally the reliability of MI depends greatly on the statistical data obtained from the corpus to be used. This leads us for our analysis to use Internet corpus rather than other corpora widely available.

Owing to its largeness in size, the Internet corpus is expected to make the data sparseness problem less problematic than that in the use of other corpora.[1] In fact, it seems that we are able to access at least 42 million pages of Japanese texts using Goo[2], a well-known Internet search engine that is said to be the biggest in Japan.

We modify formula (1) and define two types of Internet-based Mutual Information (IMI) as in formulas (2) and (3) and then use them to measure the strength of association between two words w_1 and w_2.

$$IMI_1(w_1, w_2) = \log\left(\frac{N \times hit(w_1 \ \ AND \ \ w_2)}{hit(w_1) \times hit(w_2)}\right) \quad (2)$$

$$IMI_2(w_1, w_2) = \log\left(\frac{N \times hit(w_1 w_2)}{hit(w_1) \times hit(w_2)}\right) \quad (3)$$

In the formulas, N denotes the total number of the Japanese URLs registered in a search engine. It is 42 million in our case. $hit(x)$ is the hit number we get when searching the word x in a search engine; $< w_1 \ \ AND \ \ w_2 >$ is a query formula gotten by applying the logical operator AND to w_1 and w_2; $hit(w_1 w_2)$ is the hit number when searching a compound word of the two constituent words w_1 and w_2.

In our algorithm to finding $x \rightarrow y$ in a compound word, we first find the most unacceptable constituent pair (w_{mua} , w_n) that has the weakest strength of association in all pairs of (w_i , w_n),[3] where $1 \le i \le n-2$.[4] We then take w_{mua} as the starting point, find w_{mac} for which w_{mua} has the highest strength of association in the sequence of $w_{mua+1} \ldots w_{n-1}$, compare (w_{mua} , w_{mac}) with (w_{mua} , w_n), let $mac = n$ if (w_{mua} , w_n) is more acceptable, and choose the pair with bigger associa-

---

[1] Bergh et al. (1998) estimates that the Internet material published in English and accessed through the AltaVista search engine is about 25 times bigger than that in the Bank of English (320 million words), 80 times bigger than that in the BNC (100 million words), 160 times bigger than that in the CobuildDirect corpus (in its 50-million-word version), and 8000 times bigger than that in Brown and LOB corpora (one million words each). The usefulness of the Internet corpus has been proven in various linguistic applications (e.g., Mihalcea and Moldovan, 1999; Miyahira et al., 2000; Han et al., 2001b).

[2] http://goo.ne.jp

[3] We try to find inner relations first by doing this. Although intuitive, we believe this is a better way of finding the structure of a compound word.

[4] We do not take w_{n-1} into account as it is obvious that we have $w_{n-1} \rightarrow w_n$.

tion value as a dependency relation in the compound word. After this, we divide the sequence $w_1...w_{mua}...w_{mac}...w_n$ into two parts $w_1...w_{mua}w_{mac}...w_n$ and $w_{mua+1}...w_{mac}$, and for each one apply the above process recursively. Below is an algorithm to do the task.

Main {
  For each w_i in $w_1w_2.....w_i.....w_{n-2}$ {
    set $mark[w_i]=0$
  }
  call $R\_search(w_1w_2.....w_i.....w_n)$
}
Sub $R\_search(w_1w_2.....w_i.....w_n)$ {
  if (exist $w_i|\{mark[w_i]=0,\ i\in(1...n-2)\}$) {
    compute and find the smallest $IMI(w_i,w_n)|\{mark[w_i]=0\wedge i\in(1...n-2)\}$
      and let it be $IMI(w_{mua},w_n)$
    compute and find the largest $IMI(w_{mua},w_j)|\{j\in(mua+1...n-1)\}$
      and let it be $IMI(w_{mua},w_{mac})$
    if $IMI(w_{mua},w_{mac})$ < $IMI(w_{mua},w_n)$
      set $mac=n$
    acquire a dependency relation $w_{mua}\rightarrow w_{mac}$
    set $mark[w_{mua}]=1$
    call $R\_search(w_{mua+1}.....w_{mac})$
    call $R\_search(w_1.....w_{mua}w_{mac}.....w_n)$
  }
}

Here, *mark* is an array storing 2-value flags for $w_i(i=1...n-2)$. When the dependency relation for w_i has been found, we let $mark[w_i]$ be 1, otherwise 0; w_{mua} indicates the word for which the strength of association (w_{mua},w_n) is weaker than between any other constituent word and w_n. Similarly, w_{mac} is the word for which (w_{mua},w_{mac}) takes the strongest association; R_search is a subprogram that is recursively invoked for seeking dependency relations among the constituent words in the compound word to be analyzed.

