Processing Homonyms in the Kana-to-Kanji Conversion

Masahito Takahashi
Fukuoka University
8-19-1, Nanakuma,
Jonan-ku, Fukuoka,
814-01, Japan
takahasi@helio.tl.
fukuoka-u.ac.jp

Tsuyoshi Shinchu
Fukuoka University
8-19-1, Nanakuma,
Jonan-ku, Fukuoka,
814-01, Japan
shinchu@helio.tl.
fukuoka-u.ac.jp

Kenji Yoshimura
Fukuoka University
8-19-1, Nanakuma,
Jonan-ku, Fukuoka,
814-01, Japan
yosimura@tlsun.tl.
fukuoka-u.ac.jp

Kosho Shudo
Fukuoka University
8-19-1, Nanakuma,
Jonan-ku, Fukuoka,
814-01, Japan
shudo@tlsun.tl.
fukuoka-u.ac.jp

Abstract

This paper proposes two new methods to identify the correct meaning of Japanese homonyms in text based on the noun-verb co-occurrence in a sentence which can be obtained easily from corpora. The first method uses the *near co-occurrence data sets*, which are constructed from the above co-occurrence relation, to select the most feasible word among homonyms in the scope of a sentence. The second uses the *far co-occurrence data sets*, which are constructed dynamically from the *near co-occurrence data sets* in the course of processing input sentences, to select the most feasible word among homonyms in the scope of a sequence of sentences. An experiment of *kana*-to-*kanji*(phonogram-to-ideograph) conversion has shown that the conversion is carried out at the accuracy rate of 79.6% per word by the first method. This accuracy rate of our method is 7.4% higher than that of the ordinary method based on the word occurrence frequency.

1 Introduction

Processing homonyms, i.e. identifying the correct meaning of homonyms in text, is one of the most important phases of *kana*-to-*kanji* conversion, currently the most popular method for inputting Japanese characters into a computer. Recently, several new methods for processing homonyms, based on neural networks(Kobayashi,1992) or the co-occurrence relation of words(Yamamoto,1992) , have been proposed. These methods apply to the co-occurrence relation of words not only in a sentence but also in a sequence of sentences. It appears impracticable to prepare a neural network for co-occurrence data large enough to handle 50,000 to 100,000 Japanese words.

In this paper, we propose two new methods for processing Japanese homonyms based on the co-occurrence relation between a noun and a verb in a sentence. We have defined two co-occurrence data sets. One is a set of nouns accompanied by a case marking particle, each element of which has a set of co-occurring verbs in a sentence. The other is a set of verbs accompanied by a case marking particle, each element of which has a set of co-occurring nouns in a sentence. We call these two co-occurrence data sets *near co-occurrence data sets*. Thereafter, we apply the data sets to the processing of homonyms. Two strategies are used to approach the problem. The first uses the *near co-occurrence data sets* to select the most feasible word among homonyms in the scope of a sentence. The aim is to evaluate the possible existence of a *near co-occurrence relation*, or co-occurrence relation between a noun and a verb within a sentence. The second evaluates the possible existence of a *far co-occurrence relation*, referring to a co-occurrence relation among words in different sentences. This is achieved by constructing *far co-occurrence data sets* from *near co-occurrence data sets* in the course of processing input sentences.

2 Co-occurrence data sets

The *near co-occurrence data sets* are defined.

The first *near co-occurrence data set* is the set $\Sigma_{N_{near}}$, each element of which(n) is a triplet consisting of a noun, a case marking particle, and a set of verbs which co-occur with that noun and particle pair in a sentence, as follows:

$$n = (noun, particle, \{(v_1, k_1), (v_2, k_2), \cdots\})$$

In this description, *particle* is a Japanese case marking particle, such as が (nominative case), を (accusative case), or に (dative case), $v_i (i = 1, 2, \cdots)$ is a verb, and $k_i (i = 1, 2, \cdots)$ is the frequency of occurrence of the combination *noun*, *particle* and v_i, which is determined in the course of constructing $\Sigma_{N_{near}}$ from corpora. The following are examples of the elements of $\Sigma_{N_{near}}$.

```
(雨(rain), が(nominative case),
 {(降る(fall),10),(止む(stop),3), ..})
(雨(rain), を(accusative case),
 {(警戒する(take precautions),3), ..})
```

