AN EFFECTIVE APPLICATION OF NATURAL LANGUAGE PROCESSING IN SECOND LANGUAGE INSTRUCTION

Noriko Nagata, University of San Francisco

ABSTRACT

This paper presents an intelligent CALI system called "**Nihongo-CALI**" (Japanese Computer Assisted Language Instruction), which employs natural language processing to provide immediate, grammatically sophisticated feedback to students in an interactive environment. Using this system, Nagata (1993) previously compared the effectiveness of the two different levels of computer feedback for teaching Japanese passive sentences: traditional feedback (which follows simple pattern-matching error analysis and indicates only missing/unexpected words in the learners' responses) and intelligent feedback (which utilizes a parsing technique to provide detailed grammatical explanations for the source of the learners' errors). The study found a statistically significant difference between traditional and intelligent feedback, favoring intelligent feedback. The present study compares the efficacy of intelligent CALI feedback with that of a more advanced, traditional CALI feedback (which also indicates the positions of missing particles in the learners' responses) for teaching basic sentence constructions in Japanese. The result indicates that the Intelligent CALI feedback is more effective than even the enhanced version of traditional CALI feedback, underscoring the importance of natural language processing technology in second language instruction.

KEYWORDS

Natural language processing, artificial intelligence, parsing, CALI, ICALI, Japanese, error analysis, feedback, second language instruction.

INTRODUCTION[1]

Natural Language Processing is an artificial intelligence technique that parses linguistic input according to the grammatical rules and lexicon of the target language. Many researchers have been developing intelligent CALI (computer assisted language instruction) or CALL (computer assisted language learning) systems employing such parsing techniques. For example, **Grammar-Debugger** (Si-Quing and Luomai 1990) is a parser designed to analyze both grammatical and ungrammatical English sentences typically produced by Chinese EFL learners and to provide feedback in response to their errors. **LINGER** (Yazdani 1991) consists of a language-independent parsing mechanism, a language-specific dictionary, and a language- specific grammar to cope with most aspects of Romance and Germanic languages. **Miniprof** (Labrie and Singh 1991) is a program designed to provide French grammar exercises, in which the system parses the learners' responses and provides tutorial messages. **ALICE-chan** (an extended version of **ALICE**, Levin, Evans, and Gates 1991) is a multimedia workbench that includes tools and resources for syntactic parsing, morphological analysis, error detection, on-line dictionaries, and the development of instructional programs for Japanese. Liou (1991) describes a program using a parser to check the grammar in EFL students' compositions. **Syncheck** (Sanders 1991; Juozulynas 1994) is a syntactic parser designed to help learners writing German compositions. **GPARS** (Loritz 1992) is a generalized transition network system which includes parsers for English, Russian, Japanese, and Chinese. **BRIDGE** (Holland, et al. 1993) is a parser-based German tutor which allows learners to respond with free form sentences.

The motivation for developing intelligent CALI (ICALI) systems is to analyze learners' response sentences in more general way (using the grammatical rules and lexicon of the target language rather than numerous pre-stored correct or incorrect sentences) in order to facilitate more sophisticated error analysis and feedback than the traditional CALI systems can provide. When developing intelligent CALI systems, we must think of what intelligent CALI can really do better than traditional CALI. Otherwise, we may end up using sophisticated parsing techniques to produce feedback that could have been generated more efficiently using simple traditional techniques. Loritz (1992) tested **ENGRASP** (an English parser of **GRASP** mentioned above) with several classes of deaf students. He reports that one third of the students would type an erroneous sentence like “My brother like me,” and sit bewildered before a screen which presented them an error message such as "add *s* to *like*." He conjectures that their learning style was not sufficiently developed to pursue useful interaction with the system (p. 17). But we also need to think of the quality of the error message provided. This type of error message (i.e., pointing out what is missing and where it is missing) is easily and typically implemented in traditional CALI using simple pattern-matching techniques. One of the touted features of intelligent CALI is its capability to provide detailed explanations about the nature of the errors. If we use an intelligent system, we should examine carefully what kind of error messages should be provided by intelligent CALI and how effective they are. The existing intelligent CALI systems show how the systems work and what they can do, but there have been very few empirical studies demonstrating instructional effectiveness of intelligent CALI compared to traditional CALI. It seems reasonable to suppose that intelligent CALI is better than traditional CALI, but we need empirical studies to know more about the conditions in which this is the case. The purpose of this paper is to present an intelligent CALI system called **Nihongo-CALI**[2] (Japanese Computer Assisted Language Instruction) (Nagata 1992, 1993) and to investigate the relative effectiveness of intelligent and traditional CALI for teaching Japanese sentence constructions.

The next section will describe Japanese sentence constructions briefly.

