Available online at www.sciencedirect.com

ScienceDirect

Speech Communication 51 (2009) 995–1005

SPEECH COMMUNICATION

www.elsevier.com/locate/specom

Computer Assisted Language Learning system based on dynamic question generation and error prediction for automatic speech recognition

Hongcui Wang *, Christopher J. Waple, Tatsuya Kawahara

School of Informatics, Kyoto University, Sakyo-ku, Kyoto 606-8501, Japan

Received 30 June 2008; received in revised form 13 March 2009; accepted 16 March 2009

Abstract

We have developed a new Computer Assisted Language Learning (CALL) system to aid students learning Japanese as a second language. The system offers students the chance to practice elementary Japanese by creating their own sentences based on visual prompts, before receiving feedback on their mistakes. It is designed to detect lexical and grammatical errors in the input sentence as well as pronunciation errors in the speech input. Questions are dynamically generated along with sentence patterns of the lesson point, to realize variety and flexibility of the lesson. Students can give their answers with either text input or speech input. To enhance speech recognition performance, a decision tree-based method is incorporated to predict possible errors made by non-native speakers for each generated sentence on the fly. Trials of the system were conducted by foreign university students, and positive feedback was reported.

Keywords: Computer Assisted Language Learning (CALL); Second language learning; Automatic speech recognition; Error prediction

1. Introduction

Computer Assisted Language Learning (CALL) systems can offer many potential benefits for both learners and teachers, because they offer learners the chance to practice extra learning material at their convenient time in a stress-free environment (Witt, 1999; Zinovjeva, 2005). And there is a significant interest in the development of CALL systems recently. Many research efforts have been made for improvement of such systems especially in the field of second language learning (Tsubota et al., 2004; Abdou et al., 2006).

There are a number of CALL systems that have already been developed covering almost every aspects of language learning. Some systems concentrate on vocabulary and grammar learning. Some focus on pronunciation learning. And also some allow training of an entire situation-based conversation. However, most of systems tend to be limited either by the repetitiveness of the learning material, or by the lack of freedom offered to the learners, because they mostly allow reading the defined words or sentences. Considering these, we have designed and developed a new CALL system named CALLJ to aid students learning the elementary Japanese grammar, vocabulary and pronunciation via a set of dynamically generated sentence production exercises.

In CALLJ, a sentence concept (situational context within which the sentence is to be formed) is presented with a concept diagram, a picture that graphically depicts the situation to be described, along with the appropriate grammar rules. Students can input answers by speech or keyboard. Errors made by the students are detected and appropriate feedback will be provided. The system also features an interactive hint system through which the students may choose to receive guidance to complete each task. Additionally, a scoring system has been included in the system to penalize students for making mistakes, or for using

[*] Corresponding author.
E-mail address: wang@ar.media.kyoto-u.ac.jp (H. Wang).

0167-6393/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.specom.2009.03.006

the hint system, and also to motivate them to improve further.

The system generates each question dynamically, thus reducing the repetitiveness. For the speech input, since the system has an idea of the desired target sentences, it is natural to generate a dedicated grammar network as a language model for automatic speech recognition (ASR). To be an effective CALL system, the grammar network should cover errors that non-native learners tend to make. On the other hand, considering all possible errors would significantly increase the perplexity of the network, thus degrade the ASR performance. A decision tree-based error classification algorithm is proposed for effective error prediction, which means predicting critical error patterns without a large increase in perplexity. In order to evaluate the usefulness of the system, we have conducted a set of trials in which students practice a number of lessons.

The remainder of this paper is organized as follows. Several Japanese CALL systems are reviewed in Section 2, and the system design of our CALLJ is presented in Section 3. In Sections 4–6, the system's main features of dynamic question generation, grammar generation for ASR, and scoring mechanisms are explained, respectively. Then, Section 7 presents the evaluation results, findings and feedback from students. Section 8 concludes with a summary.

2. Review of CALL systems

There have been a number of studies on CALL systems, addressing various areas of language learning. Several systems related with this work, mainly focusing on Japanese, are reviewed.

The BANZAI system (Nagata, 2002) was developed to improve grammatical ability, and has been successfully used in real lessons. The system provides intelligent feedback by analyzing the grammar components of input sentences, but does not offer any help or hints to the students so that they make a correct answer by themselves. It does not give an indication score for overall proficiency through the lesson, either. And since the set of questions are fixed, the students are always asked to create the same sentences each time they run the system. In this work, we investigate automatic generation of questions together with hints and scores, to fully exploit potential advantages of the CALL compared with conventional textbooks. Moreover, we also deal with speech input and conversational-style sentences while the BANZAI system was limited to written Japanese.

With incorporation of ASR, CALL systems have been used for pronunciation learning, specifically evaluating of pronunciation in speech inputs and correcting errors, such as the system in (Kawai and Hirose, 2000), FLUENCY (Eskenazi and Hansma, 1998), WebGrader (Neumeyer et al., 1998), and EduSpeakTM (Franco et al., 2000). The Japanese pronunciation learning system (Kawai and Hirose, 2000) focused on *tokushuhaku* double mora, which is most difficult for non-native speakers of Japanese. Students are asked to read out minimal pairs of words with/without *tokushuhaku* mora, and then the system evaluates intelligibility and outputs a score. In this work, we cover more general Japanese sentences in a certain context.

