Rule- and Case-Based Adaptive Knowledge Base and Its Application to Japanese-to-Braille Translation

SATOSHI ONO TAKASHI YAMASAKI SHIGERU NAKAYAMA
Department of Information and Computer Science,
Faculty of Engineering, Kagoshima University
1-21-40 Korimoto, Kagoshima, 890-0065, JAPAN

http://www.ics.kagoshima-u.ac.jp/˜ono/

Abstract: This paper proposes an adaptive knowledge base (AKB) involving two knowledge representations – rule and case. Combining rules and cases makes it possible to solve problems accurately and quickly, and to acquire new cases from problem-solving results. The proposed AKB does not require manually adjustment of the thresholds and provides higher qualified solutions than an existing method with the same knowledge source. This paper also proposes a Japanese-to-Braille translation system which uses the adaptive knowledge base as mentioned above. Experimental results have showed that the threshold adjustment reduces segmentation errors, and that the proposed system reaches almost the same translation quality as the most popular software on the market.

Key–Words: Adaptive knowledge base, case-based reasoning, rule-based reasoning, Japanese-to-Braille translation

1 Introduction

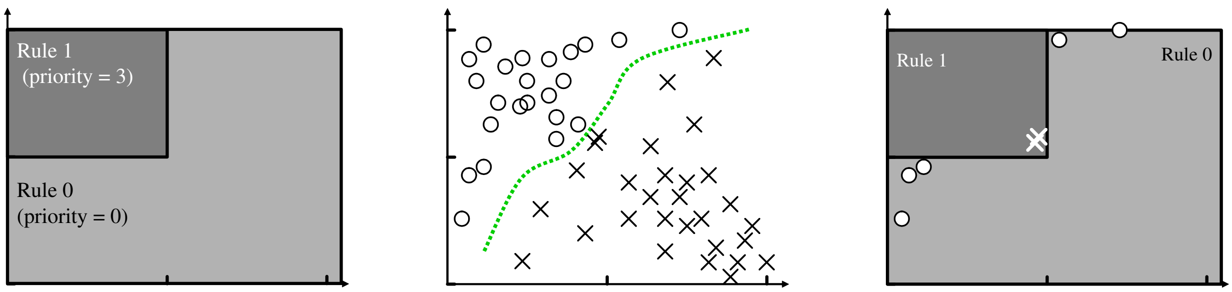
Rules and cases are valuable knowledge representations that mutually supplement drawbacks of each. While rules embody understanding that has been codified over the years by experts, cases contain the knowledge of a domain in a relatively unprocessed form. Rules are appropriate for representing general domain knowledge and cases are appropriate for representing the exceptions of that knowledge [1, 2]. By combining rules and cases, it is possible to solve problems accurately and quickly, and to acquire new cases from problem-solving results[3].

In previous methods using both rules and cases[3, 4, 5], rule-based reasoning (RBR)[6] is performed first, and then case-based reasoning (CBR) [1, 2, 7] is performed. That is, rules are taken to be approximately correct to obtain a preliminary answer for a given problem, and to index cases. This part of the method solves a problem tentatively, but also chooses an exceptional case set. Retrieving the chosen cases, instead of all cases, reduces the processing time of case application, consequently a drawback of CBR. In the previous methods, only a common threshold of similarity is used to judge which exceptional cases should be applied. Although the threshold is manually adjusted, such adjusting is difficult because the appropriate threshold differs with each exceptional case.

In this paper, we propose an adaptive knowledge base (AKB), which is composed of a rule base, an indexed case base and a method for adjusting cases' thresholds. The proposed method eliminates the manual adjustment of the threshold and provides higher qualified solutions than the existing methods even from the same knowledge source. In AKB, each case has a threshold in order to reuse exceptional cases that the existing methods cannot reuse because of the unified threshold. The thresholds are automatically adjusted after case acquisition . Providing a threshold for each case makes it easier to automatically adjust thresholds. Moreover, adjusting the threshold for each case increases the opportunity of reuse and decreases the risk of incorrect use.

This paper also proposes a Japanese-to-Braille translation system AJ2B which uses AKB as mentioned above. Japanese-to-Braille translation is a task involving two procedures – sentence segmentation and kanji-to-kana conversion. Accurate automatic translation is difficult due to the ambiguous, complicated rules peculiar to Japanese-to-Braille translation and Japanese language itself. A Japanese sentence is a string of characters concatenated without blanks, so spaces must be inserted between words to get a proper interpretation. Kanji must be converted to kana, because Braille characters expressing Japanese correspond only to kana.