SENTENCE CONSTRUCTIONS IN JAPANESE

As has been described in detail in Kuno (1973), Japanese can be characterized broadly as a left-branching, verb-final language. Among the surface features of Japanese syntax, post-positional particles are employed to mark syntactic and semantic roles. For example, the particle *ga* is used to indicate the subject who performs the action, the particle *o* to indicate the object that the action operates upon, and the particle *ni* to indicate the direction which the action moves toward or the goal of the action. In the sentence, *John ga Mary ni tegami o kaita*,[3] "John wrote a letter to Mary," *John ga* indicates the subject, *Mary ni* the direction/goal, and *tagami o* the object. Although Japanese has an underlying SOV order, the word order before a verb is relatively flexible because each noun phrase is followed by a case marker that indicates its grammatical function unambiguously. For example, the above sentence may be expressed as *John ga tegami o Mary ni kaita, Mary ni John ga tegami o kaita, Tegami o John ga Mary ni kaita*, and so forth. If we replace one particle with another in a sentence, the meaning of the sentence becomes quite different; for example, compare *John [ga] tukaimasu*, "John will use it" and *John [o] tukaimasu*, "I will use John (for some task)." There are several location markers in Japanese. For instance, the particle *de* marks the location where an activity takes place such as *[Tokyo de] shyoppingu o simasita* "I did shopping [in Tokyo]." The particle *ni* indicates the location where a non-activity referent is located such as *John ga [Tokyo ni] imasu* "John is [in Tokyo]," as well as the direction/goal mentioned above. The particle *o* indicates the location through which the action moves such as *[Kooen o] arukimasita* "I walked around in a park," as well as the object mentioned before. Japanese employs over ten particles. Different particles make fine semantic distinctions that may not be made in the first language. The same particle can play multiple semantic roles, too. Certain pragmatic functions (e.g., marking topic and/or contrast using the particle *wa*) may not occur'n the first language. In short, the Japanese sentence constructions involve a complex particle system which is, in fact, a major hurdle for second-language learners of Japanese at the college level. The learners who have trouble with Japanese particles usually do not have a clear picture of grammatical and semantic functions encoded by Japanese particles. This would seem to be an area in which systematic grammatical instruction and ongoing grammatical feedback for learners' errors could be useful.[4] This study uses the **Nihongo-CALI** system to provide such detailed grammatical feedback and investigates the effectiveness of intelligent CALI compared to traditional CALI for teaching the Japanese sentence constructions.

The next section describes the **Nihongo-CALI** system.

THE NIHONGO-CALI SYSTEM

The **Nihongo-CALI** system employs natural language processing to provide immediate grammatically sophisticated feedback to students in an interactive environment. The system diagnoses errors in three steps.

Step 1 Check unexpected and missing words in a learner's response by a simple pattern-matching analyzer (which is written in the cT language[5]). The simple pattern-matching analyzer does not have a lexicon and grammatical rules to parse a sentence, so the analysis is limited to checking only whether the learner's input exactly matches the one stored in the computer as a string.

Step 2 Send the sentence to an NLP (natural language processing) analyzer (Which is written in Lisp). The NLP analyzer includes a morphological parser (Hausser 1989), a syntactic parser (Tomita, et al. 1988), and three kinds of data bases: a core lexicon, a set of morphological rules, and a set of syntactic rules (Nagata 1992). The NLP analyzer parses the sentence and generates information on the grammatical properties of each component of the sentence. If the sentence is ungrammatical, error messages are generated.

Step 3 Send the grammatical information generated by the NLP analyzer back to the simple pattern-matching analyzer to check whether the meaning of the sentence is appropriate for the question in the exercise. This checking is necessary because even though the learner's response sentence is grammatical, it may not be appropriate for the particular exercise. Finally, error messages are presented on the screen.

The system runs both on a Macintosh personal computer (with minimum of five MB of RAM and five MB of disk) and on a workstation (DECstation). At present, the morphological rules in **Nihongo-CALI** handle basic Japanese conjugations such as perfective and imperfective, negative and affirmative, distal (formal) and direct (informal), passive and active, gerund, and extended-predicate forms. The syntactic rules handle Japanese simple sentences (both active and passive) which include the particles *wa*, *ga*, *o*, *ni e*, *de*, *no*, *kara*, *made*, *ka*, *yo*, and *ne*. The vocabulary of the system has about 1000 words including verbal and adjectival roots, suffixes for conjugations, nominals, pronouns, adverbials, and particles. The morphological and syntactic rules and the lexicon in **Nihongo-CALI** are expandable.

Sentence-production exercises are implemented in **Nihongo-CALI**, in which students are asked to type Japanese sentences in light of a communicative context provided in English. The exercises are designed for beginning and intermediate students to practice various sentence constructions in Japanese.[6] The Japanese writing systems (kana and kanji) are not incorporated into the **Nihongo-CALI** exercises. Since it takes at least a semester to learn kana (i.e., hiragana and katakana), and another semester to start using kanji (according to the typical college curriculum), integration of kana and kanji into **Nihongo-CALI** exercises should be targeted for students who finished at least first semester Japanese. Implementation of the Japanese writing systems into **Nihongo-CALI** is a future project.

The **Nihongo-CALI** exercise program is written in cT (together with the simple pattern-matching analyzer mentioned in the above). Figure 1 illustrates one of the **Nihongo-CALI** exercises as it is presented on the computer screen. The students are asked to type a response sentence next to the arrow prompt on the bottom of the screen.

Options

I. Respond to the following conversations, according to the situation described. In this exercise, do not drop any phrase particle.

(I) Your friend has asked if you ate lunch. Tell him that you ate sushi in Japan Town.

>

Figure 1: A Nihonogo-CALI exercise

To respond to this question, suppose a student typed, "*Nihonmati ni susi ga tabemasu*," which includes two particle errors and a verbal tense error. A correct answer would be something like "*Nihonmati de susi o tabemasita*."[7] The **Nihongo-CALI** exercise program stores each exercise along with a list of grammatical information for the expected answers to the question. The grammatical information is used to check what words are unexpected or missing in the learner's response. For example, the following shows some of the grammatical information involved in this exercise:

1. Some of the grammatical information for the expected answers to the question

required = "nihonmati, de, susi, o"

citation = "taberu"

predicate = "affirmative, perfective, distal"

location_de = "nihonmati"

object = "susi"

The above information means that the required words for this exercise are "*nihonmati, de, susi, o*," the citation form of the verbal predicate is "*taberu*," the location *de* is "*nihonmati*," the object is "*susi*," the verbal predicate is "affirmative, perfective, distal," and so forth. First, the simple pattern-matching analyzer uses this grammatical information to search for unexpected/missing words in the learner's response, and generates such error messages as "*Ni* is not expected to be used here," "*Ga* is not expected to be used here," "*De* is missing," and "*O* is missing." (The correctness of the verbal tense is checked in Step 3.) Next, the sentence is sent to the NLP analyzer to check whether it is grammatical or not. The NLP analyzer contains a core lexicon, a set of morphological rules, and a set of syntactic rules to parse not only grammatical sentences but also ungrammatical ones. If an ungrammatical sentence (e.g., a sentence which includes an incorrectlyusedparticle,anincompleteverbalco 'ugation,etc.)isparsedlerror n) messages will be Venerated. The following shows some of the lexical items (used in the above sentence) in the core lexicon.

