Combination of an Automatic and an Interactive Disambiguation Method

Masaya Yamaguchi, Takeyuki Kojima,
Nobuo Inui, Yoshiyuki Kotani and **Hirohiko Nisimura**
Department of Computer Science, Tokyo University of Agriculture and Technology,
Nisimura, Kotani unit, 2-24-16 Naka-cho, Koganei, Tokyo, Japan

Abstract

In natural language processing, many methods have been proposed to solve the ambiguity problems. In this paper, we propose a technique to combine a method of interactive disambiguation and automatic one for ambiguous words. The characteristic of our method is that the accuracy of the interactive disambiguation is considered. The method solves the two following problems when combining those disambiguation methods: (1) when should the interactive disambiguation be executed? (2) which ambiguous word should be disambiguated when more than one ambiguous words exist in a sentence? Our method defines the condition of executing the interaction with users and the order of disambiguation based on the strategy where the accuracy of the result is maximized, considering the accuracy of the interactive disambiguation and automatic one. Using this method, user interaction can be controlled while holding the accuracy of results.

1 Introduction

In natural language processing, many methods have been proposed to solve the ambiguity problems(Nagao and Maruyama, 1992). One of those technique uses interactions with users, because it is difficult to make all the knowledge for disambiguation beforehand. That technique is classified into two types according to the condition of executing user interaction. One type(TypeA) is that the disambiguation system executes interactions(Blanchon et al., 1995), (Maruyama and Watanabe, 1990), (Yamaguchi et al., 1995). Another type(TypeB) is that users executes interactions(D.Brawn and Nirenburg, 1990), (Muraki et al., 1994). In this paper, TypeA will be adopted because TypeB gives users more trouble than TypeA does. For example, in TypeB, a user may have to find where is wrongly analyzed in input sentences.

In TypeA, the two following conditions must be determined: (1) when should interactive disambiguation be executed? (2) which ambiguous words should be disambiguated when more than one ambiguous word exist in a sentence? Considering the accuracy of the analyzed result, they should be decided by both the accuracy of the interactive disambiguation and that of the automatic disambiguation. The traditional methods did not considered the accuracy of the interactive disambiguation. For instance, the accuracy of the analyzed result may decrease in spite of executing the user interaction if the accuracy of the interactive disambiguation is low.

In this paper, we propose the method to combine the interactive disambiguation and the automatic one, considering each accuracy. The method allows the disambiguation system to maximize the accuracy of the analyzed result. This paper focuses on the ambiguity caused by ambiguous words that have more than one meaning. Section 2 represents preconditions for disambiguation. In Section 3, we describe the condition of executing the interactive disambiguation. Section 4 shows the procedure that decides the order of disambiguation. The performance of the method is discussed by the result of the simulation under assuming the both accuracy of the interactive disambiguation and the automatic one.

2 Preconditions for Disambiguation

This section describes preconditions for disambiguation and methods of the disambiguation.

In this paper, the disambiguation for ambiguous words means that all ambiguous ones in an input sentence are disambiguated. Describing it formally, the disambiguation is to decide one element of the following MS.

$$MS = M_1 \times M_2 \times \ldots \times M_l,$$

where an input sentence contains l ambiguous words. M_i means the set of meanings in the ambiguous word w_i.

Each disambiguation method has preconditions as follows:

Interactive Disambiguation

- In the interaction, the system shows explanations for each meaning of an ambiguous word to a user, who selects one explanation from them.

- The system can calculate the probability where a user selects the right explanation.

Automatic Disambiguation

- The occurrence probabilities for each candidate can be calculated for preference.
- The result is the candidate with the maximum occurrence probability.

To show the information mentioned above, candidates are expressed by the tree in Figure 1. This tree is an example in the case that an input sentence is "I saw a star.", which contains two ambiguous words 'see' and 'star' and each word has two meanings.