In our algorithm, we always take w_{mua} as the starting point to find its most acceptable companion. This is meaningful. Among all the dependency relations, the ones related to the headword are more significant for determining the structure. The analysis would become futile when a wrong dependency relation in which the headword takes part is obtained. We are to avoid such a disaster by putting our hand on w_{mua} that always has the weakest relation with the headword.

4 Experiments and Results

We examine how effective our method is by conducting some experiments for our method. We first describe the nature of test data and then show the experimental results.

4.1 Test Data

We use Mainichi News'1994, an annual volume of a newspaper in Japanese, as the source for the test

data. We select from it the top 100 most frequently utilized compound words for the lengths of 4, 5, 6, 7, and 8 and segment them into their constituent words using the system Han et al. have built (2001a). But we use the four sets of 100 compound words for the length of 5 to 8 in the test, since we found that many of 4 kanji-character words are divided into two constituent words and it therefore makes the structural analysis meaningless.

4.2 Preliminary Experiments

Table 2 contains the experimental results with baseline methods and ours. The result in (a) is obtained from a baseline method called leftmost derivation (Lauer, 1995) in which a word is always attached to its predecessor in a phrase. The result in (b) shows the performance of another baseline method opposite to the first one, i.e., rightmost derivation that seems to be effective in analyzing the structure of a certain type of Japanese noun phrases (Furugori & Alves, 1999).

Table 2: Experimental Results 1 (% correct)

Method \ Length	five	six	seven	eight
(a)	79	53	43	36
(b)	82	56	43	20
(c)	83	65	37	25
(d)	86	67	56	49
(e)	83	71	57	55

The results in (c), (d) and (e) show the performances from our methods with the use of formulas (1), (2), and (3), respectively. Here, we used the EDR Japanese Corpus[5] in the calculation of MI in (c).

The success rate in any method decreases as the length of kanji character sequence increases. This is natural as the candidate structures in a compound word increase with the length. In fact, 5 kanji-character sequences in our test data have 2.4 constituents on average and so for each of them, we would have 1 or 2 possible structures. The average number of constituents for 8 kanji-character sequences is about 4 with the maximum possible structures of 5.

The results in (c), (d) and (e) are better than those of baseline methods. The difference in performance becomes greater as the character length becomes bigger, except in the case of (c).

We contend that the result in (c) is something to do with the data sparseness problem that comes from the use of a conventional corpus. For this, we have found that the numbers of constituent words that do not occur or co-occur are 2, 4, 16, and 19 in the EDR corpus, respectively, for the compound words of the length 5, 6, 7, and 8.

Between the results of (d) and (e), we see the latter is better than the former, except in the case of character length being 5. This is expected. In formulas (2) and (3) used for (d) and (e), w_1 and w_2 are constituent words of a compound word, and $hit(w_1 \ AND \ w_2)$ in (2) searches web pages where w_1 and w_2 co-occur in a document, while $hit(w_1 w_2)$ in (3) searches the pages where w_1 and w_2 appear in succession of $w_1 w_2$. In other words, $hit(w_1 \ AND \ w_2)$ makes the search operation less focused or less contextually bound but produces matches more in volume, while $hit(w_1 w_2)$ makes the search operation more focused or more contextually bound but produces

[5] The EDR Japanese Corpus is provided by the Japanese Electronic Dictionary Research Institute. It contains 210,000 sentences with all the words segmented.

matches less in volume. So, when we have no data sparseness problem with the corpus we use, it is better to use $hit(w_1 w_2)$, rather than $hit(w_1 \ AND \ w_2)$, to analyze the structure of a compound word whose constituents are assumed to have a tendency to appear closely with each other.