2. Some of the lexical items in the core lexicon.
("nihonmati" (<cat noun> <place +>) nihonmati)
("ni" (<cat prt) <particle ni> ni)

```
("susi" (< cat noun >) susi)

("ga" (< cat prt > < particle ga > ga)

("tabe" (< cat verb > < class c1 > < citation taberu > < o + > < gloss eat>)
tabe)

(“masu" (< cat sx > < aspect imperfective > < neg affirmative > < style
distal >) masu)
```

Each word or part of a word is defined by a list of < feature value > statements in the core lexicon (we can call the one listed on the left side of the parentheses "feature" and the one in the right side a "value"). For example, "*nihonmati*" is defined by <cat noun> (i.e., "category" is "noun") and <place + > (i.e., the word is a "place" word). “*Ni*" is defined by <cat prt> (i.e., "category" is particle") and <particle ni > (i.e., "particle" is "*ni*"), and so forth. The verb "*tabemasu*" consists of "*tabe*" < cat verb > (i.e., "category" is "verb") and "*masu*" <cat sx> (i.e., "category" is "suffix"): the morphological parser combines “*tabe*" and "*masu*" as a conjugated verb "*tabemasu*." After each word in the identified by the core lexicon and the morphological parser, the sentence is sent to the syntactic parser. The following are some of the grammatical rules involved in parsing a sentence including a *ni*-marked noun phrase (NP).

3. Some of the grammatical rules required to parse a sentence including a *ni*-marked NP

```
(<S> <==> (<NP> <S>)
        ((((- 1 place) = c +)
        ((xi particle) =c ni)
        ((xi syn feedback) > (xi value))
        ((x2 cat) = c verb)
        (*EOR* (((x2 place-ni) = c +)
                ((x1 syn feedback) > this-is-location-ni))
                (((x1 syn feedback) > (x1 value))
                ((x1 syn feedback) > should-be-de-not-ni)))
        ((x0 place) = x1))
         (x0 = x2)))
```

If a condition leading to a boldfaced label (e.g., "**this-is-location-ni**") is satisfied, then the program finds a corresponding message template paired with this label in a database (e.g., "*x1* is the "location" where a non-activity referent is located."). When this happens, I say that the boldfaced label is activated to obtain the corresponding message template. The variable **x1** in the message template is then filled with an actual word marked by the particle *ni* in the sentence. If a parsed sentence consists of a noun phrase of "place +" and "particle ni" but not a verb of "place-ni +," then the label "**should-be-de-not-ni**" is activated to obtain the message template, "Particle error: *ni* after *x1* is wrong. *Ni* is used with the verb *arimasu* or *imasu* to indicate the "location" where the non-activity referent is located. *x2* is the "location" where the activity takes place. Use *de* to mark it." Since the verb "*tabemasu*" in the above sentence is not defined by "place-ni +' (i.e., it does not take a *ni*-marked place noun), the label "**should-be-de-not-ni**" is activated to obtain the error message, and the variables *x1* and *x2* are both filled with a word "*Nihonmati*" "Japan Town." The following illustrates the final grammatical information on the sentence generated by the NLP analyzer.

4. Grammatical information generated by the NLP analyzer

   Susi is the "subject" who performs the action.

   Tabemasu is a verbal predicate (affirmative, imperfective, distal, non-passive, *taberu* "eat").

   Particle error: *Ni* after *Nihonmati* is wrong. *Ni* is used with the verb *arimasu* or *imasu* to indicate the "location" where the non-activity referent is located. *Nihonmati* is the "location" where the activity takes place. Use *de* to mark it.

All grammatical information in (4) is sent to the simple pattern-matching analyzer to check whether the meaning of the sentence is appropriate to the question in the exercise. The simple pattern-matching analyzer detects any gap between information in (1) (grammatical information for the expected answers to the question) and information in (4) (grammatical information generated by the NLP analyzer). For example, the information in (4) says that *susi* in the learner's response is the subject (which is syntactically possible but pragmatically odd because *susi* does not eat anything), while the information in (1) says that *susi* should be the object, This unmatched grammatical function generates an error message In your sentence, *susi* is the "subject who performs the action but it should be the "object" that the action operates upon. Use the particle *o* to mark it." The information in (4) also says that the verbal predicate in the learner's response is "imperfective whereas the information in (1) says that the verbal predicate should be "perfective." This unmatched information again generates an error message, "The predicate you typed is in the imperfective form. Change it to perfective." Finally, the following error messages will be provided for the student on the screen.

5. Error messages on the screen

Read the following messages:

< Particle error >

Ni is not expected to be used here.

Ga is not expected to be used here.

De is missing.

O is missing.

Ni after *Nihonmati* is wrong. *Ni* is used with the verb *arimasu* or *imasu* to indicate the "location" where the non-activity referent is located. *Nionmati* is the "location" where the activity takes place. Use *de* to mark it.

