Developing an Intelligent Tutor for Japanese Honorifics using Constraint Based Modelling

Zachary Chung Takehito Utsuro
Graduate School of Systems and Information Engineering, University of Tsukuba

1 Introduction

An Intelligent Tutoring System (ITS) is a computer-based instruction system that integrates cognitive theories in instructional design and delivers adaptive instruction to learners. These tutors are designed with the underlying idea that learning occurs through context-based performance. An ITS identifies student misconceptions and provides appropriate feedback to remediate these misconceptions. An ITS is comprised of two main models: a domain model and a learner model [2].

A domain model is a representation of the subject matter being taught. It is a database of correct knowledge or reasoning modeled after an actual expert in the domain [9]. It can also support information selection and representation, problem selection and feedback generation [2]. There are many possible representations, including semantic networks, production rules and constraints, but what representation is adopted depends on the usage.

While the domain model is common to all users of the system, the learner model varies between each learner. The learner model is a representation of common misconceptions of users with respect to substandard reasoning patterns of the content. It provides descriptions of learning at a level of granularity that facilitates the encoding of principles and rules in a teaching system [9]. Commonly, this includes information such as which parts of the teaching content the student has visited, what problems were the user able to solve or not, and which concepts the user has learned. The student model provides adaptability to the ITS. The ITS decides how to proceed depending on student performance, information from the system current state, plus the information from the student model.

In this research, we use Constraint-Based Modelling as our learner model and expert model. The concept is based on Stellan Ohlsson's theory of learning from performance errors [7]. The theory states that people already have declarative knowledge learned for a task, but people still commit mistakes due to the incomplete internalization of declarative knowledge as procedural knowledge. However, by practicing a task and catching ourselves (or by a mentor) making a mistake, we modify our procedural knowledge to incorporate the appropriate rule we violated. Over time, once declarative knowledge about the task has been internalized, the number of mistakes we commit is reduced.

2 Task-Based Language Learning

Conventional language learning is focused on acquiring the form and structure of language as concepts, however this alone is insufficient. The key to language acquisition is communication in an applied setting [8]. Hence, task-based language learning is more effective compared to the conventional approach because it offers students the opportunity to acquire language while they engage in communication to achieve a task. The task relates to students' real-life language needs, which develop intrinsic motivation to learn a language. In the process, students acquire a deeper understanding of language since they not only learn vocabulary and expressions, but also the correct context of their application according to a need.

3 Task-Based Learning of Japanese Honorifics

3.1 Proposed Research Framework

Language acquisition in controlled environments is an example of learning from performance errors [3]. Language is taught to students as a kind of declarative knowledge by providing sample utterances, facts about word meanings and their grammatical properties, rules about how to combine words to build a meaningful language construct and guidelines about the appropriateness of expressions given a context. This learning process is complemented by exercises to help students internalize the declarative knowledge.

In this research, we aim to facilitate learning Japanese honorifics through task-based learning.