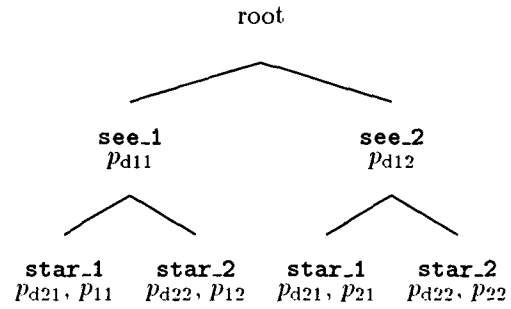

Figure 1: An example of the tree of candidates

The depth of the tree expresses the order of disambiguation. In Figure 1, the ambiguities are resolved in the order from 'see' to 'star'. The occurrence probability is calculated at each leaf node by the automatic disambiguation method. For example, p_{11} expresses the probability for the candidate {see_1, star_1}. Furthermore, the accuracy of interaction is also calculated at the leaf node by the interactive disambiguation method. p_{d21} is the probability where the meaning of 'star' is 'star_1' and the system shows explanations of 'star_1', 'star_2' for 'star' to a user and (s)he selects the explanation of 'star_2'. At Nodes besides leaf ones, only the accuracy of interaction is calculated.

3 The Condition of Executing the Interactive Disambiguation

3.1 Basic Idea

At each node besides leaf ones, the disambiguation system decides which disambiguation method is used. Basically, the interactive disambiguation is executed when its accuracy is higher than the accuracy of the automatic disambiguation. First of all, let us consider the case where an input sentence contains one ambiguous word that has n meanings. Figure 2 shows the tree of candidates for this case.

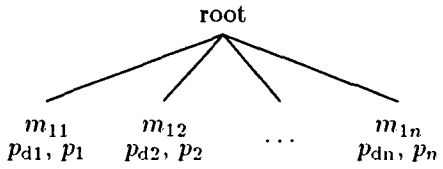

Figure 2: An example of the tree of candidates for one ambiguous word in an input sentence

The accuracy of the interactive disambiguation P_{intr} and that of the automatic disambiguation P_{auto} are defined as follows:

$$P_{\text{intr}} = \sum_{i}^{n} p_{di} p_i$$

$$P_{\text{auto}} = \max_{i} p_i$$

The interactive disambiguation is executed, when the following condition is satisfied.

$$P_{\text{intr}} > P_{\text{auto}}$$

Considering the condition more carefully, the accuracy of the interactive disambiguation is influenced by the explanations that are shown to users. Thus the accuracy may be improved by limiting to show some explanations to users. For example, this may be caused when the accuracy of m_{11} is very low and a user may select m_{11} wrongly by the higher similarity of the explanation for m_{11} to other explanations. The automatic disambiguation corresponds to showing only one explanation to users in the interactive disambiguation. Therefore the condition of executing the interactive disambiguation can be defined as the exceptional case of the limitation.

3.2 The Accuracy at a Node

In the case that the number of ambiguous words is one as Figure 2, the accuracy of the deeper nodes below the root node needs not to be decided because they are leaf nodes. When more than two ambiguous words exist in an input sentence, a node may often have one that is not a leaf one. To calculate the accuracy of such a node, it is necessary to determine what kind of disambiguation will be executed at the deeper nodes. For instance, the disambiguation system has to fix each accuracy of node 'see_1' and 'see_2' in Figure 1 to calculate the accuracy of the root node. Therefore, the definition of the accuracy at any node i is the recursive one. The accuracy of the interactive disambiguation $P_{\text{intr}}(i)$ and that of the automatic disambiguation $P_{\text{auto}}(i)$ at node i is defined as follows:

$$P_{\text{intr}}(i) = \sum_{m \in M} p_{\text{d}}(m|M) \times P_{\text{r}}(m) \quad (1)$$

$$P_{\text{auto}}(i) = \max_{m \in M}(P_{\text{r}}(m)) \quad (2)$$

where M is the set of the node directly under node i, $p_{\text{d}}(m|M)$ is the accuracy of the interactive disambiguation at node m, that is, the probability that a user selects m provided that the system shows explanations for all the elements of M to him(her).

$P_{\text{r}}(m)$ is the accuracy at node m and the definition is as follows:

$$P_{\text{r}}(m) = \begin{cases} P_{\text{intr}}(m) \\ \text{(if the interactive disambiguation is executed at node } m\text{)} \\ P_{\text{auto}}(m) \\ \text{(if the automatic disambiguation is executed at node } m\text{)} \\ p_{\text{occur}}(m) \quad \text{(if } m \text{ is a leaf node)} \end{cases}$$

where $p_{\text{occur}}(m)$ is the occurrence probability of the candidate that includes nodes between the root node and node m.