In Japanese-to-Braille translation, rules are used to represent general translation rules derived from advisory resources such as "Ten-yaku no Tebiki" ("A Guide to Braille Translation") and cases are used to represent exceptional examples derived from "Braille



(a) Rule-based reasoning (2 rules). (b) Case-based reasoning (52 cases). (c) AKB (Combining rules and cases) (2 rules and 7 cases).

Fig. 1: Rule-based reasoning, case-based reasoning, and AKB.

expression dictionary" and existing Braille documents. Applying AKB to Japanese-to-Braille translation, the proposed system can acquire new cases from existing Braille documents and users' revisions on interactive interface.

Experimental results show that the case acquisition can reduce errors, that the threshold adjustment reduces segmentation errors, and that the proposed system reaches almost the same translation quality as the most popular software on the market called "Extra Ver.4.0."

2 Proposed Adaptive Knowledge Base

2.1 Overview

In an attribute space, rules are represented as hyperrectangles as shown in Fig. 1(a), and cases are represented as points whose coverage areas become Voronoi diagram as shown in Fig. 1(b). Although RBR quickly solves problems by applying a few rules, irregular or exceptional problems are difficult for RBR to solve. CBR can accurately solve these problems by applying many cases. Unfortunately, processing time increases proportionally to the number of cases used.

AKB utilizes rules to represent general knowledge, and cases to represent exceptional knowledge. This causes the knowledge base to maintain only negative cases, which have different operators from rules that they belong to as shown in Fig. 1(c). Positive cases, which have the same operator as rules that they belong to, are used only for learning process, not for solving problems. Consequently, the proposed knowledge base solves problems more accurately than RBR and more quickly than CBR because of less amount of knowledge than CBR.

In recent years, non-linear learning methods have been proposed and applied to various real-world problems. For example, Support Vector Machine (SVM)[8] is one of the effective supervised learning methods. SVM simultaneously minimizes the empirical classification error and maximizes the geometric margin.

Compared with SVM and other learning methods, the characteristics of the AKB are as follows:

- AKB is applicable not only to classification task but also to any kind of problem solving, while SVM is mainly for classification.
- AKB attaches greater importance to exceptional cases which most of the existing learning methods treat as noises.

2.2 Problem Solving Process

The fundamental idea of combining rules and cases is to apply the rules to a target problem to produce a draft solution; but if the target problem is judged to be compellingly similar to a known exceptional case of the rules, then the exceptional case is applied rather than the rules [3, 4, 5]. The idea above is therefore realized through the following procedure:

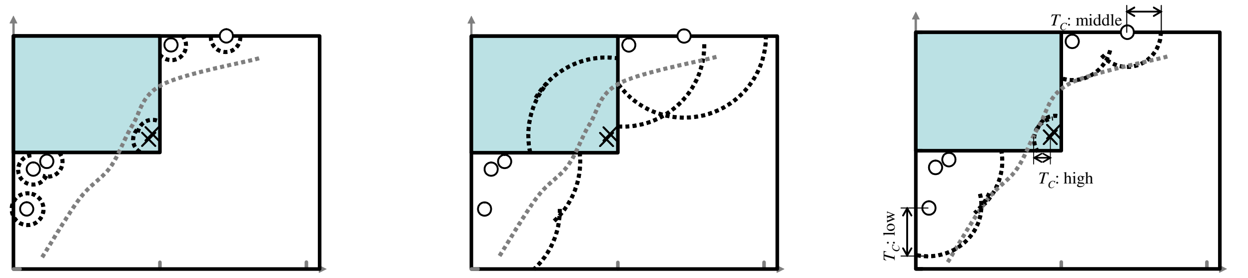
Step 1: Use the rules to select an operator to apply.

Step 2: Search for exceptional cases that would contradict this choice of operator, stopping if and when a compelling case is found.

Step 3: If a compelling case was found, apply the operator it suggests, else proceed to apply the operator suggested by the rules.

2.3 Threshold Adjustment

In the existing method using both rules and cases, the case coverage areas become hypersphere whose radiuses are decided by the unified threshold T_C in the attribute space. High T_C prevents incorrect case application, but it also decreases the chances to reuse cases correctly are also decreased. In contrast, low T_C makes case coverage areas large and increases the



(a) Boundary with high T_c. (b) Boundary with low T_c. (c) Boundary with adjusted T_c.

