Dynamic Situation Based Sentence Generation Used in Creating Questions for Students of Japanese

Christopher Waple*, Yasushi Tsubota†, Masatake Dantsuji†, Tatsuya Kawahara*†
School of Informatics, Kyoto University
chris@ar.media.kyoto-u.ac.jp

1 Introduction

There has been much interest in the development of Computer Assisted Language Learning (CALL) systems, and the various approaches represented to their development. Some focus on correcting pronunciation errors in the students speech[1]. Others concentrate on vocabulary or grammar learning[2]. The systems also differ in the level of interactivity that they offer the student. It has been shown that the more flexible the system is in terms of the freedom it allows the student in their answers, the more beneficial the learning experience [3]. The benefits of practicing via sentence production exercises have also been shown [2].

We are developing a system to aid a student learn Japanese grammar via sentence production exercises. The student has the freedom to enter their own sentences, and receive feedback based on any errors found. To reduce the student's frustration, an interactive hint system has been included that allows the student to choose when to receive help, and how much help to receive in order to solve a task.

The system generates each question on-the-fly, thus reducing the repetitiveness that disadvantages most other systems. This involves the automatic generation of a concept that the student must describe, the valid sentences describing that concept that the student must produce, and the hints that the student may or may not choose to use. This is technically challenging, and in this paper we will present the NLP methods employed to achieve it.

We will also present some results obtained from a recent trial of the prototype system.

2 CallJ - System Design

2.1 Overview

The system is aimed at beginner to intermediate level students of Japanese. Specifically, it contains contents from levels 4 and 3 of the Japanese Language Proficiency Test (JLPT) [4]. The material specified within these boundaries consists of approximately 1,500 words (of which around 200 are verbs), 300 kanji, and 95 grammatical points. We cover these grammatical points over the course of approximately 30 lessons.

The main tasks that are carried out by the system are as follows.

- Lesson Definition: the contents of each lesson, in terms of the target grammars (sentence patterns), vocabulary and so on must be defined.
- Concept and Vocabulary Definition: each question involves the student being asked to describe a situation which is dynamically generated within the confines of the lesson. The choice of concept may be arbitrary, but appropriate vocabulary must be used depending on the verb. Concept templates which are based on the case-frame selection process are used.
- Concept to Sentence: a target Japanese sentence is generated based on the grammar specified in the lesson for a given concept and vocabulary.
- Hint Generation: a set of hints is automatically generated for each question, with an appropriate penalty assigned to each hint.
- Error Handling: any errors in the student's input must be detected and classified. Feedback must then be given on these errors.

2.2 Concept Definition

The questions for each lesson involve the student describing a situation, and the first task in generating a question is to generate this situation. A concept template covers a range of related situations, and defines the semantic components that are required, optional or omitted when defining a situation. The concept template is very similar in terms of structure and meaning to a *case-frame* [6].

Within this frame, the various information fields are known as *slots*. Once the template has been selected, the system then selects which slots are to be activated (the optional slots are decided randomly). For the active slots, the system selects an appropriate value, known as a *filler*. The fillers are selected depending on the nature of the slot specification.

Figure 1 shows a simple example of a concept template complete with example fillers, and a demonstration of the selection process. This example concerns the concept "person is", which defines the description of an aspect about a person, for instance