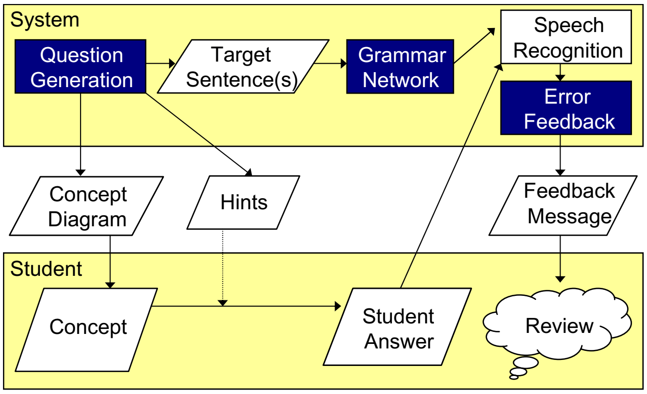


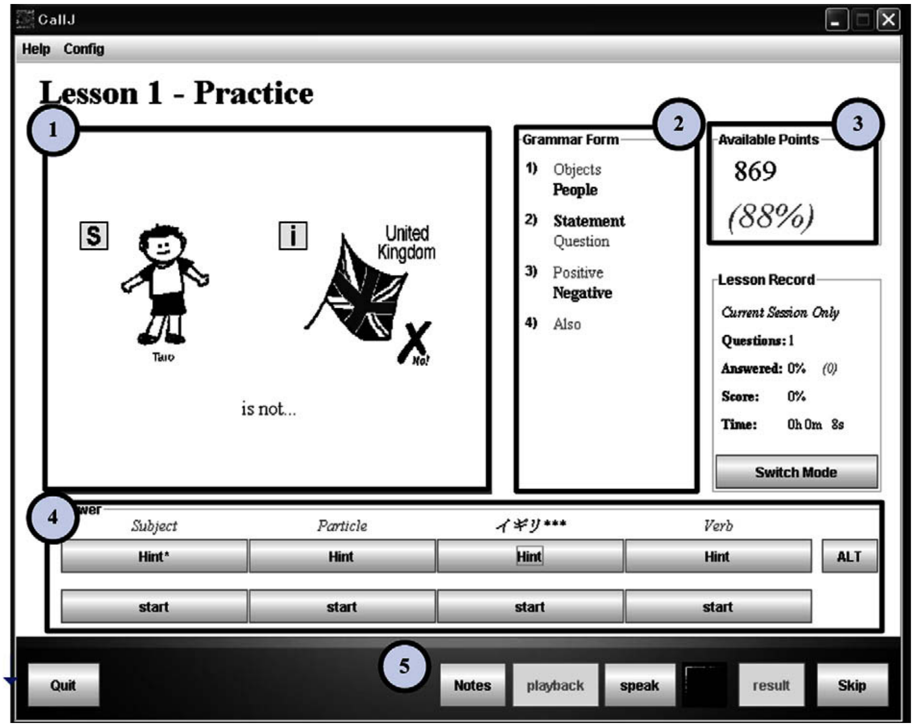
Fig. 1. System overview.

The Subarashii system (Bernstein et al., 1999) is a conversational system that offers beginners of Japanese the opportunity to solve simple problems through spoken interactions. In a series of everyday situations, the system poses problems in written English (e.g., inviting a friend to go to a movie) and offers occasional support in the form of written reminders, but problems can only be solved by speaking an appropriate Japanese sentence. Since the set of situations are fixed, a dedicated grammar is prepared beforehand to recognize speech inputs for each situation. In the CALLJ presented in this paper, we implement a mechanism of dynamic question generation and error prediction, although we focus on the elementary Japanese sentences and do not offer a conversational environment.

3. System overview of CALLJ

The system is organized in lessons, covering elementary grammar points and vocabulary from levels 4 and 3 of the Japanese Language Proficiency Test (JLPT[1]). These levels cover approximately 1500 words (of which around 200 are verbs), 300 kanji characters, and 95 grammar points. The grammar points are distributed across a set of 30 lessons. Each lesson consists of exercises and self-learning material, which help students master key grammar points and key sentence patterns. The exercises are a collection of related questions (sentences) connected to some key sentence patterns (grammar points), such as "like to do something". Before practicing, students look through the overview of the lesson points, notes of the grammar points, and examples of questions. Specifically, the overview briefly shows key sentence patterns and grammar forms. The notes give more information on the grammar structures that are used in the lesson. With these documents,

[1] <http://en.wikipedia.org/wiki/JLPT>.



Lesson 1 - Practice

is not...

Grammar Form
1) Objects
People
2) Statement
Question
3) Positive
Negative
4) Also

Available Points
869
(88%)

Lesson Record
Current Session Only
Questions: 1
Answered: 0% (0)
Score: 0%
Time: 0h 0m 8s

wer
Subject | Particle | イギリ*** | Verb

Fig. 2. Question practice screen: (1) concept diagram; (2) desired form guide; (3) score; (4) answer area and hint display; and (5) control button panel.

students get an idea on sentence patterns in the current lesson before they exercise using the system.

A process flow of the exercises is depicted in Fig. 1. Each question involves the students being shown a "concept diagram", which is a picture representing a certain situation. The students are then asked to describe this situation with an appropriate Japanese sentence using text input or speech input. Thus, the system allows students the freedom to create their own sentences. If the answer is given via a microphone, ASR is conducted using a language model in the form of a grammar network for the target sentence. Errors will be detected and feedback information is generated for the students. This process of question, answer and feedback is repeated.

Unlike the conventional textbooks or prepared materials, the system generates questions on the fly, by selecting subjects, objects and optional phrases with regard to time and place and so on. Accordingly, the diagram and the grammar network is generated by dynamically combining the relevant parts. Thus, students can try as many questions as they want. As every question is generated randomly, there is no relation in a sequence of the questions; later we revise the system so that it generates a question focusing on the observed errors of the current student.