When the following condition is satisfied, the interactive disambiguation is executed at node i.

$$P_{\text{intr}}(i) > P_{\text{auto}}(i) \quad (3)$$

3.3 The Limitation of Explanations

In user interaction, the presentation of many explanations gives users trouble to select one explanation. So it is desirable that the disambiguation system shows fewer explanation to users, if possible. In this section, we describe the condition where the number of explanations is limited without losing the accuracy of the analyzed result.

By formula (1), the accuracy of the interactive disambiguation P'_{inter} in the case of limiting the set of explanations M' is defined as follows:

$$P'_{\text{intr}}(i) = \begin{cases} \max_{M'} \sum_{m \in M - M'} p_{\text{d}}(m|M - M')P_r(m) \\ \qquad \text{if } |M - M'| > 1 \\ P_{\text{r}}(t) \qquad \text{if } |M - M'| = 1 \end{cases}$$

If formula (4) is satisfied, the set of the explanation M' is not shown to users in the interaction at node i.

$$P_{\text{intr}}(i) \leq P'_{\text{intr}}(i) \quad (4)$$

Furthermore, if $|M - M'| = 1$, then the automatic disambiguation is executed at node i. Therefore, formula (4) implies formula (3).

4 Determination of the Order of Disambiguation

4.1 Procedure

Up to here, we have discussed P_{intr} and P_{auto} under a certain order of disambiguation. In this section, we describe a procedure to decide the order of disambiguation where the accuracy is maximum.

The accuracy of the analyzed result may be different in each order of disambiguation. This is the reason that the disambiguation of one ambiguous word leads to constrain the meaning of other ambiguous ones. Therefore, the contents of the interaction may differ from each order of disambiguation. The order with the maximum accuracy is obtained in the following procedure:

1. Calculating each occurrence probability of candidate for the analyzed result by the automatic disambiguation method.
2. Obtaining the accuracy in each order of disambiguation based on the method described in the previous sections.
3. Disambiguating by the order with the maximum accuracy.

4.2 Example

In this section, we illustrate the determination of executing the interactive disambiguation and the order of disambiguation. The values at leaf nodes are the occurrence probabilities. The accuracy of the interactive disambiguation is 0.9 at the any nodes. Since the number of ambiguous words is two, the number of the order of disambiguation is 2! as shown in Figure 3, 4.

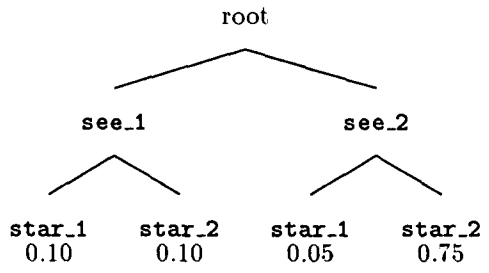

Figure 3: An example of the order of disambiguation(1)

To begin with, we intend to calculate what kind of disambiguation is executed at node '**star_1**' and '**star_2**', in Figure 3. By formula (1), (2), P_{intr}(**see_1**) and P_{auto}(**see_1**) are as follows (since both ambiguous words have two meanings, $P'_{intr}(i)$ = $P_{auto}(i)$):

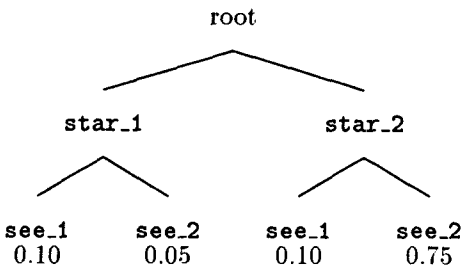

Figure 4: An example of the order of disambiguation(2)

$$\begin{aligned} P_{intr}(\textbf{see\_1}) &= 0.9 \times (0.75 + 0.05) \\ &= 0.72 \\ P_{auto}(\textbf{see\_1}) &= \max(0.75, 0.05) \\ &= 0.75 \end{aligned}$$

Because of $P_{intr}(\textbf{see\_1}) < P_{auto}(\textbf{see\_1})$, the automatic disambiguation is executed at node **see_1**. On the other hand, at node **see_2**, $P_{intr}(\textbf{see\_2})$ and $P_{auto}(\textbf{see\_2})$ are as follows:

$$\begin{aligned} P_{intr}(\textbf{see\_2}) &= 0.18 \\ P_{auto}(\textbf{see\_2}) &= 0.10 \end{aligned}$$

$P_{intr}(\textbf{see\_2}) > P_{auto}(\textbf{see\_2})$ is satisfied. So the system interacts with users at this node.