In your sentence, *Susi* is the "subject" who performs the action, but it should be the "object" that the action operates upon. Use the particle *o* to mark it.

< Predicate error>

The predicate you typed is in the imperfective form. Change it to perfective.

Using this system, Nagata previously investigated the effectiveness of the two different levels of computer feedback on the production of Japanese passive sentences: traditional feedback and intelligent feedback. The traditional feedback was developed without using the NLP analyzer (i.e., skipping Step 2). The traditional feedback was similar to the feedback provided by the **PLATO** system (Hart 198 1), which was regarded as a very sophisticated type of feedback among traditional CALI programs. That is, traditional feedback indicates what is missing, unexpected, or wrong, but does not explain the nature of the errors. The intelligent feedback was developed using both the simple pattern-matching analyzer and the NLP analyzer (i.e., using Step 1 through Step 3). The study found a statistically significant difference between traditional and intelligent feedback in the learners' achievement in producing Japanese passive sentences, favoring intelligent feedback (Nagata 1992, 1993; Nagata and Swisher 1995). When the students committed a particle error in the study, traditional feedback informed them which particles were not expected to be used and which particles were missing, but did not tell them the location of the missing particles. On the other hand, intelligent feedback provided grammatical explanations of why the particle is wrong, including the location of the expected particle. This leaves open the question whether the extra effectiveness of intelligent feedback was primarily due to the fact that it specifies where particles are missing rather than merely informing the student that they are missing.

In the present study, two CALI programs were developed using the **Nihongo-CALI** system. One is an intelligent CALI program and the other is a traditional CALI program-. these programs are similar to the ones used in the previous study, except that the traditional program in the present study also specifies the positions of missing particles in the sentence. (In order to eliminate the effects of different runtimes, both programs use all three steps of error analysis explained in the above, but the traditional program provides only limited information.) The study compares the efficacy of intelligent CALI with that of more advanced traditional CALI.

THE STUDY

Eighteen students in a first-semester Japanese course at the . University of San Francisco participated in this study. The students received six ' one-hour experimental sessions (two classroom sessions and four computer sessions) over the course of two weeks. The target structures in this study were Japanese sentence constructions using the particles, *ga, o, wa, ni,* and *de*. All Japanese particles were new to the students. The students were paired based on the scores they obtained on the midterm exam and were randomly divided into either Group 1 or Group 2, so that the two groups had no significant difference in language proficiency before they started the experimental sessions.

The first session was conducted in the classroom: the students were given a brief (thirty-minute) introduction to the particles and a pretest. The pretest included 10 blanks to be filled in with appropriate particles and 2 questions requiring t students to construct Japanese sentences using the particles. A perfect score on the pretest was 18.5.[8] The students' scores were converted into percentages.

The second through fifth sessions were conducted at the CIT laboratory (Center for Instruction and Technology, University of San Francisco).[9] The students were provided with the two CALI programs described in the preceding section. Group 1 received the intelligent version and Group 2 received the modified traditional version. Both programs provided the students with (a) the same grammar notes which explained the grammatical and semantic functions of the Japanese particles and the constructions of Japanese sentences using these particles and (b) the same sentence-production exercises. In each computer session, the students spent seven to ten minutes to read a grammar note, and then moved to the exercises. The difference between the intelligent CALI program and the traditional CALI program was the content of the feedback given to the students when they made errors. For example, the students were given the following exercise, "You've told your colleague that you did exercise yesterday and felt good. She has asked what you did. Tell her that you played tennis in a park." To respond to this question, suppose a student typed, "*Kooen ga tenisu o simasu*," which should be "*Kooen de tenisu o simasita*." The traditional CALI program provides the following feedback for this response:

6. Traditional CALI feedback

Read the following messages:

< Particle error >

Ga is not expected to be used here.

De is missing.

De should be attached to *kooen*.

<Predicate error>

The form of the predicate you typed is wrong. Change the form.

The message "*De* should be attached to *kooen*," informs the student of the correct position for the missing particle de. Intelligent feedback, however, includes .more detailed grammatical information on the error as follows:

7. Intelligent CALI feedback

Read the following messages:

<Particle error>

Ga is not expected to be used here.

De is missing.

In your sentence, *kooen* is the "subject" who performs the action, but it should be the "location" where the activity takes place. Use the particle *de* to mark it.

<Predicate error>

The predicate you typed is in the imperfective form. Change it to perfective.

The errors are classified into vocabulary errors, particle errors, and verbal predicate errors. Both Group 1 and Group 2 received the same messages for the vocabulary error (i.e., missing or unexpected words), while Group 1 received more detailed feedback than Group 2 for the particle errors (regarding grammatical and semantic functions of the particles in question) and verbal predicate errors (regarding perfective or imperfective and negative or affirmative). When the verbal predicate used is a different verb or an incomplete form, both traditional and intelligent feedback returned such a message as "Use the predicate *sirnasu* in an appropriate form."

Through four computer sessions, the students learned sentence constructions using nine different particle functions. The first computer session covered the subject marker *ga*, the object marker *o* and the topic/contrast marker *wa*.[10] The second computer session introduced the double *ga* constructions[11] the location marker *ni* and the location marker *de*.[12] The third computer session included the direction marker *ni*, the instrument marker *de*, and the location marker *o*.[13] The fourth session reviewed the sentence constructions including all particles practiced in the previous sessions. When the students were doing the exercises, they could go back to see the grammar note that they read in the beginning of the session. They also had access to the vocabulary hint which shows a list of words expected to be used for each exercise (although particles were not listed in the vocabulary hint). The students received a total of 67 exercises through four computer sessions.