Fig. 2 shows the user practice interface. In the following sections, we describe further details regarding the main modules of the system, namely question generation, ASR grammar network generation, error feedback and the scoring system.

4. Dynamic question generation

Fig. 3 shows an overview of question generation. In order to reduce the repetitiveness of the questions offered by the system, we dynamically generate each question at run time from the set of vocabulary and grammar rules available. This involves the creation of four main components: a concept or situation that the students must describe, a diagram that represents this situation, target sentence instances that the students are expected to produce, and hints for the target sentences.

4.1. Concept definition

The first task in generating a question is to generate the situation to be described. Each lesson consists of one or two question types corresponding to grammar points, and each question type uses several concept templates.

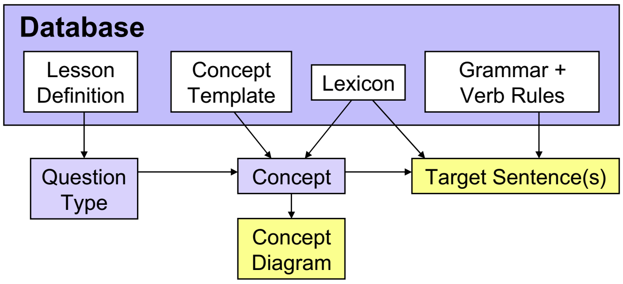


Fig. 3. Question generation.

```
<question people 0.5>
   <concept person_is_1 0.5>
      <text [person] is...>
      <grammar DESU_SIMPLE>
      <diagram diagram2>
   </concept>
   .......
</question>
```

Fig. 4. Example of question type.

```
<frame person_is_1>
   <slot Subject 1.0>
      [name] 1.0
   </slot>
   <slot Description 1.0>
      [job] 0.5
      [nationality] 0.5
   </slot>
   <slot Verb 1.0>
      (is) 1.0
   </slot>
</frame>
```

Fig. 5. Example of concept template.

Examples for a question type and a concept template are shown in Figs. 4 and 5.

A single template covers a range of related situations, and defines the semantic components or slots that are required, optional or to be omitted when defining a specific situation. Once the template is selected, the system then selects which information slots are to be activated (the optional slots are decided randomly). For the active slots, the system selects an appropriate value. This value is selected depending on the nature of the slot specification. The filler may be either a word selected from the lexicon, or another concept template. Selections of concepts and individual words are done randomly under the above-mentioned constraints and preference weights attached to the optional slots.

4.2. Concept diagram

The concept diagram is a system-generated diagram which depicts the situation or concept that the students have to describe. Displaying such information graphically helps avoid the problem of expressing the situation via a specific language, which could be problematic in cases where the native language of the students vary widely.

Also, it has been hypothesized by Nelson et al. (1976) that pictures are easier for the students to process and recall (a phenomena known as the Picture Superiority Effect), since they enable the students to comprehend the semantic meaning behind the situation quicker than with text (Smith and Magee, 1980). This in turn may lead to more satisfying and effective learning (Levie and Lentz, 1982).

Having the system generate the diagram offers a number of advantages. Firstly, it significantly reduces the cost timewise in creating the images. Secondly, it leads to a greater consistency in style across the images. The diagram is created by combining a number of smaller sub-images, each representing a component in the concept instance. A diagram template is defined for each concept template to specify the set of sub-images that should be used, along with their coordinates and size. A text label in English is attached to many objects to reduce the ambiguity.

4.3. Sentence generation

The sentences are created in a network form, as shown in the lower half of Fig. 6. The network is created by taking the information in the concept instance (the completed case frame), and applying a set of grammar rules. The grammar rules define a hierarchical structure based on a set of top level sentence templates, with each component in the template being defined by a further rule.

Consider the example given in Fig. 6. The top-level grammar rule template specifies that the sentence should consist of three components: subject, description and verb. These three components are each parsed in turn. The Subject component, for example, is comprised of two sub-components: a sub-rule that expands into the subject itself (appending a suffix to the name if appropriate), and the associated particle. The rest of the sentence network is created in a similar fashion, from the top level template, through all the sub-rules and their associated templates, adding the relevant words to the network in a recursive manner. Whilst not shown in this example, the grammar rules often contain many restrictions and conditional clauses to deal with particular exceptional cases.

4.4. Hint system

To help students and constrain their answers, the system shows segmented slots, corresponding individual words, labeled with their class (subject, particle, verb, etc.), as shown in Fig. 2.

Moreover, we prepare a hint system which allows the students to reveal each word in the target sentence in stages, thus allowing them to receive just the amount of help they need to complete the task. Each word is not simply revealed in one step, but incrementally with the word class being given first, then the word length, and then character by character till the whole word is revealed.

The hints are generated by breaking down the target sentence (one sentence arbitrarily selected from the sentence network) into its constituent components, and then for each component creating an ordered set of hints. Note that whilst the hints are based on just one of the target sentences, the remaining target sentences are also valid answers to the question, and thus the students' answer does not necessarily have to match that given by the hint system to be classified as a correct answer.[2]

[2] Although the hints are generated independently of the students' answer, the students can change the answer based on the hints.