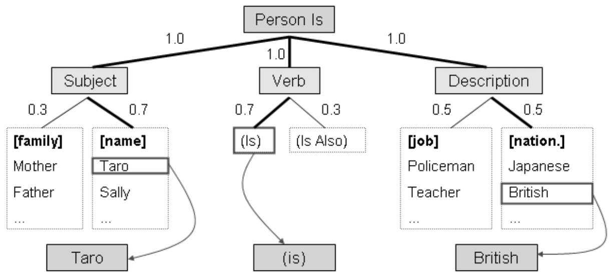

Figure 1: *Generating the concept instance*

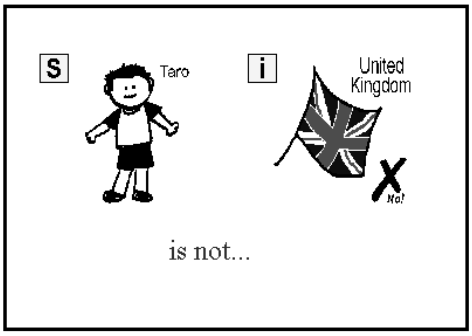

Figure 2: *The concept diagram generated from the concept instance above*

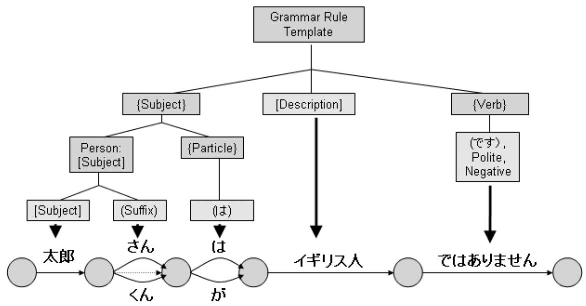

Figure 3: *Grammar based sentence generation*

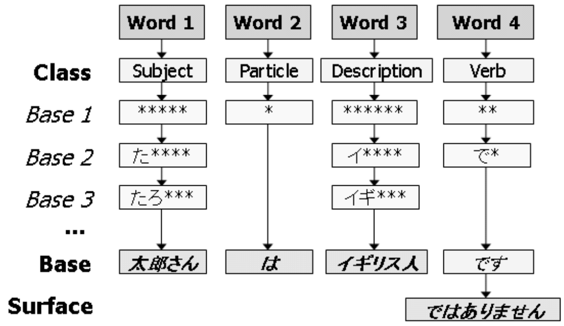

Figure 4: *The component based hint "chains"*

what job they have, or what nationality they are. The case-frame has three slots. Note that the numbers in the network represent the probabilities of each selection, and may be altered to add bias to the nature of the situations generated within a lesson.

2.3 Sentence Generation

The sentences are created in a network form, as shown in the lower half of Figure 3. 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 hierarchal 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 Figure 3. The top-level grammar rule template specifies that the sentence should consist of three components, the 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 complete word network representing the valid sentences for the given question is generated in this way, the words being defined by the leaf nodes of the grammar network.

2.4 Interactive Help System

One of the key features that distinguishes this software from more traditional methods of learning is the ability for the student to choose the level of help they receive in answering a question, something that is difficult to achieve in printed textbooks. In this system, we allow the student to uncover the target sentence word by word. Each word is not simply revealed in one step, but incrementally (character by character) allowing the student to guess the word without having it all revealed to them.

The hints are generated based on breaking down one target sentence into its constituent components, and then for each component creating an ordered set of hints. The number of hints or hint levels per component varies with the base type of that component. For example, with a verb, the final-form appropriate for the given situation is revealed as a separate hint from the base form. With nouns this extra step is not necessary. Figure 4 shows an example of a sentence being broken down into a set of hints.

We assign a "cost" to the revealing of each hint. This cost would deduct from an overall score for that question, and act as a motivator to encourage the student to attempt solving the question themselves before resorting to guidance. The idea of having a score for each question, and thus lesson, was introduced to add a more game-like feel to the software, to keep the students interested in progressing. Higher costs were assigned to those hints whose usage is shown to have a decisive impact on proficiency. The impact of different hint components and types is estimated from trial data, as covered in *Section 3.2*.

2.5 Error Handling

For the student to learn from their mistakes, it is vital that they be told where these mistakes are, the nature of the mistake, and how the mistake might be corrected. Thus, once the student enters their answer, the system must first detect if there are any errors in that answer. For each word in the student's sentence, the sentence grammar was searched to find closest matched word in the sentence position. If there is a mismatch the input word is labeled as an error.

The error classification results from comparing the features of the input word to that of the closest matched word in the target answer. Features determined and analyzed for error classification include whether both words are of the same grammatical type, whether they share semantic tags, the string distance, any inflections etc. A decision tree is then used to take these features and assign an error category. The categories used are not too dissimilar to those covered in [7], and include component insertion, deletion, and numerous types of substitutions (based on the relationship between the observed and target words). Each error is also classed as being a Grammatical error, a Lexical error, a Conceptual error, or an Input error.