A questionnaire was administered at the end of the last computer session. On the questionnaire, the students were asked to rate 23 items by a 5-point scale (1 strongly disagree, 2 disagree, 3 undecided, 4 agree, and 5 strongly agree) and to write comments on the computer program. The questionnaire was used to investigate whether there is any significant difference between the two groups in their attitudes toward the computer program.

The sixth session was conducted in the classroom to give the students a posttest. The posttest included all questions used in the pretest (10 fill-in-the-blank and 2 sentence-production questions) plus 10 fill-in-the-blank and 9 sentence-production questions which followed the same format and difficulty as those given in the pretest. All questions on the posttest were similar to those provided by the computer exercises. The perfect score on the posttest was 71 and the students' scores were converted to percentages.[14] The posttest was analyzed to see whether there was any significant difference between the two groups in their performance in using the Japanese particles and in producing Japanese sentences. The study also investigated whether there is any significant difference between the pretest scores and the posttest scores in each group and whether there was any significant difference between the two groups in their gain scores from these two tests.

THE RESULTS AND DISCUSSION

In the first session, the students took a pretest after a brief lecture on the Japanese particles. Table 1 shows the means and standard deviations of the two groups for the pretest scores, and the result of the t-test for these correlated groups. The result shows that there is no significant difference between Group 1 and Group 2 in their pretest scores.

	Mean	SD	t	Sig of t
Group 1 (n=9)	53.8	9.5	0.15	NS
Group 2 (n=9)	52.9	20.9		

Table 1: The result of the pretest (scores out of 100)

Next, the student took the computer sessions: Group 1 received the intelligent feedback and Group 2 received the traditional feedback. Table 2 presents the means and standard deviations of the two groups for the posttest scores.

	Mean	SD	t	Sig of t
Group 1 (n=9)	80.1	12.4	3.04	0.02
Group 2 (n=9)	67.5	21.1		

Table 2: The result of the posttest (scores out of 100)

The t score shows that the difference between Group 1 and Group 2 in their performance on the posttest is statistically significant at the 0.02 level, favoring Group 1. Table 3 presents the numbers of different types of errors observed on the posttest.

	Particle errors in fill in the blank	Particle errors in sentence production	Verbal errors in sentence production	Vocab. Errors in sentence production
Group 1 (n=9)	40	66	17	17
Group 2 (n=9)	65	102	24	39

Table 3: The numbers of different types of errors on the posttest

The counts of different types of errors on the posttest reveals that the significant difference between the test scores of the two groups resulted largely from particle errors. This is consistent with the finding in the previous study. The present study confirms that the effectiveness of intelligent feedback in the previous study was not due to the fact that they specify where particles are missing but was due to the extra explanations on the grammatical and semantic functions of the particles. Although Group 1 received more information on the verbal conjugation errors (i.e., regarding whether it is perfective or imperfective and negative or affirmative), the difference between the two groups in the numbers of verbal predicate errors was very small. A few reasons are considered. First, the students in the study were already familiar with the verbal conjugations used in the **Nihongo-CALI** exercises, so the traditional feedback such as "The form of the predicate you typed is wrong. Change the form," may have sufficed. Another possibility is that since the target conjugations are quite simple, this would be an area in which detailed grammatical feedback for errors may not be necessary. The question of whether and ' when detailed feedback on verbal conjugations is useful should be investigated further. Regarding vocabulary errors, both Group I and Group 2 received the same feedback (i.e., what word is missing and what word is not expected to be used), but Group 2 committed more vocabulary errors than Group 1. One possible account for this difference is that since the intelligent feedback explained the grammatical and semantic roles of particles in relation to the noun phrases attached by those particles. Such feedback may have drawn the learner's attention to the meaning of each noun phrase and facilitated vocabulary memorization.

When the posttest scores were compared to the pretest scores, both Group 1 and Group 2 obtained significantly better scores on the posttest ($t = 7.91$, $p < 0.001$ for Group 1, and $t = 3.24$, $p < 0.02$ for Group 2). This suggests that students could benefit from the **Nihongo-CALI** programs (regardless of whether the intelligent or advance-version of traditional feedback was provided) to learn the Japanese particles and sentence constructions. However, since the study did not include a control group (a group receiving no computer instruction), this conclusion must await a future study. Table 4 compares which group gained more on the posttest.

The t score in Table 4 shows that the difference between the two groups in their gain scores is statistically significant at the .02 level, favoring Group 1. The result suggests that the intelligent CALI feedback is more effective than even the

	Mean	SD	t	Sig of t
Group 1 (n=9)	26.3	10.0	3.26	0.02
Group 2 (n=9)	14.7	13.6		

Table 4: The gain scores from the pretest to the posttest (scores out of 100)

enhanced version of traditional CALI feedback, underscoring the importance of automated language processing technology in second language instruction.

As noted, the students were paired based on the scores on the midterm exam and were randomly divided into either Group 1 or Group 2. For example, the student who obtained the highest score and the student who obtained the second highest score on the midterm exam were paired as Pair I and were randomly assigned to either Group 1 or Group 2. The same paring system was applied until the last two students (the students of the two lowest scores) were paired. Figure2 illustrates the individual students' posttest scores for each pair. It appears that the difference between the two students (one in Group 1 and one in Group 2) in each pair becomes larger toward Pair 9. The following factor may account for this phenomenon. In each computer session, the students read the grammar notes and moved to the exercises. The grammar notes provided detailed descriptions on the Japanese particles and sentence constructions. It may be the case that the students whose performance levels were higher were able to learn the target structures from the grammar notes alone and did not have to depend on feedback messages as much as the low-performance students had to. On the other hand, even though detailed grammatical explanations were provided in the grammar notes, the students in the low performance level needed ongoing grammatical feedback to understand the use of the target structures. Whether there is any relationship between the effect of intelligent CALI feedback and the learners' performance level would be another interesting topic for future research.