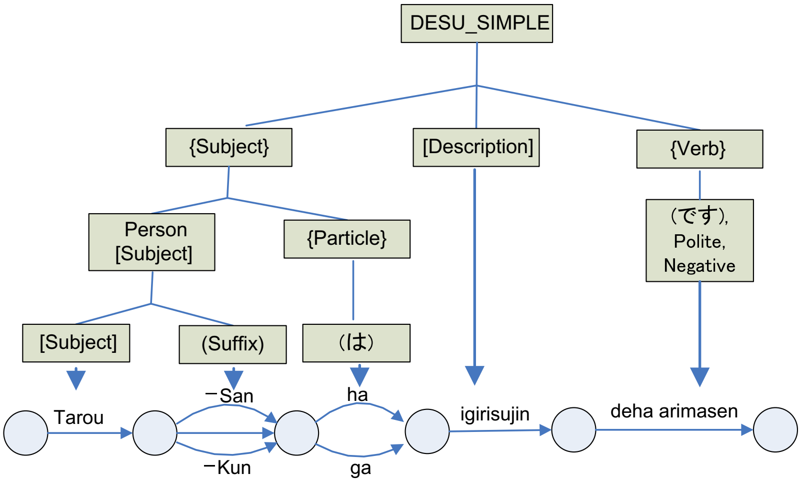


Fig. 6. Grammar-based sentence generation.

Fig. 7 shows an example of a sentence being broken down into a set of hints. In this diagram, all of the sentence components have a word class label and several level hints. The hint levels include length hints, base-form hints and surface form hints. The base-form hint is actually divided into a number of intermediate hints, revealing the target word character by character. If the students actually know a word but have forgotten it, initially giving them small fractions of the words may be enough to help them remember the word, and would thus be more useful than just giving them the whole word straight away. If the students use all the hints on a particular word, they will score no points for that sentence component. Deciding the cost for each hint will be discussed in relation to the scoring system in Section 6.

5. ASR grammar network generation with error prediction

As the system has an idea of the desired target sentences, the system easily generates a language model to cover them in the form of a network. The major problem is to predict errors (possible answers different from target sentences) that non-native students tend to make, and to integrate them into the language model.

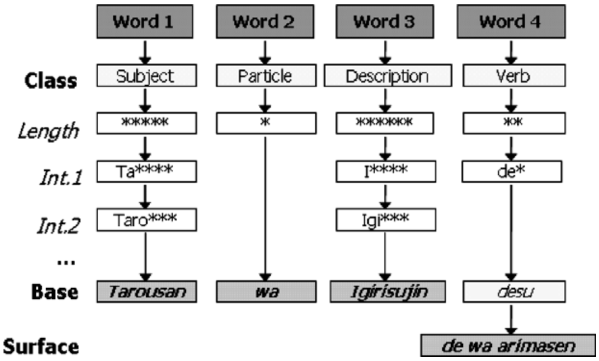


Fig. 7. Component-based “hint chains”.

In the conventional studies mentioned in Section 2 which use ASR for the production practice of words or sentences, the linguistic knowledge is widely used to achieve better error prediction. In (Tsubota et al., 2002), 79 kinds of pronunciation error patterns according to linguistic literatures were modeled and incorporated to recognize Japanese students' English. However, the learner of the system was limited to Japanese students. Obviously, a larger number of error patterns will exist if the system allows any non-native speakers. Moreover, we need to handle more variations in the input, if we allow more freedom in the sentence generation, like CALLJ. These factors, when counted together, would drastically increase the perplexity of the grammar network, causing adverse effects on ASR. In order to find critical errors and avoid redundant errors, a decision tree is introduced for error classification (Wang and Kawahara, 2008).

5.1. Error classification

The error classification is conducted by comparing the features of the observed word to those of the target word. The features include same POS (part-of-speech; verb, noun, etc.), same base form, similar concept, wrong inflection form, and so on. To select effective features and find critical error patterns, an “impact” criterion is introduced to find an optimal decision tree that balances the tradeoff of the error coverage and perplexity. It is used to expand a certain tree node from the root node (containing everything), and partition the data contained in the node according to some feature. For a given error pattern, it is defined as below:

$$impact = \frac{\text{error coverage}}{\text{perplexity}}. \qquad (1)$$

Error coverage is defined as the proportion of errors being predicted among all errors. It is measured by the frequency in the training data set, so that more frequent errors are given a higher priority. Perplexity is defined as an exponential of the average logarithm of the number of possible

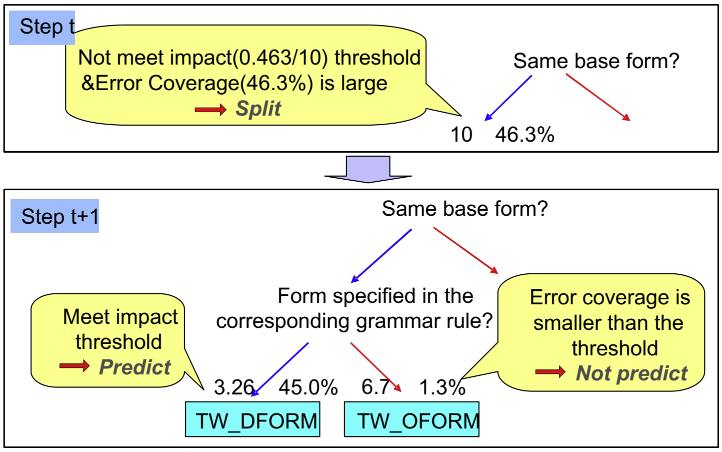

Step t
Not meet impact(0.463/10) threshold &Error Coverage(46.3%) is large
Split
Same base form?
10 46.3%
Step t+1
Same base form?
Form specified in the corresponding grammar rule?
Meet impact threshold
Predict
Error coverage is smaller than the threshold
Not predict
3.26 45.0% 6.7 1.3%
TW_DFORM
TW_OFORM

Fig. 8. Example of decision tree training process.

Table 1
Error patterns being predicted for verbs.