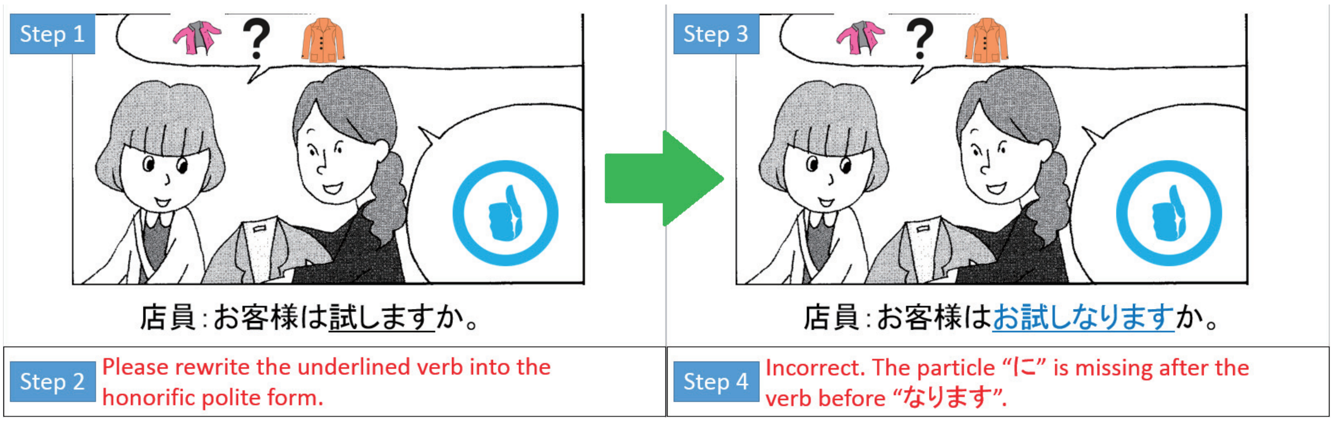


Figure 1: Dekiru Keigo User Interface: Honorific Polite Form Tutoring

Since language production is an activity primarily guided by a speaker's personal intentions towards the completion of a task, we create an explicit exercise environment by providing a textual or pictorial description of a real-world scenario. In this way, we narrow down the space of possible intentions without rendering the language activity completely unnatural by taking full control of a person's intentions. At the same time, we provide contextual learning through application and we support this learning environment by using CBM.

3.2 Dekiru Keigo

Dekiru Keigo is a task-based tutor for learning Japanese honorifics (See Figure 1). Learners are provided a task in which they learn how to use Japanese honorifics in an applied setting. In this scene (step 1), the learner takes on a store clerk helping a customer find a jacket that fits her needs. The store clerk (learner) is asking the customer if she wants to try a coat, which the learner must ask using the honorific polite form of the verb (step 2). In the next scene (step 3), the learner tried to use the *naru* honorific, however the learner committed an error generating the *naru* honorific of "試します", which is supposed to be "お試しになります". As feedback (step 4), the learner is shown that he failed to add the particle "に" after the verb before "なります" and the learner has to correct his input for resubmission.

4 Constraint Based-Modelling

4.1 Learning from Performance Errors

According to Ohlsson [7], the learning process from errors occur in two phases: *error recognition* and *error correction*. A student requires declarative knowledge to detect an error. An ITS may play the role of the mentor and identify the mistakes of the student. After detection, these errors can be corrected in context so the student learns when and where to apply the solution [4]. The basic idea of Constraint-Based Modelling (CBM) is to equip a tutor with a set of constraints for a target domain, and to inform the learner about his constraint violations. Each constraint describes a pedagogically significant state, an important concept the student should learn.

Constraints in CBM represent both the domain and student knowledge, where each constraint represents the basic principles of the underlying domain [1]. It works by representing the domain in the form of constraints on correct solutions and it serves to identify errors, which is important for students lacking declarative knowledge because they are unable to detect errors themselves [4]. Each constraint violation represents a domain concept the student is not conforming to, which is subject to remedial action.