The feedback given to the student for each error is determined by the error category. Each error class has a template feedback text, into which information such as the target and observed words, the types of words, any difference in inflection, and shared semantic meanings may be inserted. The feedback is displayed to the students via a dialog box as shown in Figure 4. Currently the correct answer is also displayed, but it may be desirable to allow the student to try re-entering the word based on the feedback given, an idea raised in [8].

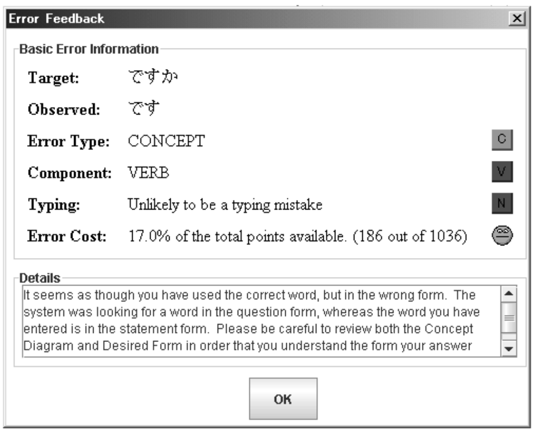

Figure 5: *The system provides feedback on errors*

3 Experiments

A trial was conducted using a prototype version of the system, with a number of students who are currently enrolled in Japanese classes at Kyoto University. The level of class that each student was enrolled in (elementary, intermediate 1, intermediate 2, or intermediate 3) was known, and this level was used as that student's proficiency level when it came to training the proficiency estimating SVM.

Each student was asked to run through a set of 8 lessons (the same set for each student, and chosen to cover a range of grammatical difficulty levels), answering 6 questions per lesson. The questions were all generated dynamically by the system, and thus varied for each student. The student was also asked to complete a questionnaire after the trial.

3.1 Student Errors and Hint Usage

We recorded all of the student errors, and classified them by the component upon which the error occurred, along with the specific error type (insertion, deletion, along with numerous types of substitution). Each error was also classified as a grammar, lexical, concept, or input based grammar. Figure 5 shows the rate per component that these errors occurred for both elementary and intermediate students. We can clearly see that, as expected, the elementary students make more errors on average, and in particular with relation to grammatical and lexical errors.

We also analyzed the hints that each student used, and broke these down by the component that the hint was based, along with the level to which the student unveiled the hint. Figure 7 shows there is a large difference in the rates observed with elementary students compared with intermediate students.

By looking at both the error and hint rates, it is interesting to see that, for elementary students, the most common errors are lexical, whilst the most often used hints are to reveal the dictionary form of a word. These two issues are clearly related, and we may infer that a student at elementary level has more problems with vocabulary than in any other area. For the intermediate student, this was also the case, but the difference between the dictionary form hint usage and other hints was not so pronounced.

3.2 Proficiency Classification

From the various error types and hint usage statistics, we trained an SVM (Support Vector Machine) classifier that takes the student's record, and estimates whether he is an elementary or intermediate student. For this task we used the libSVM library [9]. From an initial large set of features, we determined which features were most significant in producing an accurate SVM by a greedy linear regression algorithm, thus reducing the feature set. The resulting SVM mis-classifies achieved an F-Measure of 90.1%. The observed effect of each feature on