Pattern	Type	Description
TW_DForm	Grammatical	Target word (base form) in different form
DW_SForm	Lexical	Different word in same form
DW_DForm	Lexical	Different word in different form
TW_WIF	Grammatical	Target word in wrong inflection form

competing candidates at every word in consideration. In this work, for efficiency and convenience, we approximate it by the average number of predicted competing candidates for every word in the training data set. The larger value of this impact, the better recognition performance can be achieved with this error prediction. Our goal is reduced to finding a set of error patterns that have large impacts. If a current node in the tree does not meet this criteria (threshold), we expand the node and partition the data iteratively until we find the effective subsets and mark "to predict", or the subset's coverage becomes too small and marked "not to predict". Fig. 8 shows an example of one step of the tree training for verbs. In each node, perplexity and error coverage of the node is labeled from left to right.

The training data for the decision tree learning were collected through the trials of the prototype CALLJ system with text input. They consist of 880 sentences, containing 653 errors. Since some errors can never happen or be tolerant in the speech input, we performed a pre-processing. Specifically, we corrected the input errors which are caused by typing or spelling mistakes and result in same pronunciation, such as "*o*" for "*wo*" (a particle) and "*tanaka san*" for "*tanakasan*".

After the training process, a decision tree is derived for each POS. As for verbs, 11 leaves are extended with a maximum depth of six in a binary tree. Among them, four leaf nodes are chosen for prediction as listed in Table 1.

Each error pattern falls within one of four error types: *Lexical*, *Grammatical*, *Concept*, and *Input*. Lexical errors are out-of-vocabulary words and the inappropriate choice of words which are similar in concept. Features to identify similar-concept word pairs depend on the word component type. For verbs, they are: the substitution between words that are grammar points (such as "*ageru*", "*kureru*", and "*morau*"), between words having same meaning (such as "*honnyakusuru*" and "*yakusu*"), between the transitive and intransitive verb pair (such as "*okosu*" and "*okiru*"). Grammatical errors include wrong forms or wrong inflections of the correct word and inappropriate particles. Concept errors are mistakes not in the language itself, but in the interpretation of the situation that the students need to describe. Input errors are mistakes in the input format, such as *hiragana* being used instead of *katakana*.

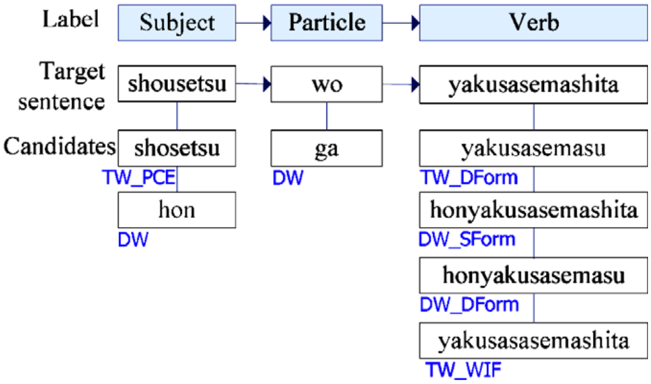


Fig. 9. Prediction result for given sentence.

5.2. Error prediction integrated to language model

As we identified the errors to predict, we can exploit this information to generate a finite-state grammar network. Given a target sentence, for each word in the surface form, we extract its features needed such as POS and the base form, and compare the features with error patterns to predict using the decision tree. Then, we generate potential error patterns with the prediction rules and add them to the grammar node. Fig. 9 shows an example of a recognition grammar based on the proposed method for a sentence "*shousetsu wo yakusasemashitaka*".

6. Feedback and scoring

6.1. Feedback to learners

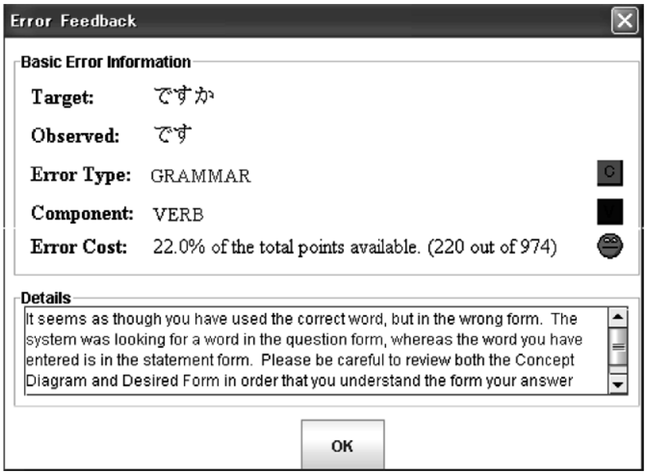
The CALL system should provide pertinent corrective feedback of errors made by students. The feedback is performed based on the error classification in Section 5.1. After the decision tree analysis, we get the features of error patterns non-native speakers tend to make. Based on the information, the system can identify the reason of the error and then tell how the learner can correct it. These are defined in the error help structure and displayed in a short text in the feedback dialogue interface. An example of the help dialogue for a "TW_DForm" error is shown in Fig. 10.

6.2. Scoring system

To provide students an idea of how they are progressing, we devise a scoring system. The system penalizes the students for making mistakes as well as for using hints to answer a question. We define the penalty score for an error e of a word w by the following equation:

$$ErrorPenalty(w,e) = W_c \times W_e, \tag{2}$$

where W_c is the weight associated with the component POS, and W_e represent the cost associated with each possible error pattern, as listed in Table 1.



Basic Error Information

Target: ですか

Observed: です

Error Type: GRAMMAR

Component: VERB

Error Cost: 22.0% of the total points available. (220 out of 974)

Details

It seems as though you have used the correct word, but in the wrong form. The system was looking for a word in the question form, whereas the word you have entered is in the statement form. Please be careful to review both the Concept Diagram and Desired Form in order that you understand the form your answer

Fig. 10. Feedback for "TW_DForm" errors.