By the result of the above, $P_{intr}(root)$ and $P_{auto}(root)$ are as follows:

$$\begin{aligned} P_{intr}(root) &= 0.9(P_r(\textbf{see\_1}) + P_r(\textbf{see\_2})) \\ &= 0.9(P_{auto}(\textbf{see\_1}) + P_{intr}(\textbf{see\_2})) \\ &= 0.9(0.75 + 0.18) = 0.837 \\ P_{auto}(root) &= \max(P_r(\textbf{see\_1}), P_r(\textbf{see\_2})) \\ &= \max(0.75, 0.18) = 0.75 \end{aligned}$$

Therefore, the interactive disambiguation is executed at the root node because of $P_{intr}(root) > P_{auto}(root)$, and $P_r(root) = 0.837$.

Next, let us explain the case of Figure 4. Calculating the same way as Figure 3, the interactive disambiguation is executed in any node besides leaf ones, and $P_{intr}(root)$, $P_{auto}(root)$ are as follows:

$$\begin{aligned} P_{intr}(root) &= 0.9(P_r(\textbf{star\_1}) + P_r(\textbf{star\_2})) \\ &= 0.9(P_{intr}(\textbf{star\_1}) + P_{intr}(\textbf{star\_2})) \\ &= 0.9(0.765 + 0.135) = 0.81 \\ P_{auto}(root) &= \max(P_r(\textbf{star\_1}), P_r(\textbf{star\_2})) \\ &= \max(0.10, 0.75) = 0.75 \end{aligned}$$

Therefore, $P_{intr}(root) > P_{auto}(root)$, and $P_r(root)$ becomes 0.81. Comparing with $P_r(root)$ of each order, $P_r(root)$ of Figure 3 is greater than that of Figure 4. Thus the system interacts with users against 'see' in the first place.

5 Experiments

We applied the proposed method(abbreviated as MP) to the disambiguation of trees of candidates that are made for experiments, and compared it with the method (abbreviated as MA) that executes interaction in all nodes.

We set the following properties to the tree of candidates.

- the number of ambiguous words included in an input sentence
- the number of meanings in an ambiguous word
- the occurrence probability of candidates

To assign an occurrence probability to each candidate, a random value is given to each candidate above all, and each value is divided by the sum of values given to all candidates.

Figure 5, 6 show the accuracy at the root node and the number of interaction, respectively. In these figures, a mark '+' indicates results of MP. Each of them is the average of 300 trees. A mark '*' indicates results of MA. Because MA does not prescribe the order of disambiguation, the result of each tree is the average of all the orders.

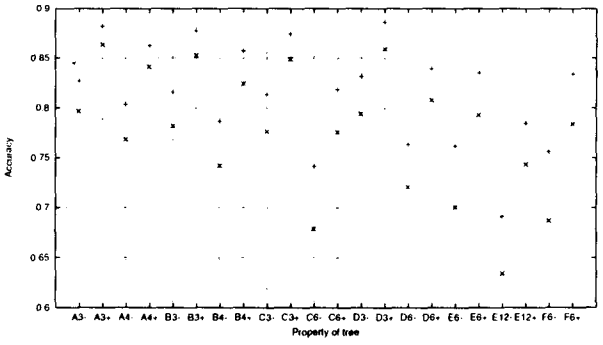

Figure 5: The accuracy of MP, MA

The horizontal axis means the property of the tree. Each Alphabet in the value of the horizontal axis stands for the number of ambiguous words in a tree and the number of meanings of a word as follows:

A:	2×4	**D:**	$2 \times 4 \times 4$
B:	$2 \times 2 \times 4$	**E:**	$2 \times 2 \times 4 \times 4$
C:	$2 \times 2 \times 2 \times 4$	**F:**	$2 \times 2 \times 2 \times 4 \times 4$