Fig. 2: Boundaries depending on T_C.

chances to apply cases, but it increases the possibility of inadequate case application.

The AKB involves a threshold adjustment method for the hybridization of rules and cases. First, in the proposed knowledge base, each exceptional case has a threshold for deciding whether the case should be applied . This enables control of influence sphere at every exceptional case. Second, each threshold is adjusted automatically at a learning stage, while the existing method requires a developer to adjust the unified threshold manually. The AKB, therefore, reuses cases which are not reused efficiently in the existing method and prevents reusing cases which are reused incorrectly in the existing method.

The thresholds are adjusted one by one by the following procedure:

Step 1: Initialize a threshold T_{C_x} of an exceptional case C_x by $1.0 - \delta_T$.

Step 2: If $T_{C_x} \leq T_{C_x}^{(L)}$, terminate.

Step 3: For all learning data P_L in the root rule of C_x:

- **Step 3a:** Apply C_x if $Sim(P, C_x) > T_{C_x}$.
- **Step 3b:** If no error occurs, $T_{C_x} \leftarrow T_{C_x} - \Delta_T$ and go to Step 2.
- **Step 3c:** If errors occur, $T_{C_x} \leftarrow T_{C_x} + \Delta_T$ and terminate.

3 Application to Japanese-to-Braille translation

3.1 Overview

A computer extends information resources for visually disabled people who have difficulty getting information[9]. As use of the Internet has become widespread, the need for computer skills has increased. Technical manuals take three to four months to translate into Braille, even by experts. Despite the existence of advisory resources on Japanese-to-Braille translation such as ‘Ten-yaku no Tebiki’, (“A Guide to Braille Translation”) translators must often clarify ambiguous expressions caused both by ambiguous, complicated translation rules and the peculiarities of the Japanese language. Texts must be made consistent throughout, even if a translation can be interpreted in more than one way. In the final phase, translations must be completely checked manually to eliminate any remaining errors or doubtful meanings – a process that further compromises translation productivity.

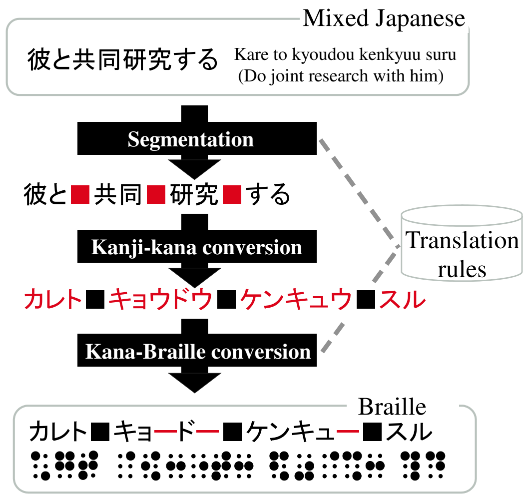


Fig. 3: Japanese-to-Braille translation process.

We define both kana sentences obeying the translation rules and sentences written in Braille characters as Braille sentences; they can be converted each other without ambiguity. We also define Japanese sentence that uses several types of characters – kanji, hiragana, katakana, letters, numerals, and symbols – as mixed Japanese. The task for translating Japanese into Braille is thus defined as translation of mixed Japanese into Braille, and is done in three steps – sentence segmentation, kanji-to-kana conversion and kana-Braille conversion. A Japanese sentence is a string of characters concatenated without blanks, so spaces must be inserted between words to get a proper interpretation. Kanji must be converted to kana, because Braille characters expressing Japanese correspond only to kana.

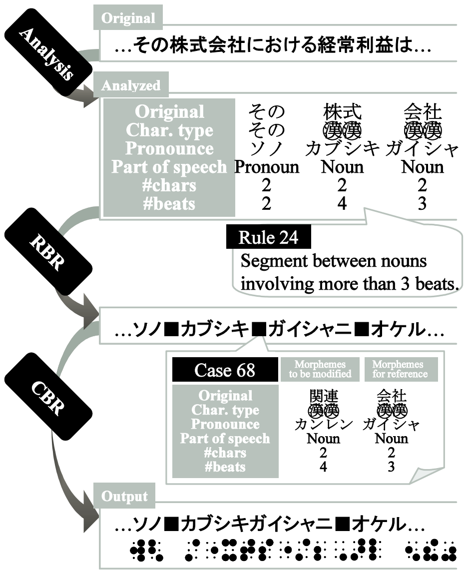


Fig. 4: Translation process of the proposed method.