Determining the values of the penalties is an important issue. It should be determined by considering how difficult for students an error pattern is. When students make correct answers for more difficult questions, they will be rewarded a larger score, and vice versa. This property is realized by considering the frequency of the errors. We assume that the more the error pattern is observed, the more difficult it is for learners. Thus, we determine the weights' values based on the observed frequencies multiplied by a normalized constant. The weight formula for an error pattern e is:

$$W_e = P(e) \times 20 + 1, \tag{3}$$

where $P(e)$ is the relative frequency of the error pattern normalized by the number of possible cases in the training data. And the weight is finally an integer, larger than or equal to one. Table 2 shows the weights estimated with the above equation for the predicted verb errors. The weights W_c are estimated in a similar way.

The total number of points available for each question is the sum of maximum penalties for each word in the target sentence, which means, if a student make large errors on every word they would score zero. The total score for a sentence is expressed by the following equation:

$$Score = \sum_w (\max_e(ErrorPenalty(w,e))). \tag{4}$$

The penalty for using all hints on a particular word is assigned by the maximum error penalty associated with that word, and is distributed across the different hint levels, the percentage of the cost for each level being determined by the hint-level weights. Hint levels include length, intermediate, base form, and surface form. The formula is as following:

$$HintPenalty(w) = \max_e(ErrorPenalty(w,e))\frac{W_l}{\sum_k W_k}, \tag{5}$$

where W_l is the weight associated with the level of the current hint, and it is also estimated based on the observed frequency in the trials.

7. Experiments and evaluation

The goal of the system is to help students improve the elementary skills in vocabulary, grammar and pronuncia-

Table 2
Penalty weights for verbs.

Pattern	Weight
TW_DForm	9
DW_SForm	4
DW_DForm	1
TW_WIF	3
Others	3

	Aligned sentence				
Target	*kare*	*ni*	*shousetsu*	*wo*	*yakusasemasu*
Transcript	*kare*	*ni*	*shousetsu*	*ha*	*yakusasemashita*
ASR result	*kare*	*ni*	*shosetsu*	*wo*	*yakusasemashita*
			false alarm	error (undetected)	error (detected)

Fig. 11. Example of error detection.

tion. However, it is not easy to evaluate the system from such a viewpoint in a short period. Thus, we first evaluate whether the function modules work properly, and analyze the statistics of the trials. We also had the subjects of the trials complete questionnaires on their experience with the system.

7.1. Experiment setup

Twenty one foreign students of Kyoto University took part in the first trials using the text-input prototype system. The data collected in this trial were used for the training of the decision tree and the penalty weights. In the second trial, ten foreign students tested the system which incorporates speech-input capability. The data collected in this trial were used for evaluation of ASR and the scoring system. Ten students are from seven different countries including China, France, Germany and Korea. All students were studying Japanese in the Kyoto University Japanese language course, and thus their approximate language proficiency was known based on the course level in which they were enrolled in (Elementary, Intermediate 1 or Intermediate 2).

All students had no experience with the CALL system before the trial, but were briefly introduced before undertaking the task. Each student ran through a set of lessons, answering a set of generated questions before seeing the correct answers and feedback for errors they made. In the second trial, ASR based on a grammar network was executed at run time. After the trials, all utterances (140) were transcribed including errors by a Japanese teacher. For the analysis of the two input modes, we use data collected from seven common lessons in the two trials.

7.2. ASR performance

For the speech input, the system should recognize the sentence answered by students and detect errors. Fig. 11 shows an example of a target sentence, its correct transcript, the recognizer's output. The three sequences of sentences are aligned word by word. The system correctly detects that the student made a mistake with the word "*yakusasemasu*", but incorrectly identifies the word "*wo*" and makes a false alarm in "*shousetsu*".

To evaluate the performance of ASR, we use the conventional WER (word error rate), error detection rate and false alarm rate. We define the error detection rate as the number of detected errors divided by the total number of errors the students made. The false alarm rate is the number of words erroneously flagged as a student error, divided by the total number of words students spoke correctly. For example, in Fig. 11, the number of errors is 2. One of them is detected. Thus the error detection rate is 50%. The number of false alarm is 1 out of three correctly spoken words, thus the false alarm rate is 33%.

Comparing the system's output to the faithful transcript of utterances including errors made by the students, the WER of ASR is 11.2%. It is quite lower compared with the case (28.5%) using the baseline grammar, which is hand-crafted and does not consider errors made by foreign students. The baseline method simply includes all words in the same concept such as foods and drinks in the grammar network, and can be applied to any sentences in the same lesson. The error detection rate is 75.7% with the false alarm rate of 8.6%, though 85.7% of errors were covered by the grammar network and could be recognized in theory. The error coverage (85.7%) and perplexity measure (4.1) for the test data are comparable to those (77.9% and 5.1) for the training data. The result confirms the generality of the decision tree training.