4.3 Refined Experiment

A close examination reveals that over half of the analytical errors in (e) are correctly analyzed using the formula (2) for (d). For instance, for the compound word,

日本/相撲/協会(Wrestling Society of Japan)
(日本：Japan 相撲：wrestling 協会：society)

we get a wrong result (((日本)(相撲))(協会)) in (e): we hardly find a compound word 日本/協会 on the Internet. But the correct structure of ((日本)((相撲)(協会))) is obtainable in (d) as 日本 and 協会 co-occur with some distance in documents on the Internet.

We also find that most of the wrongly analyzed compound words in (e) begin with a constituent word, called the lead word or w_{lead} here, that represents one of the following three concepts, called TLO here:

(1) Time e.g., 前日(the day before), 戦後(postwar)
(2) Location e.g., 日本(Japan), 太平洋(Pacific Ocean), 東京(Tokyo), 関西(Kansai)
(3) Organization e.g., 国連(United Nations), 自民党(Liberal Democratic Party), 日産(Nissan), 参院(Upper House)

The lead word in general relates most strongly to the headword in a compound word. For instance, 東京 is a lead word in 東京(Tokyo)/外国(foreign)/為替(exchange)/市場(market) meaning *foreign exchange market in Tokyo* and then we know that we have 東京→市場.

With these facts in mind, we try to refine our method, devise a hybrid method combining the strengths in (d) and (e), and test it in experiment. The formula for the hybrid method is:

$$IMI_3(w_1, w_2) = \begin{cases} IMI_1(w_1, w_2) & if \ \ w_{ini} \in TLO \\ IMI_2(w_1, w_2) & otherwise \end{cases} \qquad (4)$$

The initial constituent word in a compound word denoted as w_{ini} in (4) is determined whether to be a w_{lead} using the dictionary Matsumoto et al. (2001) use for ChaSen, a morphological analyzer.

Table 3 shows the experimental results with the use of (4). Included here for the sake of comparison are the two results from dependency tree method, D-tree(1) and D-tree(2), used in Kobayasi et al. (1994) and Hisamitsu et al. (1996): D-tree(1) estimated the strength of association between words using the formula (2) and D-tree(2) the formula (3).

Table 3: Experimental Results 2 (% correct)

Method \ Length	five	six	seven	eight
Hybrid	88	74	66	64
D-tree(1)	86	68	56	48
D-tree(2)	83	71	57	55

Figure 2 exemplifies the results from the baseline method (a), D-tree(1) method, and hybrid method in graph. From these, we know that our hybrid method works well. Its performance as we can see is better than those of any other comparable methods in analyzing the structure of compound words.

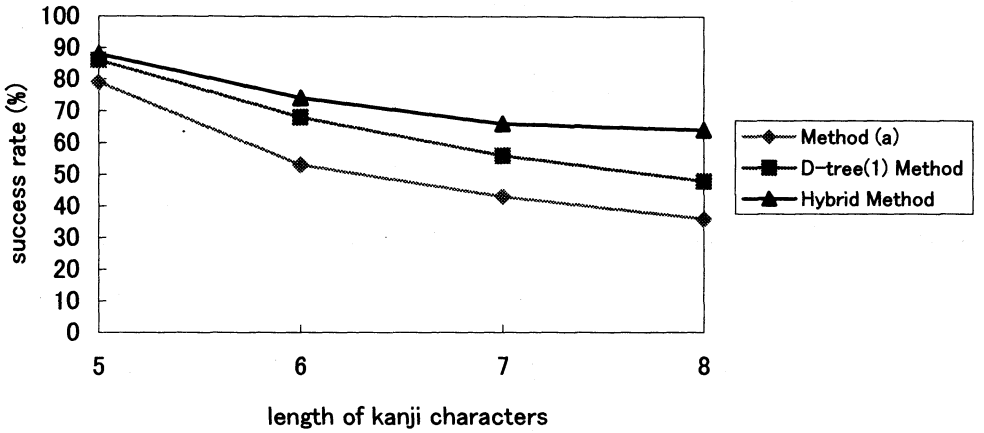


Figure 2: Performances in Various Methods

4.4 An Illustrative Example

Let us take an example to see how the analysis is done by the hybrid method. Consider this compound word:

政治/改革/関連/法案 (bills on political reform)
(政治：politics 改革：reform 関連：relation 法案：bill)

We do not find a lead word here. So, we use formula (3) to calculate the strength of association between the headword and other constituent words as is in the top of Figure 3.