The many translation rules that must be obeyed, are themselves ambiguous and full of exceptions [5, 10, 11]. In sentence segmentation, for example, rules require that semantic and phonetic information be considered. The word 'コーシュー ヨクジョー (公衆浴場)' (a public bath, pronounced 'kōshū-yokujō') must be segmented as 'コーシュー_ヨクジョー (公衆_浴場)' to be easily understood ('ヨクジョー (浴場)' means a bath), but the word 'カイスイヨクジョー (海水浴場)' (a beach, pronounced 'kaisui-yokujō') must not be segmented as exceptions, because the segmented word 'カイスイ_ヨクジョー (海水_浴場)' has different meaning, a bath using seawater. In kanji-to-kana conversion, rules for distinguishing ordinary vowels are unclear, i.e., 'ア', 'イ', 'ウ', 'エ', and 'オ', and the symbol 'ー' , denoting a long vowel, are used in writing, whereas a long vowel in regular Japanese is only used to express words of foreign origin or imitation sounds. Basically in Braille, a long vowel 'ō' is written as 'ー', so the word 'ガッコー (学校)'(a school, pronounced 'gakkō') uses 'ー'. But the word 'コオリ (氷)' (an ice, pronounced 'kōri') uses 'オ', not the symbol, to express the long vowel as exceptions. It is thus very difficult to represent rules so that they are followed automatically by a computer.

We have, therefore, proposed a system named J2B which utilizes both rules and cases[4, 5]. J2B enables to get exceptional knowledge as cases from users' revisions on the interactive interface of J2B. Although J2B performs the translation with high accuracy and efficiency, developers must adjust the unified threshold for case application. AJ2B, a system we proposed in this paper, enables to get exceptional knowledge as cases from Braille documents and automatically adjust the thresholds of the cases individually.

3.2 Rules and Cases in Japanese-to-Braille Translation

Cases and rules segment sentences and revise strings in a draft. A rule consists of an operator, conditions for applying the operator, and a priority score to resolve rule conflicts. An operator is classified into two groups: segmentation, which inserts or deletes a space between kana characters, and pronunciation, which changes kana characters themselves. A condition is stated by checking the value of an attribute. Attributes are obtained by morphological analysis, e.g., parts of speech, character type, a mora, and pronunciation which is a reading written in kana . All of those attributes are defined as symbolic attributes.

A case consists of an operator, a set of attributes of morphemes to which the operator of this case makes revision, and an identification number of a root rule, i.e., the rule for which the case is an exception. We define an object to which an operator of a rule or a case makes revision, i.e., a string or an interval between characters, as a spot.

In Japanese-to-Braille translation, because a case involves more than one morphemes, the number of attributes the case involves is six multiplied by the number of morphemes, instead of six attributes, as shown in Fig. 4 Moreover, some attributes involve large number of attribute values because each morpheme is composed of more than one characters. Total numbers of hiragana and katakana characters are almost 50 respectively. Total number of common kanji is about 2,000 and total number of kanji is about 50,000 characters. Such large quantities of attribute values make it difficult to apply SVM and some existing machine learning methods to Japanese-to-Braille translation.

3.3 Translation Procedure

The proposed system translates using the following steps.

Step 1: Analyze the source document written in mixed Japanese and make a first draft of Braille sentences from pronunciations given by the analysis.

Step 2: For each spot in a source document:

- **Step 2a:** Find the most appropriate rule R for the spot.
- **Step 2b:** Search cases whose root rule corresponds to R and calculate the similarity between the spot and each of the applicable cases.
- **Step 2c:** If similarity between the nearest case C and the spot is higher than the threshold,

then apply C. Otherwise, apply R.

The similarity between a spot in a source document and a candidate case is calculated by dividing the number of corresponding attributes by the number of all attributes.

3.4 Learning Procedure

The AKB requires both positive and negative cases for learning. In Japanese-to-Braille translation, existing documents in both Braille and source are used to generate cases. Cases are categorized into pronunciation and segmentation cases according to their operators.

At first, only rules are applied to translate the source document, errors are extracted as negative cases, and correctly translated parts are divided into positive cases. Then threshold adjustment is performed by following the procedure in Section 2.3. The lower limit $T^{(L)}_{C_S}$ of segmentation cases C_S is set to 0.7, and the limit $T^{(L)}_{C_P}$ of pronunciation cases C_P is set to 0.9. Because pronunciation cases specializes pronunciations to follow Japanese Braille rules, and revises errors of morpheme analysis, applying pronunciation cases should be performed like exact match. Segmentation cases are applied more flexible than applying pronunciation cases.