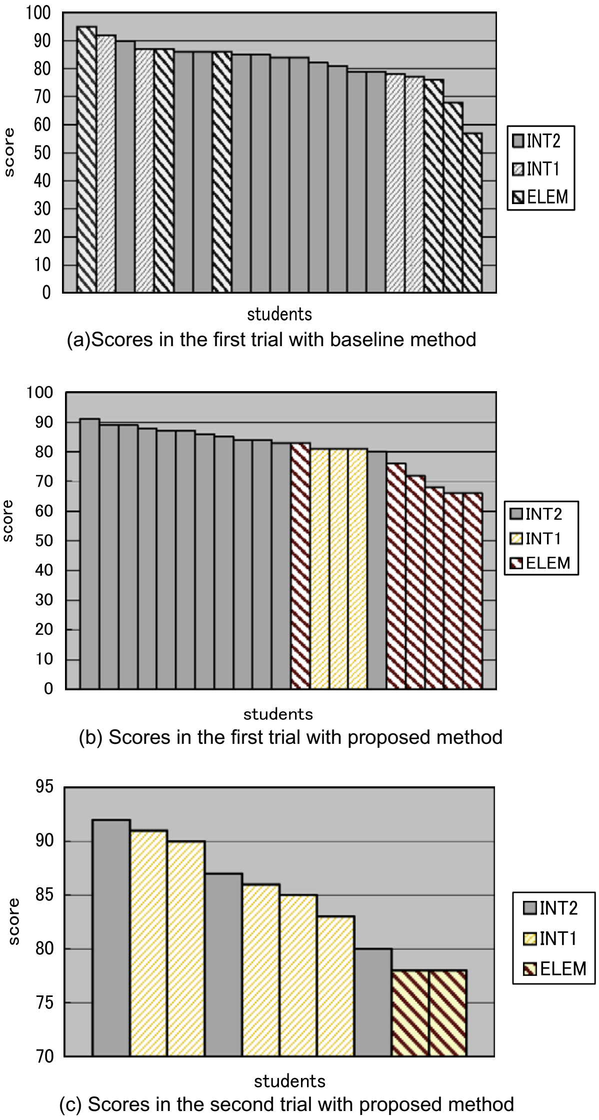
7.3. Evaluation of scoring system

In order to evaluate the scoring system, we investigate whether the score given by the system is correlated with the students' overall language proficiency. Each student was labeled as being of either "Elementary", "Intermediate 1" or "Intermediate 2" level based on the class they were enrolled in. Fig. 12 shows the scores obtained with the estimated weights for each student in the two trials separately. The scores are ordered from the highest on the left-hand side to the lowest on the right. With the proposed method, the elementary students are generally clustered to the right-hand side, with the lowest scores. We also tested a naive baseline method which penalizes equally for any error types. By comparing Fig. 12a and b, it is apparent the naive method apparently does not work as well as the proposed method.

Although the number of the subjects is not large, the overall results suggest that the system offers a meaningful measure of proficiency to the students, and that the estimated weights used for the scoring system are appropriate.

7.4. Error analysis for system improvement

Fig. 13 shows the distribution of different types of errors detected during the two trials. The error rate is calculated by dividing the total occurrence of each error type by the number of components observed on which that error type may occur. It is observed that there is not a large difference between two input modes, except that input errors never happen in the speech-input mode since all utterances are matched to some words in the vocabulary. We can also see the most frequent form of problems are lexical errors in both input modes. This result suggests that the lexical



(a)Scores in the first trial with baseline method

(b) Scores in the first trial with proposed method

(c) Scores in the second trial with proposed method

Fig. 12. Students' scores given by the system.

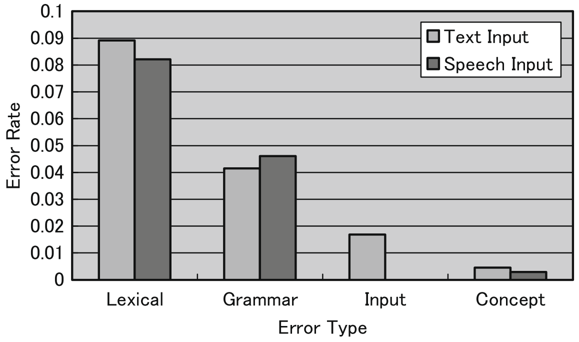


Fig. 13. Frequencies of observed error types.

issues were more significant than the grammatical issues, and that the students had more problems with vocabulary deficiencies.

For a better learning effect, the system is revised to select a relevant concept rather than generating it randomly. That is, the system keeps track of the words and forms (mainly for verbs) erroneously replied in the previous question, and tries to use them in the next question, until the learners correct them.

7.5. Hint usage

We also investigate the statistics of the hint usage by the students of the two proficiency groups, as shown in Fig. 14. The frequency is calculated by dividing the number of times each hint level was used by the number of components observed for which such a hint level was available. To view the base-form hint, for example, the students must have used the length and intermediate hints, but these were not counted; only the final level at which the students stopped using the hint is counted in order to see at which level the students reach the answer. This figure shows that the elementary students more often use the hints than the intermediate students, and that they turn to the base-form hints in most cases. The result also confirms the significance of the lexical problem. The result also validates the scoring system based on the hint usage.

7.6. Error correction by considering communication aspect

For further improvement, we investigate the false alarms and the errors of ASR that were covered by the grammar network but could not be detected, which amounts to about 10%. It is observed that the majority of such errors and false alarms by the system belong to "TW_PCE (Target Word Pronunciation Error)" type, which means the word is pronounced erroneously by adding or omitting a single double consonant, long vowel or voiced pronunciation, for example, "*kipu*" instead of "*kippu*". Actually, most of these errors do not cause difficulty for people to understand in a context of a whole sentence. Thus, we offer students an option to "weigh communication more than pronunciation details". If students choose this option, the system will automatically correct the above-mentioned errors (either by the students or by the system) before displaying the ASR results. This would improve the robustness of the system.

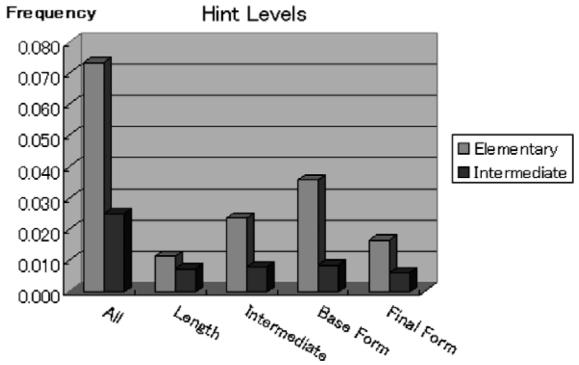


Fig. 14. Frequencies of hint-level usage.