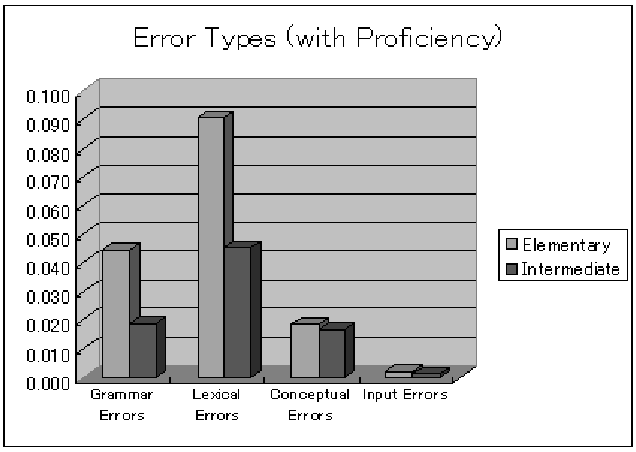

Figure 6: *A comparison of elementary and intermediate students' error rates*

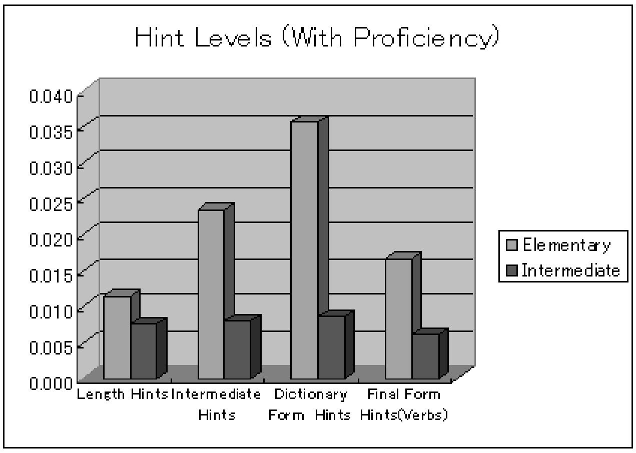

Figure 7: *A comparison of Elementary and Intermediate students' error rate hint usage rates*

the SVM's performance during the feature reduction process was used to approximate the error penalties along with the costs for using the hints. In other words, we induce that an error that is based on error type features significant to proficiency estimation should have a higher cost than one based on insignificant features.

Whilst the approach we have used has enabled us to create a seemingly efficient and accurate SVM, the actual significance of each feature is still under investigation, and thus the costs within the system are currently just interpretations of the estimated feature significance. To more accurately determine which features, and also which combinations of features, have a large impact on the proficiency rating, we should also consider other feature ranking approaches, such as those covered in [10].

3.3 Questionnaire Results

The students who took part in the trial were generally very enthusiastic about the system. On a score system of 1 (low) to 5 (high), the system scored an average of 4.2 for enjoyment, and 4.3 for perceived usefulness. Out of 21 students, 15 (71%) indicated that they would be interested (awarding a score of 4 or 5) in using such a system.

4 Conclusions

We have designed and implemented a new CALL system for students of Japanese, which allows a student to practice constructing grammatically correct Japanese sentences. The system is able to generate a large variety of questions, analyze the student's answers for errors, and give feedback on these errors. An experiment shows that it would be possible to automatically estimate a student's proficiency level from their answers. The feedback from students who took part in the trial was generally very positive, showing a strong interest in this kind of system.

References

[1] Tsubota,Y., Kawahara,T., and Dantsuji,M., "Practical Use of English Pronunciation System for Japanese Students in the CALL Classroom", ICSLP, 2004

[2] Nagata,N., "Japanese Courseware for Distance Learning", AILA, 2000

[3] Yang, J.C., and Kanji, A., "An Evaluation of Japanese CALL Systems on the WWW Comparing a Freely Input Approach with Multiple Selection", CALL Journal, Vol.12 No.1, pp.59-79

[4] Japanese Language Proficiency Test (JLPT), Japan Educational Exchanges and Services, http://www.jees.or.jp/jlpt/en/

[5] Hatasa, K., "Royalty-Free Clip Art Collection for Foreign/Second Language Instruction", http://tell.fll.purdue.edu/JapanProj/FLClipart/default.html

[6] Kawahara, D., and Kurohashi, S., "Japanese Case-Frame Construction by Coupling the Verb and its Closest Case Component", In Proceedings of the HLT Conference, pp.204-210, 2001

[7] Yang, J.C., and Kanji, A., "Error Analysis in Japanese writing, and it's implementation in a Computer-Assisted Language Learning System on the World Wide Web", CALICO Journal, Vol.15 No.1-3, pp.47-66

[8] Yang, J.C., and Kanji, A., "Development of computer assisted language learning system for Japanese writing using natural language processing techniques: A study on passive voice", Proceedings of the workshop "Intelligent Educational Systems on the World Wide Web", 1997

[9] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: a library for support vector machines", 2001, Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm

[10] Guyon, I. and Elisseeff, A., "An Introduction to Variable and Feature Selection", Journal of Machine Learning Research 3 (2003) 1157-1182