4.2 Definition of a Constraint

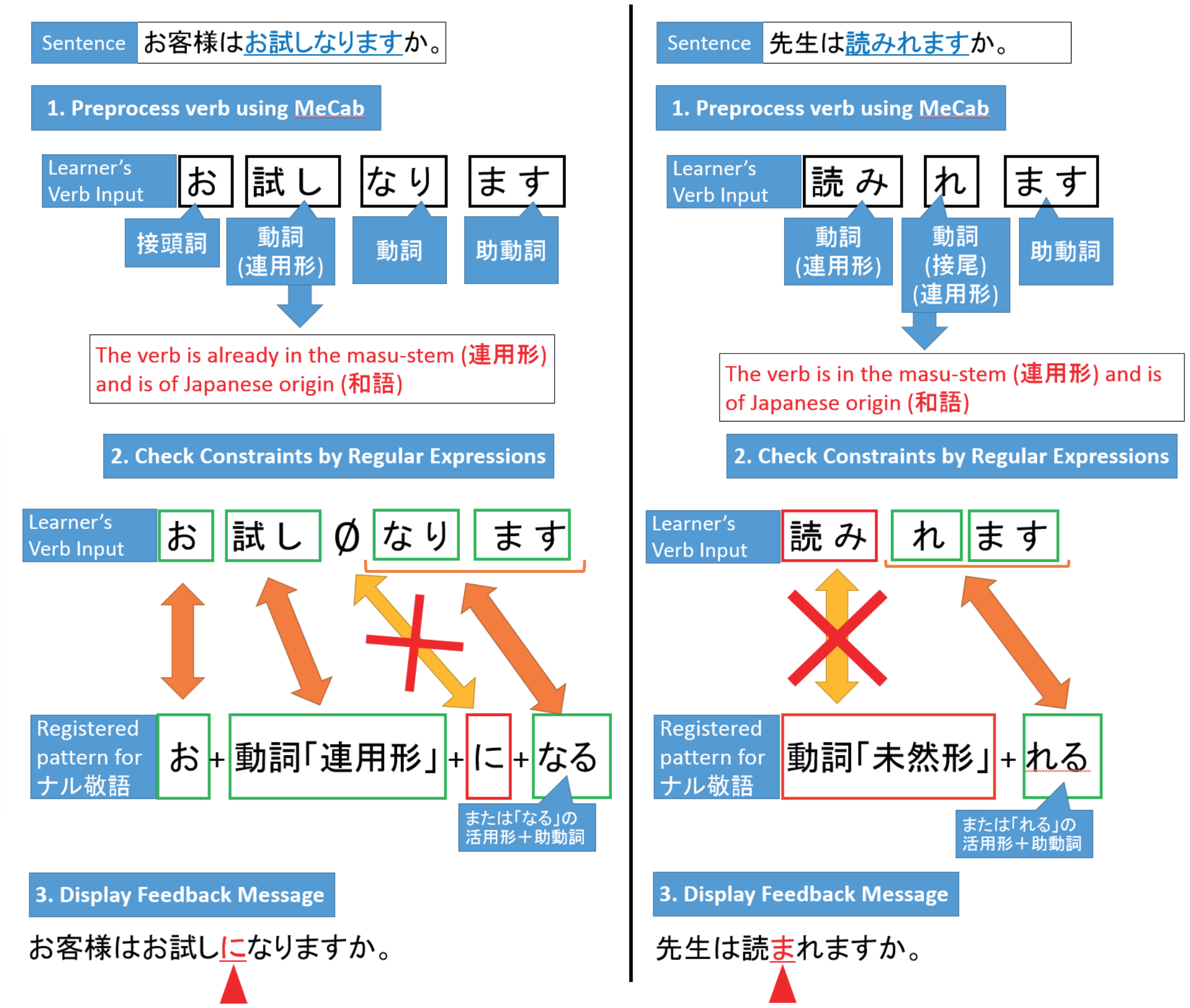
A constraint is an ordered pair with an associated feedback message.

$$(C_r, C_s)$$

C_r is the relevance condition, which describes when the constraint is applicable. C_s is the satisfaction condition, which specifies tests to check solution validity. If C_r is satisfied in a problem state, in order for that problem state to be correct, it must also satisfy C_s. Otherwise, feedback is provided depending on which relevant constraints had their satisfaction condition violated [5, 6].

4.3 Characteristics of CBM

CBM is computationally simple because it does not require an expert model and can be implemented by pattern matching; student diagnosis is performed by using constraints to compare a student's solution to a specified ideal correct solution [6]. CBM can also support multiple solutions because constraints



(a)ナル敬語 Constraint Checking

(b)レル敬語 Constraint Checking

Figure 2: Constraint Checking for Honorific Forms

can be made to identify alternative constructs in solutions that are equally valid [5]. Since language use is a problem of choice between multiple forms of expression, this feature is advantageous as it possible to support multiple answers. Finally, CBM assumes that diagnostic information lies with the problem state and not the solution process. Meaning, the learner is free to make any solutions as long as the learner never reaches a state defined to be wrong.

5 Applying CBM to Learning Japanese Honorifics

5.1 System Inputs and Outputs

The current system developed for this research is focused on teaching the syntax of honorific polite expressions. Given a verb and the expected honorific form from the learner as parameters, the system checks the input and shows feedback as necessary. If the verb is conjugated into the expected form correctly, the system allows the learner to continue around the tutoring content. Otherwise, the system shows a feedback message to inform the learner of his error, what must be done to correct his input and to resubmit his answer. The process repeats until errors are cleared and the learner is allowed to proceed.

5.2 System Design

In general, the system works in three steps: verb preprocessing using MeCab[1], constraint checking by regular expressions and feedback display. The current system supports checking for two honorific polite forms: *naru* honorific and *reru* honorific. In forming the *naru* honorific (Refer to Figure 2 (a), ナル敬語 Constraint Checking), we need to identify whether the verb is Japanese (和語) or Sino-Japanese (漢語) in origin by MeCab preprocessing. If it is Japanese, the verb should be in the *masu*-stem (連用形). Otherwise, if it is Sino-Japanese (typically *suru*-verbs), only the noun portion without *suru* should be in the input. With “お試しなります”, since “試し” is a Japanese verb, we confirm that the learner has the

[1] MeCab: Yet Another Part-of-Speech and Morphological Analyzer: `http://taku910.github.io/mecab/`

verb in the *masu*-stem and proceed to form checking by constraints.