The second *near co-occurrence data set* is the set $\Sigma_{V_{near}}$, each element of which(v) is a triplet consisting of a verb, a case marking particle, and a set of nouns which co-occur with that verb and particle pair in a sentence, as follows:

$$v = (verb, particle, \{(n_1, l_1), (n_2, l_2), \cdots\})$$

In this description, *particle* is a Japanese case marking particle, $n_i (i = 1, 2, \cdots)$ is a noun, and $l_i (i = 1, 2, \cdots)$ is the frequency of occurrence of the combination *verb*, *particle* and n_i. The following are examples of the elements of $\Sigma_{V_{near}}$. $\Sigma_{V_{near}}$ can be constructed from $\Sigma_{N_{near}}$, and vice versa.

```
(降る (fall), が (nominative case),
 {(雨 (rain),10),(雪 (snow),8), ..})
(降る (fall), に (dative case),
 {(九州 (Kyushu),1), ..})
```

3 Processing homonyms in a simple sentence

Using the *near co-occurrence data sets*, the most feasible word among possible homonyms can be selected within the scope of a sentence. Our hypothesis states that the most feasible noun or combination of nouns has the largest number of verbs with which it can co-occur in a sentence.

The structure of an input Japanese sentence written in *kana*-characters can be simplified as follows:

$$N_1 \cdot P_1, N_2 \cdot P_2, \cdots, N_m \cdot P_m, V$$

where $N_i (i = 1, 2, \cdots, m)$ is a noun, $P_i (i = 1, 2, \cdots, m)$ is a particle and V is a verb.

3.1 Procedure

Following is the procedure for finding the most feasible combination of words for an input *kana*-string which has the above simplified Japanese sentence structure. This procedure can also accept an input *kana*-string which does not include a final position verb.

Step1 Let $m = 0$ and $T_i = \varepsilon (i = 1, 2, \cdots)$.

Step2 If an input *kana*-string is null, go to Step4. Otherwise read one block of *kana*-string, that is $N \cdot P$ or V, from the left side of the input *kana*-string. And delete the one block of *kana*-string from the left side of the input *kana*-string.

Step3 Find all homonymic *kanji*-variants $W_k (k = 1, 2, \cdots)$ for the *kana*-string N or V which is read in Step2.
Increase m by 1.
For each $W_k (k = 1, 2, \cdots)$:

1. If W_k is a noun, retrieve (W_k, P, V_k) from the *near co-occurrence data set* $\Sigma_{N_{near}}$ and add the doublet (W_k, V_k) to T_m.
2. If W_k is a verb, add the doublet $(W_k, \{(W_k, 0)\})$ to T_m.

Go to Step2.

Step4 From $T_i (i = 1, 2, \cdots, m)$, find the combination:

$$(W_1, V_1)(W_2, V_2), \cdots, (W_m, V_m)$$
$$(W_i, V_i) \in T_i (i = 1, 2, \cdots, m)$$

which has the largest value of $| \bigcap(V_1, V_2, \cdots, V_m) |$. Where the function $\bigcap(V_1, V_2, \cdots, V_m)$ is defined as follows.

$$\bigcap(V_1, V_2, \cdots, V_m) = \{(v, \sum_{i=1}^{m} k_i) \mid$$
$$(v, k_1) \in V_1 \wedge \cdots \wedge (v, k_m) \in V_m\}$$

And $| \bigcap(V_1, V_2, \cdots, V_m) |$ is defined:

$$| \bigcap(V_1, V_2, \cdots, V_m) | = \sum_{(v,k) \in \bigcap(V_1, V_2, \cdots, V_m)} k$$

The sequence of words $W_1, W_2, \cdots, W_m$ is the most feasible combination of homonymic *kanji*-variants for the input *kana*-string.

3.2 An example of processing homonyms in a simple sentence

Following is an example of homonym processing using the above procedures.

For the input *kana*-string

"かわにはしを (*kawa ni hashi o*)"

"かわ (*kawa*)" means a river and "はし (*hashi*)" means a bridge. "かわ (*kawa*)" and "はし (*hashi*)" both have homonymic *kanji*-variants:

homonyms of "かわ (*kawa*)" :	川 (river)
	皮 (leather)
homonyms of "はし (*hashi*)" :	橋 (bridge)
	箸 (chopsticks)

The near co-occurrence data for "川 (river)" and "皮 (leather)" followed by the particle "に (dative case)" and the near co-occurrence data for "橋 (bridge)" and "箸 (chopsticks)" followed by the particle "を (accusative case)" are shown below.

```
(川 (river), に,{(行く (go),8),
             (架ける (build),6),
             (落す (drop),5)})
(皮 (leather), に,{(塗る (paint),6),
               (触る (touch),3)})
(橋 (bridge), を,{(渡る (walk across),9),
              (架ける (build),7),
              (落す (drop),4)})
(箸 (chopsticks), を,{(使う (use),7),
                  (落す (drop),3)})
```

Following the procedure, the resultant frequency values are as follows:

川に	橋を	22{ 架ける, 落す }
川に	箸を	8{ 落す }
皮に	橋を	0{}
皮に	箸を	0{}

Therefore, the most feasible combination of words is "川 (river) に 橋 (bridge) を."