Figure 2: The individual student's posttest score for each pair

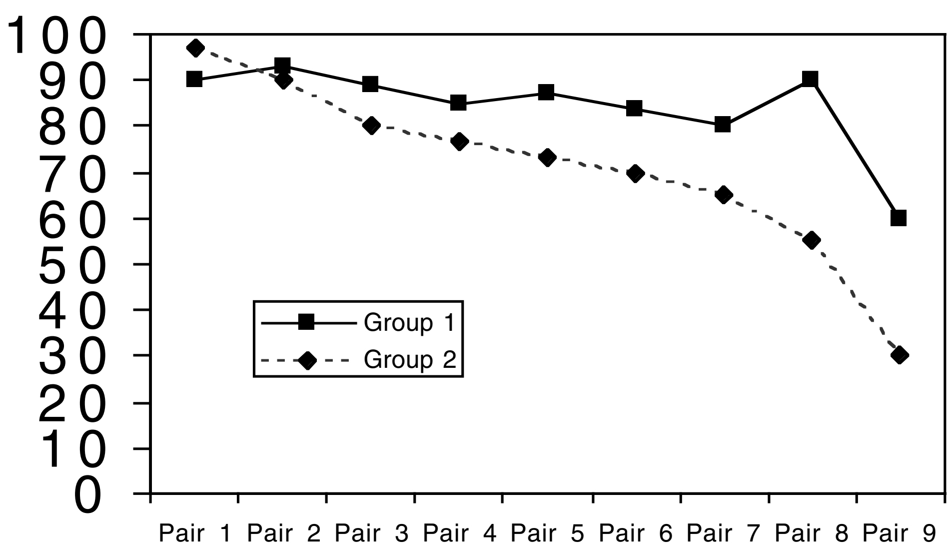

Table 5 presents the means and standard deviations of the ratings for each item on the questionnaire (1 strongly disagree, 2 disagree, 3 undecided, 4 agree, and 5 strongly agree). The result of the questionnaire shows that both groups had very positive attitudes toward the computer program, regardless of whether they received intelligent feedback or traditional feedback. This is consistent with the finding in the previous study. Regarding computer feedback (items 10 through 14), Group 2 gave a slightly higher rating than Group 1 on item 11, and Group 1 gave higher ratings than the Group 2 on items 10, 12, 13, and 14. Item 12 ("The error messages are helpful in pointing out why my response is wrong,") shows an especially big difference (although not statistically significant, i.e., 4.6 for Group 1 and 3.8 for Group 2, $p = 0.137$). This difference is reflected by the fact that Group 2 received traditional feedback which indicated only 'What is missing, unexpected, or wrong, while Group 1 received intelligent feedback which further explained why it was wrong.

The students were also asked to write their impressions of the computer program. All students in both Group 1 and Group 2 mentioned that they liked the computer program. There were a few comments on the computer feedback: a student in Group 1 wrote, "I enjoyed doing this computer program. I especially liked the feedback that the program gave you;" another student in Group 1 commented, 11 the program is clear and precise in the sense that it analyzes what the student writes instead of just giving the answer"; and a student in Group 2 wrote "the vocabulary hint and the correction were good." To the question, "What could be improved in this computer program?," one student in Group 2 (who received the traditional feedback) said that it should explain why one particle is used instead of another when the answer the student types is wrong. No students in Group 2 made such a suggestion. Seven students in Group 1 and 5 students in Group 2 wrote about slow processing time. Item 9 ("The processing of feedback is fast enough") on the questionnaire also received low ratings. **Nihongo-CALI** was originally developed on DEC workstations on which the average processing speed was 5 seconds for each student answer analyzed. However, the availability of DEC workstations was limited at the location of this study, so the system was implemented on Macintosh IIci's and these machines took an average 20 seconds to analyze each student's response. Increasing processing speed will be a future subject for the **Nihongo-CALI** research.

CONCLUSION

This study suggests that intelligent CALI feedback, which explains the grammatical and semantic functions of the Japanese particles, is significantly more effective in developing learners' sentence production skills than even the enhanced version of traditional feedback, which informs the students not only

Item	Group 1 (n=9)		Group 2 (n=9)	
	Mean	SD	Mean	SD
1. The instructions in the program are clear.	4.7	0.5	4.7	0.5
2. The program is very easy to use.	4.7	0.5	4.4	0.5
3. I didn't have technical problems when working on the program	4.1	1.1	4.6	0.5
4. The grammar notes are helpful.	4.8	0.4	4.9	0.3
5. The grammar notes are written clearly.	4.6	0.7	4.6	0.5
6. I went back to the grammar notes when doing CALI exercises	Yes (5) No (4)		Yes (7) No (2)	
7. The format of each CALI exercise is good.	4.4	0.7	4.3	0.5
8. The content of the CALI exercises is good.	4.4	0.5	4.3	0.5
9. The processing of feedback is fast enough.	2.9	1.8	2.7	1.3
10. The error messages are easy to understand.	4.2	0.7	3.7	1.1
11. The error messages are helpful in pointing out what is wrong with my response.	4.4	0.7	4.6	0.5
12. The error messages are helpful in pointing out why my response is wrong.	4.6	0.5	3.8	1.4
13. The error messages have the appropriate amount of detail.	4.6	0.5	4.2	0.7
14. The quality of the error messages is good.	4.6	0.5	4.3	0.7
15. I referred to the vocabulary hints during the CALI exercises. If yes:	Yes(8) No (1)		Yes(9) No(0)	
a. the vocabulary hints are helpful.	4.9	0.4	4.8	0.4
16. I can work on the CALI exercises at my own pace.	4.9	0.3	4.6	1.0
17. I did not get nervous while I was working on the CALI exercises.	4.7	0.7	4.7	0.5
18. The CALI exercises can help me practice Japanese by myself.	4.8	0.4	4.8	0.4
19. I find the CALI exercises interesting.	4.7	0.5	4.4	0.7
20. The CALI exercises are good as supplementary work outside of the classroom in learning Japanese.	4.6	0.5	4.4	0.7
21. I want to practice Japanese by using this type of CALI exercise on a regular basis.	4.9	0.3	4.1	1.4
22. The CALI exercises are good as supplementary work outside of the classroom in learning Japanese.	4.9	0.3	4.7	0.5
23. I learned a lot from the CALI exercises.	4.8	0.4	4.8	0.4