Using the verb type information and the expected honorific form as parameters, we load only necessary constraints. For each constraint, the relevance condition (C_r) is the verb type and the expected honorific form. The satisfaction condition (C_s) is a regular expression that describes the formation of the expected honorific polite form. For example, in "お試しなります" in Figure 2 (a), we load only constraints about the formation of *naru* honorifics for Japanese verbs and compare the verb input to each single constraint. With "お試しなります", the learner violated the constraint that the particle "に" should follow the verb before "なります". As a result, the learner is asked to correct and resubmit his answer based on an associated feedback message. The input is checked again and the process repeats until no constraint violation is made. While we only describe the *naru* honorific constraint checking, the same process applies for *reru* honorifics (See Figure 2 (b) レル敬語 Constraint Checking).

6 Related Literature: Model-Tracing

Model-Tracing (MT) is an alternative to CBM, based on the Adaptive Control of Thought-Rational theory. The theory states that humans have two memory stores: declarative and procedural and that learning is in two phases. First, people acquire declarative knowledge in the form of facts, then it is turned into procedural knowledge which is goal-oriented. Procedural knowledge is represented as production rules, which are optimized as the student becomes an expert. The fundamental assumption of this approach is that cognitive skills are realized by production rules. Hence, MT tutors organize instruction around production rules (problem-solving steps) in order to help students learn how to perform a task correctly [4].

MT checks if a student is performing correctly by comparing each of the problem solving steps taken by the student with respect to a solution path. A student solution must conform to a solution path for it to be acceptable, otherwise, it is rejected. While this endows high cognitive fidelity by teaching an explicit model of reasoning to the learner, MT-based tutoring is rigid as it requires problem solving with a specific order. In contrast, CBM only evaluates the current problem state (as opposed to the current action with MT-based tutors). As long as no constraint defined is violated, the students are free to write their solutions in any way using whatever construct they see fit.

7 Conclusion

CBM as a student model is appropriate to language tutoring due to its solution flexibility. While MT may be posed as an alternative to CBM, MT is disadvantageous as it requires modelling solution paths and all possible answers. This is impossible given that language production is not an unobservable process and there are too many forms of expression. Besides flexibility, CBM is easier to implement because it both works as an expert model and a learner model in the form of constraints and it can easily be implemented by pattern-matching (regular expressions). In our current work, we have implemented a syntax checker for Japanese honorific polite forms using CBM as a student model.

For further research, we plan to incorporate dependency analysis in our system so the system can process entire Japanese sentences and identify which verbs need to be in honorific polite forms depending on the sentence subject. We are also working on creating a task-based environment to support context-based learning, so students learn when to use Japanese honorifics. Finally, we plan to automate the constraint generation process using statistical natural language processing using a corpus.

References

[1] B. Martin. Constraint-based modeling: Representing student knowledge. *New Zealand Journal of Computing*, 7:30–38, 1999.

[2] B. Martin. Intelligent tutoring systems: The practical implementation of constraint-based modelling, 2001. `http://ir.canterbury.ac.nz/handle/10092/4834`.

[3] W. Menzel. Constraint-based modeling and ambiguity. *International Journal of Artificial Intelligence in Education*, 16:1–35, 2006.

[4] A. Mitrovic, K. Koedinger, and B. Martin. A comparative analysis of cognitive tutoring and constraint-based modeling. In P. Brusilovsky et al., editors, *User Modeling 2003*, volume 2702 of *LNCS*, pages 313–322. Springer, 2003.

[5] A. Mitrovic, B. Martin, and P. Suraweera. Intelligent tutors for all: The constraint-based approach. *IEEE Intelligent Systems*, 22(4):38–45, 2007.

[6] A. Mitrovic, M. Mayo, P. Suraweera, et al. Constraint-based tutors: A success story. In L. Monostori et al., editors, *Engineering of Intelligent Systems*, volume 2070 of *LNCS*, pages 931–940. Springer, 2001.

[7] S. Ohlsson. Learning from performance errors. *Psychological Review*, 103(2):241–262, 1996.

[8] J. Richards and T. Rodgers. *Approaches and methods in language teaching.* Cambridge University Press, 2001.

[9] B. Woolf. *Building Intelligent Interactive Tutors: Student Centered Strategies for Revolutionizing E-Learning.* Morgan Kaufmann Publishers, 2009.