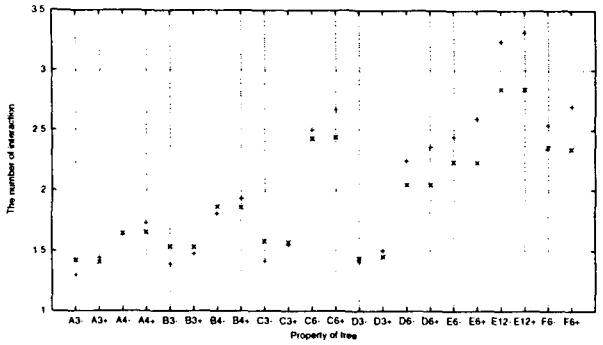

Figure 6: The number of interaction of MP, MA

For instance, '2 x 4' shows that there are two ambiguous words in a tree and one ambiguous word has two meanings and another word has four meanings.

The number in the value of the x-axis represents the number of the candidate whose occurrence probability is not zero. Two marks, '+' and '-' mean that the accuracy of interaction is 0.9, 0.85 respectively.

6 Discussion

6.1 The Accuracy of Disambiguation

The effect of the proposed method on the accuracy is expressed by the difference of distributions of two marks, '+' and '*' in Figure 5. This shows that the accuracy of the proposed method is better than that of MA in any property of tree. Table 1 (the line of 'Accuracy') shows the minimum, maximum, and average values of the ratio of improved accuracy (RIA). The definition of RIA is shown as follows:

$$RIA = \frac{ac_p - ac_a}{1.0 - ac_a}$$

where ac_p, ac_a is the accuracy the result by MP and MA respectively.

Table 1: Summary of the results

	Minimum	Maximum	Average
Accuracy	0.14	0.23	0.18
Interaction	−0.06	0.12	0.03

6.2 The Number of Interaction

The number of interaction may increase on the condition that the accuracy of the analyzed result is maximized. In this section, the degree of the increase will be estimated by comparing the number of interaction of MP with that of MA. For this purpose, 'RII' is defined as follows:

$$RII = \frac{n_p - n_a}{n_w}$$

where n_p, n_a is the number of interaction by MP and MA respectively, n_w is the number of ambiguous words in an input sentence. RII represents the ratio of the increase in the number of interaction per ambiguous word. Table 1(the line of 'Interaction') shows the minimum, maximum, and average of RII.

To reduce the number of interaction, the automatic disambiguation is executed instead of executing the interactive disambiguation, estimating the loss of the accuracy $L(i)$ in node i. $L(i)$ is defined as follows:

$$L(i) = P_r(i) - P_{auto}(i)$$

The proposed method will allow the system to reduce the number of interaction, by considering L(i) in each node.

7 Conclusion

We have proposed the method of combining the interactive disambiguation and the automatic one. The characteristic of our method is that it considers the accuracy of the interactive disambiguation. This method makes three following things possible:

- selecting the disambiguation method that obtains higher accuracy
- limiting explanations shown to users
- obtaining the order of disambiguation where the accuracy of the analyzed results is maximized.

References

Herve' Blanchon, K. Loken-Kim, and T. Morimoto. 1995. An interactive disambiguation module for English natural language utteracances. In *Proceedings of NLPRS'95*, pages 550–555.

Ralf D.Brawn and Sergei Nirenburg. 1990. Human–computer interaction for semantic disambiguation. In *Proceedings of COLING-90*, pages 42–47.

H. Maruyama and H. Watanabe. 1990. An interactive Japanese parser for machine translation. In *Proceedings of COLING-90*, pages 257–262.

K. Muraki, S. Akamine, K. Satoh, and S. Ando. 1994. TWP: How to assist English production on Japanese word processor. In *Proceedings of COLING-94*, pages 847–852.

K. Nagao and H. Maruyama. 1992. Ambiguities and their resolution in natural language processing. *Journal of IPSJ*, 33(7):741–745.

M. Yamaguchi, N. Inui, Y. Kotani, and H. Nisimura. 1995. The design and experiment of an evaluation function for user interaction cost in the interactive semantic disambiguation. In *Proceedings of HCI'95*, pages 285–290.