Table 5: The questionnaire on the computer program

of missing/unexpected particles but also of the positions of missing particles. The study demonstrates an effective application of natural language processing in second language instruction.

Traditional CALI systems using simple pattern-matching and error anticipation have some flexibility in providing different levels of feedback. However, they cannot identify the grammatical and semantic function of each component of the sentence, so it is hard to explain the nature of errors, especially if the errors occur at the sentence level. Furthermore, all possible correct answers and anticipated errors have to be coded directly. On the other hand, intelligent CALI systems using natural language processing can provide both traditional and intelligent types of feedback as seen in the **Nihongo-CALI** system. It is not necessary to install all possible correct answers because intelligent systems have grammatical rules and a lexicon for the target language, and if a sentence follows the grammatical rules and the lexical requirement, it can be parsed and grammatical information on the sentence can be generated. Error analysis can be performed by adding grammatical rules to parse ungrammatical sentences and to flag corresponding error messages. One difficulty in using Natural Language Processing is that it requires a large amount of time, cost, and knowledge to develop such a robust language analyzer. On the other hand, it is quite easy to develop traditional CALI programs using only simple pattern-matching and error anticipation, and those programs can still be very useful. Bonner (1987) proposes that traditional computer assisted instruction is a sound choice for teaching simple knowledge and skills, and that intelligent computer assisted instruction is a good choice for more complex kinds of learning. This indicates that the future direction of computer assisted language instruction is not replacing one type of system with another. Rather, we should utilize both systems, depending on which kinds of tasks we wish to assign to the students, which grammatical structures we intend to teach, and what types of feedback we intend to return. Traditional CALI systems may be suitable for vocabulary and conjugation exercises in which only word-level analysis is required and simple "wrong-try-again" feedback may suffice. Comprehension exercises with multiple-choice questions are also easily implemented into traditional CALI systems because the nature of the error is fixed by the question, and an appropriate error message can be stored for each incorrect choice. On the other hand, as seen in the present study, intelligent CALI systems are more suitable for exercises of sentence production, which require complex grammatical analysis and ongoing detailed explanations in response to the learners' errors.

There are some interesting questions for future research. For example, even though **Nihongo-CALI** does not involve speaking interaction, it might be interestingto include an oral test for both pretest and posttest to investigate how applicable learning from **Nihongo-CALI** exercises is for oral production. Second, the present study did not include a retention test. After the experimental sessions, the students continued to practice Japanese sentence production in regular classes and tutoring sessions both orally and in writing (because developing sentence production skills was a stated goal of the course). Consequently, the amount of practice and the quality of feedback provided to individual students were no longer controlled, and the study could not measure the long-term effect of the computer instruction alone. In the previous study, the target structures (e.g., passive constructions) were not included in the course syllabus and were practiced only once in the classroom after the experimental sessions; all students received an equal amount of oral practice using passive constructions, and when the students committed errors, they were provided only correct models. Therefore, the exposure to the target structures after the experiment -was controlled in the previous study to measure a long-term effect of the computer instruction. If we want to examine the long-term effect of the treatment alone, any extra variables need to be controlled carefully after the treatment.

A future study of **Nihongo-CALI** will include more varieties of grammatical structures for production exercises and will investigate the effects of different types of feedback with respect to different levels of complexity of production exercises and different performance levels of students. This -will help to establish guidelines for the appropriate application of natural language processing technology in second language instruction.

NOTES

1 A shorter version of this paper was presented at the annual conference of the American Council on the Teaching of Foreign Languages (ACTFL) in Anaheim, California in November 1995. I thank Dr. Kevin Kelly at Carnegie Mellon University for his valuable comments on an earlier draft.

2 The author developed **Nihongo-CALI** 'in the cooperation with the ALICE project team in the Laboratory for Computational Linguistics at Carnegie Mellon University.

3 The romanization used in this paper is an adaptation of the Shin-kunrei-shiki "New official system" Gorden 1987).

4 Explicit grammatical instruction has been supported by many studies. See Ellis (1990) and Long (1983, 1988) for reviews. Also see recent studies on the effectiveness of metalinguistic feedback by Carroll and Swain (1993) and by Nagata (1996).

5 The cT language is a programming language and environment under development at Carnegie Mellon University (Sherwood 1988).

6 At present, there are 13 lessons of sentence production exercises in **Nihongo-CALI**, of which five lessons are designed to be used with Jorden's textbook, *Japanese: the Spoken Language Part 1*. Four lessons provide practice on basic Japanese sentences using the particles *ga, wa, o, ni* and *de*, and another four lessons provide practice on Japanese passive sentences. These lessons have been used at the University of San Francisco on a regular basis.