After threshold adjustment, the rules and the negative cases are stored into knowledge base. The positive cases are discarded or stored into an archive which is not used for problem solving.

4 Experimental results

We compared aJ2B and the most popular Japanese-to-Braille translation software on the market — Extra Ver.4.0. Documents “The Copyright Act” and “Computer encyclopedia” are used for evaluation. In each document, all sentences are divided into two groups — even sentences for training and odd ones for evaluation. The opposite experiments in which odd sentences are used for training and even sentences for evaluation are also performed.

Extra uses a user-defined dictionary in which manually-extracted words that cause errors for learning data are stored. J2B[4, 5] is also tested for evaluating the effectiveness of threshold adjustment and case acquisition.

J2B and aJ2B utilize a Japanese morphological analyzer called ChaSen with modified dictionary which does not involve extra foreign compound words. aJ2B parameters are decided as follows: $T^{(L)}_{C_P} = 0.9$, $T^{(L)}_{C_S} = 0.7$, $\Delta_T = 0.05$.

Fig. 5 shows the number of errors and translation time of the proposed method and Extra in each experiment. Errors are categorized into pronunciation and segmentation errors.

As shown Fig. 5(a) and (b), both using the user-defined dictionary in Extra and case acquisition in J2B are significantly effective in decreasing errors. This is because “The Copyright Act” is a document that defines laws and that involves a lot of the same and quite similar wording. The important point to note is that the case acquisition in J2B can eliminate almost half of segmentation errors, while the user-defined dictionary in Extra eliminates only 15% to 35% of them. This indicates that the flexibility of cases is effective in representing exceptional segmentation knowledge of Japanese-to-Braille translation. In addition, the threshold adjustment can also reduce segmentation errors to almost 80% of J2B with cases. This indicates that the threshold adjustment can expand the flexibility of each case. In contrast, the threshold adjustment has no effects on pronunciation errors because cases for revising the pronunciation errors should be applied only to the same or quite similar wordings.

As shown Fig. 5(c) and (d), compared with the results in “The Copyright Act”, the effectiveness of both the user-defined dictionary in Extra and the case acquisition in J2B is diminished. This is because “Computer encyclopedia” does not involve not so many similar wordings as “The Copyright Act”.

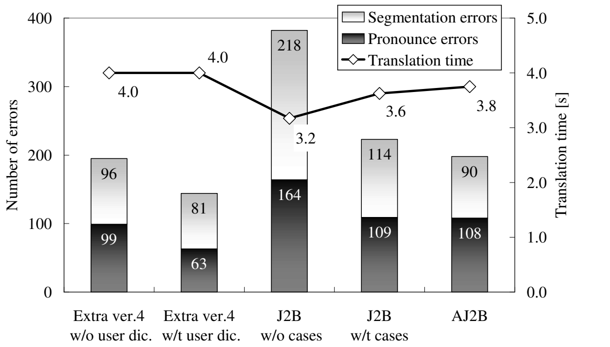
Although the user-defined dictionary eliminates 11% segmentation errors and 40% pronounce errors in the case that even sentences are used for learning and odd ones for evaluation, the dictionary could not eliminates errors in the opposite experiment. In contrast, the case acquisition in J2B eliminates 15% segmentation errors and 22% pronunciation errors in the first experiment and 16% segmentation errors and 33% pronunciation errors even in the opposite experiment. Moreover, the threshold adjustment succeeded to reduce segmentation errors to 80% and 71% even in those experiments.

5 Conclusions

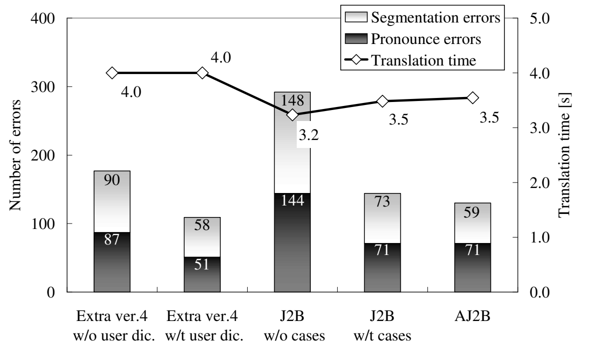
In this paper, we proposed an adaptive knowledge base composed of rules and cases. The proposed knowledge base acquires exceptional cases from existing Braille documents and adjusts their threshold for judging whether the case should be applied or not. Experiments in Japanese-to-Braille translation problem showed that acquiring cases and adjusting their thresholds reduce errors.