4 An experiment on processing homonyms in a simple sentence

4.1 Preparing a dictionary and a co-occurrence data file

4.1.1 a noun file

A noun file including 323 nouns, which consists of 190 nouns extracted from text concerning current topics and their 133 homonyms, was prepared.

4.1.2 a co-occurrence data file

A co-occurrence data file was prepared. The record format of the file is specified as follows:

> [noun, case marking particle, verb, the frequency of occurrence]

where case marking particle is chosen from 8 kinds of particles, namely, “が”,“を”,“に”,“へ”,“と”,“から”,“より”,“で”.

It includes 25,665 records of co-occurrence relation (79 records per noun) for the nouns in the noun file by merging 11,294 records from EDR Co-occurrence Dictionary(EDR,1994) with 15,856 records from handmade simple sentences.

4.1.3 an input file and an answer file

An input file for an experiment, which includes 1,129 simple sentences written in *kana* alphabet, and an answer file, which includes the same 1,129 sentences written in *kanji* characters, were prepared. Here, every noun of the sentences in the files was chosen from the noun file.

4.1.4 a word dictionary

A word dictionary, which consists of 323 nouns in the noun file and 23,912 verbs in a Japanese dictionary for *kana*-to-*kanji* conversion [1], was prepared. It is used to find all homonymic *kanji*-variants for each noun or verb of the sentences in the input file.

4.2 Experiment results

An experiment on processing homonyms in a simple sentence was carried out. In this experiment, *kana*-to-*kanji* conversion was applied to each of the sentences, or the input *kana*-strings, in the above input file and the *near co-occurrence data sets* were constructed from the above co-occurrence data file. Table1 shows the results of *kana*-to-*kanji* conversion in the following two cases. In the first case, an input *kana*-string does not include a final position verb. It means that each verb of the *kana*-strings in the input file is neglected. In the second case, an input *kana*-string includes a final position verb. The experiment has shown that the conversion is carried out at the accuracy rate of 79.6% per word, where the conversion rate is 93.1% per word, in the first case. In the same way, the accuracy rate is 93.8% per word, where the conversion rate is 14.5% per word, in the second case. And then, we also conducted the same experiment by using the method based on the word occurrence frequency to compare our method with an ordinary method. It has shown that the accuracy rate is 72.2% per word in the first case, and 77.8% per word in the second case. We can find the accuracy rate by our method is 7.4% higher in the first case and 16.9% higher in the second case compared with the ordinary method. It is clarified that our method is more effective than the ordinary method based on the word occurrence frequency.

5 An approximation of the far co-occurrence relation

We can approximate the *far co-occurrence relation*, which is co-occurrence relation among words in a sequence of sentences, from *near co-occurrence data sets*. The *far co-occurrence data sets* are described as follows:

$$\Sigma_{N_{far}} = \{(n_1, t_1), (n_2, t_2), \cdots, (n_{l_n}, t_{l_n})\}$$

$$\Sigma_{V_{far}} = \{(v_1, u_1), (v_2, u_2), \cdots, (v_{l_v}, u_{l_v})\}$$

where $n_i(i = 1, 2, \cdots, l_n)$ is a noun, t_i is the priority value of n_i, $v_i(i = 1, 2, \cdots, l_v)$ is a verb and u_i is the priority value of v_i.

The procedure for producing the *far co-occurrence data sets* is:

Step1 Clear the *far co-occurrence data sets.*

$$\Sigma_{N_{far}} = \epsilon$$

$$\Sigma_{V_{far}} = \epsilon$$

Step2 After each fixing of noun N among homonyms in the process of *kana*-to-*kanji* conversion, renew the *far co-occurrence data sets* $\Sigma_{N_{far}}$ and $\Sigma_{V_{far}}$ by following these steps:

1. Change all priority values of $t_i(i = 1, 2, \cdots, l_n)$ in the set $\Sigma_{N_{far}}$ to $f(t_i)$ (for example, $f(t_i) = 0.95t_i$). This process is intended to decrease priority with the passage of time.

Table 1: Experiment results on processing homonyms in a simple sentence

	For sentences without a verb	For sentences with a verb
Conversion rate per sentence	1053/1129 (93.3%)	166/1129 (14.7%)
Accuracy rate per sentence	663/1053 (63.0%)	136/166 (81.9%)
Conversion rate per word	2155/2315 (93.1%)	500/3444 (14.5%)
Accuracy rate per word	1716/2155 (79.6%)	469/500 (93.8%)

[1] This dictionary was made by AI Soft Co.