[7] The following alternative versions are also accepted. For example, the word order of "*Nihonmati*" and "*osusi*" can be reversed, "*susi*" can be "*osusi*"' (a polite version of "*susi*"), and the sentence particle "*yo*" (an assertion marker) can be added at the end of the sentence. The **Nihongo-CALI** program also accepts different romanizations such as "*shi*" for "*si*," "*sha*"" for "*sya*," "*tsu*" for "*tu*," "*chi*" for "*ti*," etc.

[8] The following scoring system was used for the pretest: 1.0 score was deducted for an incorrect or missing particle, 1.0 score for an incorrect or missing verb (although when the error was on the verbal conjugation such as using imperfective for perfective or negative for affirmative, only 0.5 score was deducted), and 0.5 score was deducted for an incorrect or missing word (other than incorrect/missing particles and verbs). Incorrect word order such as placing a noun phrase before a verbal predicate also resulted in a 0.5 score reduction.

[9] I thank Dr. William Garner and Michael Benedict for their cooperation and valuable assistance in the use of the CIT laboratory.

[10] *Ga* indicates the subject who performs the action, *o* the object that the action operates upon, and *wa* the topic and/or contrast of the sentence.

[11] In the double *ga* construction, the particle *ga* is used to indicate both subject and object of stative predicates.

[12] *Ni* indicates the location where the non-activity referent is located and *de* the location where the activity takes place.

[13] *Ni* indicates the direction which the action moves toward or the goal of the action, *de* the instrument by means of which the action occurs, and *o* the location through which the action moves.

[14] The posttest followed the same scoring system used in the pretest.

REFERENCES

Bonner, Jodi (1987). "Computer Courseware: Frame-Based or Intelligent?" *Educational Technology* 27, 3, 30-35.

Carroll, S., and M. Swain (1993). "Explicit and Implicit Negative Feedback: An Empirical Study of the Learning of Linguistic Generalizations." *Studies in Second Language Acquisition* 15, 357-386.

Ellis, Rod (1990). *Instructed Second Language Acquisition*. Cambridge, Massachusetts: Blackwell Publishers.

Hausser, Roland (1989). "Principles of Computational Morphology." Unpublished manuscript. Carnegie Mellon University, Laboratory for Computational Linguistics.

Hart, Robert S. (1981). "Language Study and the Plato System." *Studies in Language Learning* 3, 1-24.

Holland, Melissa, Richard Maisano, Cathie Alderks, and Jeffery Martin (1993). "Parsers in Tutors: What Are They Good For?" *CALICO Journal* 11, 1, 28-46.

Juozulynas, Vilius (1994). "Errors in the Compositions of Second-Year German Students: An Empirical Study for Parse-Based ICALI." *CALICO Journal* 12, 1, 5-17.

Kuno, Susumu (1973). *The Structure of the Japanese Language*. Cambridge, MA: MIT Press.

Labrie, Gilles, and L.P.S. Singh (1991). "Parsing, Error Diagnostics, and Instruction in a French Tutor." *CALICO Journal* 9, 1, 9-25.

Levin, Lori, David A. Evans, and Donna Gates (1991). "The ALICE System: A Workbench for Learning and Using Language." *CALICO Journal* 9, 1, 27-56.

Liou, Hsien-Chin (1991). "Development of an English Grammar Checker: A Progress Report." *CALICOJournal* 9, 1, 57-70.
Long, H. Michael (1988). "Instructed Interlanguage Development." *Issues in Second Language Acquisition: Multiple Perspectives*, edited by L. Beebe. Rowley, MA: Newbury House.
______. (1983). "Does Second Language Instruction Make a Difference?" *TESOL Quarterly* 17, 3, 359-382.
Loritz, Donald (1992). "Generalized Transition Network Parsing for Language Study: the GPARS System for English, Russian, Japanese, and Chinese." *CALICO Journal* 10, 1, 5-22.
Nagata, Noriko (1993). "Intelligent Computer Feedback for Second Language Instruction." *The Modern Language Journal* 77, 3, 330-339.
______. (1992). *A Study of the Effectiveness of Intelligent CALI as an Application of Natural Language Processing*. Dissertation, University of Pittsburgh.
______., andM. Virginia Swisher (1995). "A Study of Consciousness-Raising by Computer: The Effect of Metalinguistic Feedback on Second Language Learning." *Foreign Language Annals* 28, 3, 337-347.
______. (1996). "The Effectiveness of Computer-Assisted Metalinguistic Instruction: A Case Study in Japanese." Forthcoming in *Foreign Language Annals*.
Sanders, Ruth (1991). "Error Analysis in Purely Syntactic Parsing of Free Input: The Example of German." *CALICO Journal* 9, 1, 72-89.
Si-Quing, Chen, and Xu Luomai (1990). "Grammar-Debugger: A Parser for Chinese EFL Learners." *CALICO Journal* 8, 2, 63-75.
Tomita, Masaru, T. Mitamura, H. Musha, and M. Kee (1988). "The Generalized LR Parser/Compiler Version 8.1: User's Guide." CMU-CMT-88-MEMO. Carnegie Mellon University, Center for Machine Translation.
Yazdani, Masoud (1991). "The LINGER Project: An Artificial Intelligence Approach to Second-Language Tutoring." *Computer Assisted Language Learning* 4, 2, 107-116.

AUTHOR'S BIODATA

Noriko Nagata (Ph.D., University of Pittsburgh) is an assistant professor and a director of the Japanese program at the University of San Francisco. Her research involves the development, implementation, and assessment of intelligent CALI in applications to second language instruction.

AUTHOR'S ADDRESS

Department of Modern and Classical Languages
University of San Francisco
2130 Fulton Street
San Francisco, CA 94117
Phone: 415-666-6227
Fax: 415-666-6928
E-mail: nagatan@usfca.edu