We plan to expand the adaptive knowledge base for other real-world problems.

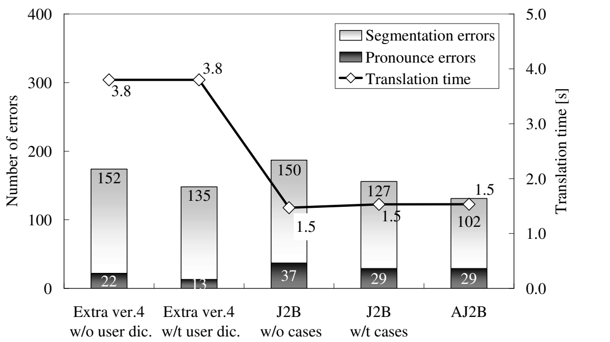
Acknowledgements: This research was partially



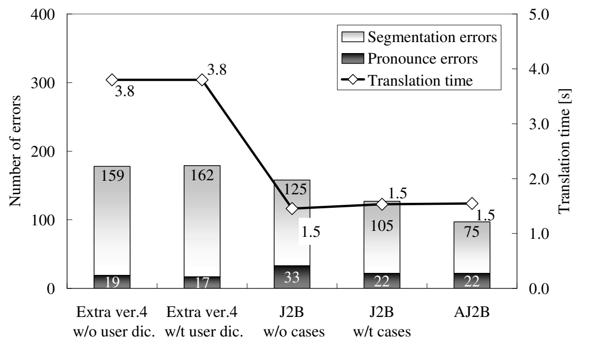
(a) The Copyright Act: Even sentences are used for learning and odd sentences are used for test.



(b) The Copyright Act: Odd sentences are used for learning and even sentences are used for test.



(c) Computer encyclopedia: Even sentences are used for learning and odd sentences are used for test.



(d) Computer encyclopedia: Odd sentences are used for learning and even sentences are used for test.

Fig. 5: Experimental results

supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Young Scientist (B), No.16700412, 2004–2006.

References:

[1] R. Schank, A. Kassand, and C. Riesbeck, *Inside Case-Based Explanation*. Hillsdale, New Jersey: Lawrence Erlbaum, 1994.

[2] B. Bartsch-Spörl, M. Lenz, and A. Hubner, “Case-based reasoning - survey and future directions,” *Proc. XPS-99, Springer Verlag, LNAI*, 1999.

[3] A. R. Golding and P. S. Rosenbloom, “Improving accuracy by combining rule-based and case-based reasoning,” *Artificial Intelligence*, vol. 87, 1996, pp.215–254.

[4] S. Ono, Y. Hamada, Y. Takagi, S. Nishihara, and K. Mizuno, “Interactive japanese-to-braille translation using case-based knowledge on the web,” in *Proc. of the Sixth Pacific Rim Int’l Conf. on Artificial Intelligence*, 2000, pp. 638–646.

[5] S. Ono, Y. Takagi, Y. Hamada, K. Mizuno, and S. Nishihara, “Japanese-to-braille trnslation and error correction support using case-based knowledge,” *Human Interface*, vol. 5, no. 4, 2003, pp.491–498.

[6] F. Hayes-Roth, D. Waterman, and D. Lenat, *Building Expert Systems*. Addison-Wesley Publishing Co., Reading, 1983.

[7] S. Ono, Y. Hamada, K. Mizuno, Y. Fukui, and S. Nishihara, “A rearrangement system for floor layouts based on case-based reasoning and constraint satisfaction,” in *Proc. of 21st IASTED Int’l Conf. on Applied Informatics (AI2003)*, 2003, pp.363–368.

[8] C. J. C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining and Knowledge Discovery*, vol. 2, 1998, pp.121–167.

[9] A. Koide, C. Asakawa, and N. Suzuki, “Research on computer aids for the visually disabled,” *World Scientific*, 1994, pp.66–69.

[10] M. Kawaahara, “Yet another japanese braille translation program,” *Technical report of IEICE*, vol. HC94-49, 1994, pp. 51–58.

[11] E. Suzuki, S. Ono, T. Hiraoka, and H. Kanoh, “Interactive japanese sentence segmentation system for translating japanese into braille,” in *Proc. of the Natural Language Processing Pacific Rim Symposium*, vol. 1, 1997, pp.621–624.