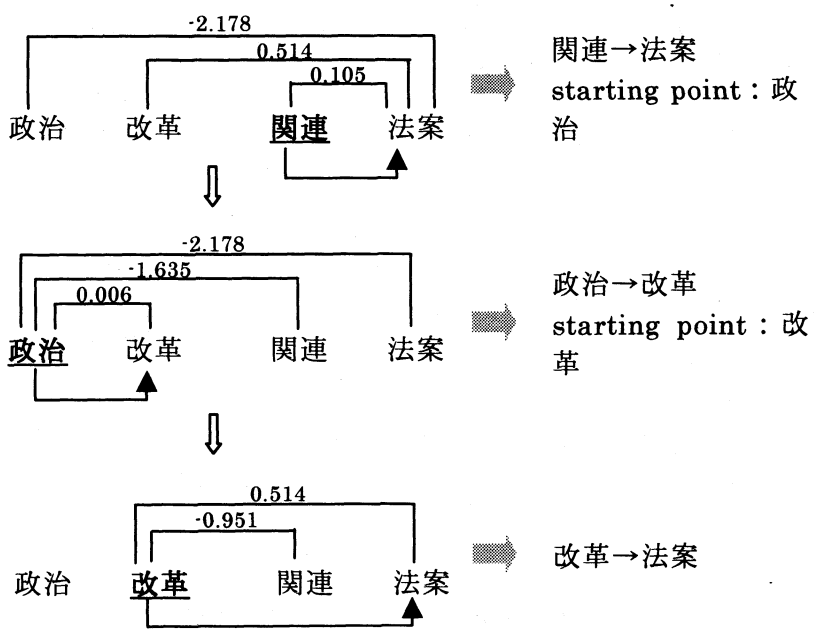


Figure 3: Processes of Finding Dependency Relations

Now we locate the starting point of the analysis at 政治, and get the first dependency relation 関連→法案 as 法案 is the only word following 関連. Then we get *IMI* (政治,改革) as the strongest one among *IMI* (政治,改革), *IMI* (政治,関連) and *IMI* (政治,法案): we acquire the dependency relation 政治→改革 as is in the middle of Figure 3. Finally, by comparing *IMI* (改革,関連) and *IMI* (改革,法案), we obtain the dependency relation for 改革, the last constituent word: 改革→法案 as is in the bottom of Figure 3. The dependency relations for all the constituent words are:

$$\begin{pmatrix} 政治 \rightarrow 改革 \\ 改革 \rightarrow 法案 \\ 関連 \rightarrow 法案 \end{pmatrix}$$

The analytical processes, after getting the dependency relations, are already explained in Section 3.1.

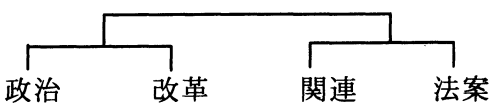


Figure 4: Resulting Structure

Figure 4 shows the resulting structure, (((政治)(改革))((関連)(法案))), and Table 4 contains some more exemplary results from the hybrid method.

Table 4: Exemplary Results[6]