2. Change all priority values of $u_i(i = 1, 2, \cdots, l_v)$ in the set $\Sigma_{V_{far}}$ to $f(u_i)$ as well.
3. Let N be the noun determined in the process of *kana*-to-*kanji* conversion. Find all
$$(v_i, k_i)(i = 1, 2, \cdots, q)$$
which co-occur with the noun N followed by any particle, in the *near co-occurrence data set* $\Sigma_{N_{near}}$. Add new elements
$$(v_i, g(k_i))(i = 1, 2, \cdots, q)$$
to the set $\Sigma_{V_{far}}$. If an element with the same verb v_i already exists in $\Sigma_{V_{far}}$, add the value $g(k_i)$ to the priority value of that element instead of the new element. Here, $g(k_i)$ is a function for converting frequency of occurrence to priority value. For example,
$$g(k_i) = 1 - (1/k_i)$$
4. Let v_i be the verb described in the previous step. Find all
$$(n_j, l_j)(j = 1, 2, \cdots, q)$$
which co-occur with the verb v_i and any particle in the *near co-occurrence data set* $\Sigma_{V_{near}}$. Add new elements
$$(n_j, h(k_i, l_j))(j = 1, 2, \cdots, q)$$
to the set $\Sigma_{N_{far}}$. If an element with the same noun n_j already exists in $\Sigma_{N_{far}}$, add the value $h(k_i, l_j)$ to the priority value of that element instead of the new element. Here, $h(k_i, l_j)$ is a function for converting frequency of occurrence to priority value. For example,
$$h(k_i, l_j) = g(k_i)(1 - (1/l_j))$$

6 Processing homonyms in a sequence of sentences

Using the *far co-occurrence data sets* defined in the previous section, the most feasible word among homonyms can be selected in the scope of a sequence of sentences according to the following two cases.

Case1 An input word written in *kana*-characters is a noun.

Case2 An input word written in *kana*-characters is a verb.

6.1 Procedure for case1

Step1 Find set S_n:
$$S_n = \{(N_1, T_1), (N_2, T_2), \cdots\}$$
where $N_i(i = 1, 2, \cdots)$ is a homonymic *kanji*-variant for the input word written in *kana*-characters and T_i is the priority value for homonym N_i, which can be retrieved from the *far co-occurrence data set* $\Sigma_{N_{far}}$.

Step2 The noun N_i which has the greatest T_i priority value in S_n is the most feasible noun for the input word written in *kana*-characters.

6.2 Procedure for case2

Step1 Find set S_v:
$$S_v = \{(V_1, U_1), (V_2, U_2), \cdots\}$$
Here, $V_j(j = 1, 2, \cdots)$ is a homonymic *kanji*-variant for the input word written in *kana*-characters and U_j is the priority value for homonym V_j , which can be retrieved from the *far co-occurrence data set* $\Sigma_{V_{far}}$.

Step2 The verb V_j which has the greatest U_j priority value in S_v is the most feasible verb for the input word written in *kana*-characters.

7 Conclusion

We have proposed two new methods for processing Japanese homonyms based on the co-occurrence relation between a noun and a verb in a sentence which can be obtained easily from corpora. Using these methods, we can evaluate the co-occurrence relation of words in a simple sentence by using the *near co-occurrence data sets* obtained from corpora. We can also evaluate the co-occurrence relation of words in different sentences by using the *far co-occurrence data sets* constructed from the *near co-occurrence data sets* in the course of processing input sentences. The *far co-occurrence data sets* are based on the proposition that it is more practical to maintain a relatively small amount of data on the semantic relations between words, being changed dynamically in the course of processing, than to maintain a huge universal "thesaurus" data base, which does not appear to have been built successfully.

An experiment of *kana*-to-*kanji* conversion by the first method for 1,129 input simple sentences has shown that the conversion is carried out in 93.1% per word and the accuracy rate is 79.6% per word. It is clarified that the first method is more effective than the ordinary method based on the word occurrence frequency.

In the next stage of our study, we intend to evaluate the second method based on the *far co-occurrence data sets* by conducting experiments.

References

Kobayashi, T., et al. 1992. Realization of *Kana*-to-*Kanji* Conversion Using Neural Networks. *Toshiba Review*, Vol.47, No.11, pages 868-870, Japan.

Yamamoto, K., et al. 1992. *Kana*-to-*Kanji* Conversion Using Co-occurrence Groups. *Proc. of 44th Conference of IPSJ*, 4p-11, pages 189-190, Japan.

EDR. 1994. *Co-occurrence Dictionary Ver.2*, TR-043, Japan.