Table 3
System assessment.

	Question	Statistics
Clarity	The key point of each lesson was clear	4.2
	I could clearly understand the concept diagram	3.3
	I found that the diagram became easier to understand over time	4.7
Hint system	I found the hints to be useful in solving the problem	4.0
	I would like to be able to configure costs	3.7
ASR	In general, did you feel you experienced a lot of problems with the speech recognition function in CALLJ	10%
Input mode	I prefer to speak my answer than to type it	3.8
	I prefer to type my answer as opposed to speaking it	3.0
	I would always like to be able to choose whether to speak or type	4.2
Overall	I would enjoy using such a system	90%
Satisfaction	I would like to have used such a system before coming to Japan	90%

7.7. System assessment by students

After the trials, the students were asked to evaluate the system with a questionnaire. Some questions and statistics were listed in Table 3. In the table, percentage is the ratio of students who selected each statement as appropriate and the score is from 1 (strongly disagree with statement) to 5 (strongly agree with statement).

It is confirmed that the key grammar point of each lesson is clear and the situational concept represented by a picture is easier to understand over time. Most students could tolerate the ASR problems and would enjoy using such a system, especially before coming to Japan. It is also observed that more students like to have the choice of using text input or speech input, which is now available. Some suggestions were given and adopted, for example, adding a function of listening to what students have said to help them find pronunciation errors by themselves.

7.8. System's portability

At present, the CALLJ system has 14 lessons although it is designed with 30 lessons to cover elementary Japanese. Making a lesson material (templates and grammars) for this system requires modest programming skills expected in Computer Science departments, but it needs deep insights of the target language itself. Thus, we have asked a Japanese teacher to proof the generated questions, and revised the grammars and vocabulary. Since the system architecture is not dependent on Japanese, it can be ported to other languages.

8. Conclusion

We have designed and implemented a new CALL system called CALLJ for the study of the elementary Japanese grammar and vocabulary as well as pronunciation. The system features dynamic generation of questions together with language model for ASR using decision tree-based error prediction. The system also includes an interactive hint system, allowing the students to decide how much help they need during the exercise. A scoring system is also incorporated to motivate the students to improve their language skills.

We have conducted a set of trials with the system. We collected a large amount of data regarding the errors the students made, along with the hints they used throughout the trials. With the information obtained from these data, we trained decision trees to classify errors for error prediction in ASR. In the open evaluation, the WER of ASR is 11.2%, which realizes satisfactory performance as a CALL system. We also estimated the values of all weights for the scoring system, and confirmed that the students of the same language proficiency level are clustered by the score.

We also collected the students' opinions on the system via a questionnaire. It is confirmed that the dynamic generation of questions and hints is feasible and that the hint and scoring systems are useful for students. Generally, the students enjoyed using the system and found it useful for language learning. We plan to add more content to the software, and make it open to the public via the website of our university.

References

Abdou, S.M., Hamid, S.E., Rashwan, M., Samir, A., Abd-Elhamid, O., Shahin, M., Nazih, W., 2006. Computer aided pronunciation learning system using speech recognition technology. In: Proc. ICSLP.

Bernstein, J., Najmi, A., Ehsani, F., 1999. Subrashii: encounters in Japanese spoken language education. CALICO 16, 361–384.

Eskenazi, M., Hansma, S., 1998. The fluency pronunciation trainer. In: STiLL.

Franco, H., Abrash, V., Precoda, K., Bratt, H., Rao, R., Butzberger, J., Rossier, R., Cesari, F., 2000. The SRI EduSpeakTM system: recognition and pronunciation scoring for language learning. In: Proc. InSTIL (Integrating Speech Technology in (Language) Learning).

Kawai, G., Hirose, K., 2000. Teaching the pronunciation of Japanese double-mora phonemes using speech recognition technology. Speech Comm. 30, 131–143.

Levie, W.H., Lentz, R., 1982. Effects of text illustrations: a review of the research. Educ. Technol. Res. Develop. 30, 195–232.

Nagata, N., 2002. BANZAI: computer assisted sentence production practice with intelligent feedback. In: Computer Assisted System for Teaching and Learning/Japanese.

Nelson, D.L., Reed, V.S., Walling, J.R., 1976. Pictorial superiority effect. Human Learning Memory 2, 523–528.

Neumeyer, L., Franco, H., Abrash, V., Julia, L., Ronen, O., Bratt, H., Bing, J., Digalakis, V., Rypa, M., 1998. WebGrader: a multilingual pronunciation practice tool. In: STiLL.

Smith, M.C., Magee, L.E., 1980. Tracing the time course of picture-word processing. J. Exp. Psychol. 109, 373–392.

Tsubota, Y., Kawahara, T., Dantsuji, M., 2002. Recognition and verification of English by Japanese students for computer-assisted language learning system. In: Proc. ICSLP, pp. 1205–1208.

Tsubota, Y., Kawahara, T., Dantsuji, M., 2004. Practical use of English pronunciation system for Japanese students in the CALL classroom. In: Proc. ICSLP, pp. 1689–1692.

Wang, H., Kawahara, T., 2008. Effective error prediction using decision tree for ASR grammar network in CALL system. In: Proc. ICASSP.

Witt, S.M., 1999. Use of Speech Recognition in Computer-Assisted Language Learning. PhD's Thesis.

Zinovjeva, N., 2005. Use of speech technology in learning to speak a foreign language. In: Speech Technology.