Length of Compound	Segmentation	Structure
5	官房/副/長官(cabinet / vice / secretary)	((官房)((副)(長官)))
	参院/本/会議(Upper House / plenary / session)	((参院)((本)(会議)))
	中/選挙区/制(medium / constituency / system)	(((中)(選挙区))(制))
	国内/総/生産(domestic / gross / product)	* (((国内)(総))(生産))
	内外/価格/差(home and abroad / price / gaps)	* (((内外)(価格))(差))
6	関西/国際/空港(Kansai / international / airport)	((関西)((国際)(空港)))
	実験/用/原子炉(experiment / use / atomic reactor)	(((実験)(用))(原子炉))
	日本/野球/連盟(Japan / baseball / association)	((日本)((野球)(連盟)))
	希望/小売/価格(desired / retail / price)	* (((希望)(小売))(価格))
	米/通商/代表/部(USA / trade / representative / section)	* (((米)(通商))((代表)(部)))
7	政治/資金/規正/法(politics / funds / regulation / act)	(((政治)(資金))((規正)(法)))
	自民党/前/副/総裁(Liberal Democratic Party / former / vice / president)	((自民党)((前)((副)(総裁))))
	米/連邦/準備/制度(USA / federation / preparation / system)	(((米)(連邦))((準備)(制度)))
	比例/代表/並立/制(proportion / representative / parallel / system)	* (((比例)((代表)(並立)))(制))
	国連/平和/維持/軍(United Nations / peace / keep / force)	* (((国連)(平和))((維持)(軍)))
8	政治/改革/関連/法案(politics / reform / relation / bill)	(((政治)(改革))((関連)(法案)))
	総合/外交/政策/局長(general / diplomacy / policy / bureau chief)	(((総合)((外交)(政策)))(局長))
	北米/自由/貿易/協定(North America / free / trade / agreement)	((北米)(((自由)(貿易))(協定)))
	小/選挙区/比例/代表(small / constituency / proportion / representative)	* ((((小)(選挙区))(比例))(代表))
	中小/企業/金融/公庫(small and medium / enterprise / finance / corporation)	* ((((中小)(企業))(金融))(公庫))

Note: * indicates a failure, i.e., wrong structure.

[6] A compound word consists of nouns alone or noun(s) and affix(es) attached to them. In this table and other places, we tried to use English nouns when transcribing its constituents. For instance, 政治改革 (*political reform*) consists of two nouns and thus we give the transcription: 政治/改革(politics / reform).

4.5 Evaluation

The results we got are better than those of others, though a direct comparison with other studies is impossible for various reasons. In reference, Kobayasi et al. (1994) reported their results on 5 and 6 kanji compound words with the success rate of 79% and 70%. Hisamitsu and Nitta (1996) got the rates of 89% and 70% for 5 and 6, 58% and 58% for 7 and 8 kanji compound words.

A problem we encounter is the compound word whose constituents consist of the words something like:

大阪/府警/捜査/四/課
(the forth criminal investigation section of Osaka Police Department)
(大阪：Osaka 府警：Police Department 捜査：criminal investigation
四：four 課：section)

We get ((((大阪)(府警))(捜査))((四)(課))) in our method, but the correct one should be(((大阪)(府警))((捜査)((四)(課)))). This error comes from the improper segmentation of the compound word. If we get 四課 as a word, in stead of 四/課, then our method works perfectly.

An expression relating to some kinds of numbers like in this example is troublesome. The same is true for a prefix or a suffix that appears in a compound word. We see about 30% of the errors in our experiment are of this nature. Some post-segmentation strategies on numerical expressions, prefixes and suffixes may be necessary to improve the performance.

Another problem is seen in a compound word like:

日本/記録/保持者 (Japanese record holder)
(日本：Japan 記録：record 保持者：holder)

This is a case where the refinement gets against us. When we are to apply (e) to this, we get the right answer as (((日本)(記録))(保持者)), but in the hybrid method 日本 is a lead word and we get the wrong answer, ((日本)((記録)(保持者))). This type of errors counts about 20% in all the errors.

The rest of the errors come from various reasons. Let us see a typical example.

現行/中/選挙区/制 (current medium constituency system)
(現行：current 中：medium 選挙区：constituency 制：system)

We get the result from our method of (((現行)((中)(選挙区)))(制)). Here, 現行 is estimated to depend semantically on 選挙区, while the correct relation should be 現行→制. The reason for the failure is that we have few co-occurrence of 現行 and 制 on the Internet. When we change 制 to 制度, a word having the same meaning with 制, we get the correct result of ((現行)(((中)(選挙区))(制度))).

5 Conclusion

We have presented a method of analyzing the structure of compound words in Japanese using the strengths of word association among the constituent words. The experimental result that followed indicates that our method sounds better both in the computation time and accuracy.

However, we may have ways to get better results. Since the analysis depends partly on the result of word segmentation, we should refine the segmentation system so that it gives us better and suitable results fit for our analysis. It may be desirable in the structural analysis to incorporate a distance measure that is proven effective in analyzing semantic structure of compound words (Kobayasi et al., 1994) and dependency relations among the phrases in Japanese sentences (Zhang & Ozeki, 1997).

References

Alves, E. (1996). "The selection of the most probable dependency structure in japanese using mutual information". In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pp. 372-374.

Bergh, G., Seppänen A., & Trotta, J. (1998). "Language corpora and the internet: a joint linguistic resource". In Renouf, A. (Ed.): *Explorations in Corpus Linguistics*, Rodopi B.V., Amsterdan, pp. 41-54.

Church, K., & Hanks, P. (1990). "Word association norms, mutual informaion and lexicography". *Computational Linguistics*, 16, 22-29.

Fujii, A., & Ishikawa, T. (1999). "Cross-language information retrieval using compound word translation". In *Proceedings of the 18th International Conference on Computer Processing of Oriental Languages (ICCPOL'99)*, pp. 105-110.

Furugori, T., & Alves, E. (1999). "Disambiguation of syntactic structures using the strength of association in three word dependency relations". *Journal of Quantitative Linguistics*, 6(2), 101-107.

Han, D., Kato, K., & Furugori, T. (2001a). "Automatic segmentation of compound word in Japanese using contextual information". *Technical Report of IEICE. NLC 2001-05*, 29-34. (in Japanese).

Han, D., Wu, H., & Furugori, T. (2001b). "Resolving overlapping ambiguities and selecting correct word sequence in chinese using Internet corpus". *Journal of Natural Language Processing*, 8(3), 107-121.

Hisamitsu, T., & Nitta, Y. (1996). "Analysis of Japanese compound nouns by direct text scanning". In *Proceedings of the 16th International Conference on Computational Linguistics (COLING'96)*, pp. 550-555.

Karov, Y., & Edelman, S. (1998). "Similarity-based word sense disambiguation". *Computational Linguistics*, 24, 41-59.

Kobayasi, Y., Tokunaga, T., & Tanaka H. (1994). "Analysis of Japanese compound nouns using collocational information". In *Proceedings of the 15th International Conference on Computational Linguistics (COLING'94)*, pp. 865-869.

Lauer, M. (1995). "Corpus statistics meet the noun compound: some empirical results". In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pp. 47-54.

Matsumoto, Y et al. (2001). *Morphological analysis system ChaSen version 2.2.8 manual.* http://chasen.aist-nara.ac.jp/chasen/doc/chasen-2.2.8.pdf.

Mihalcea, R., & Moldovan, D. I. (1999). "A method for word sense disambiguation of unrestricted text". In *Proceedings of 37th Annual Meeting of the Association for Computational Linguistics*, pp. 152-158.

Miyahira, T., Watanabe, H., Tazoe, E., Kamiyama, Y., & Takeda, K. (2000). *Internet kikaihonyaku no sekai*. Mainichi Communications, Inc. (in Japanese).

Miyazaki, M. (1984). "Automatic segmentation method for compound words using semantic dependent relationships between words". *Transactions of Information Processing Society of Japan*, 25, 970-979.

(in Japanese).

Miyazaki, M., Ikehara, S., & Yokoo, A. (1993). "Combined word retrieval for bilingual dictionary based on the analysis of compound words". *Transactions of Information Processing Society of Japan*, 34, 743-753. (in Japanese).

Nishino, T., & Fujisaki, T. (1988). "A stochastic parsing of kanji compound words". *Transactions of Information Processing Society of Japan*, 29, 1034-1042. (in Japanese).

Shimohata, S., & Sugio, T. (1997). "Keyword extraction using character type and decomposition of compound word". *Technical Report of IEICE. NLC1997-07*, 13-18. in Japanese).

Takeda, K., & Fujisaki, T. (1987). "Automatic decomposition of kanji compound words using stochastic estimation". *Transactions of Information Processing Society of Japan*, 28, 952-961. (in Japanese).

Utiyama, M., & Itahashi, S. (1992). "Division of Japanese compound nouns using co-occurrence relation". *SIG notes, NL-91, Information Processing Society of Japan*, 47-54. (in Japanese)

Wu, H., & Furugori, T. (1996). "A hybrid disambiguation model for prepositional phrase attachment". *Literary and Linguistic Computing*, 11, 187-192.

Zhang, Y., & Ozeki, K. (1997). "Dependency analysis of Japanese sentences using the statistical property of dependency distance between phrases". *Journal of Natural Language Processing*. 4(2), 3-19. (in Japanese).