Salinlahi III: An Intelligent Tutoring System for Filipino Language Learning

Ralph Vincent Regalado , Michael Louie Boñon, Nadine Chua, Rene Rose Piñera,
Shannen Rose Dela Cruz
Center for Language Technologies
De La Salle University, Manila
ralph.regalado@delasalle.ph,
{michael.bonon, chuanadine}@gmail.com,
{renepinera, shannenrose.delacruz}@yahoo.com

Abstract

Heritage language learners are learners of the primary language of their parents which they might have been exposed to but have not learned it as a language they can fluently use to communicate with other people. Salinlahi, an Interactive Learning Environment, was developed to teach these young Filipino heritage learners about basic Filipino vocabulary while Salinlahi II included a support for collaborative learning. With the aim of teaching learners with basic knowledge in Filipino we developed Salinlahi III to teach higher level lessons focusing on Filipino grammar and sentence construction. An internal evaluation of the system has shown that the user interface and feedback of the tutor was appropriate. Moreover, in an external evaluation of the system, experimental and controlled field tests were done and results showed that there is a positive learning gain after using the system.

1   Introduction

The idea behind Intelligent Tutoring Systems is letting a computer simulate a sophisticated human tutor and be able to use a teaching strategy appropriate for each student (Murray, 1999) (Massey, Psotka, & Mutter, 1988). According to (Kassim, Kazi, & Ranganath, 2004), there are four important elements that must be present in instructional systems. These four are the tutor, the student, the domain knowledge to be learned by the student, and lastly, the computer itself. These four elements were the basis of the functional model of an ITS according to (Kassim, Kazi, & Ranganath, 2004) which consists of the following: expert module, tutoring module, student model and user interface module. This functional model is accepted as a standard in building the system architecture of ITSs (Polson & Richardson, 1988). Many ITSs have been built for different fields (such as cardiovascular physiology, algebra, language learning, electronics troubleshooting etc). Some of these ITSs are AutoTutor and ELM-ART. AutoTutor, developed by (Graesser et al., 2004), features tutoring done in natural language dialogue while ELM-ART, developed by (Brusilovsky, Schwarz, & Weber, 1996), pioneered ITSs in the World Wide Web.

Intelligent Tutoring Systems served as the inspiration in developing Salinlahi III. Unlike the two previous iterations of Salinlahi which are purely Interactive Learning Environments, Salinlahi III was designed from the ITS perspective. In the previous systems, students have the freedom to choose which lessons to take while in Salinlahi III, students are directed toward a sequential progress of lessons by a tutor which is the system itself. The lessons in this system are structured in terms of content and skill complexity, starting from basic concepts to lessons dealing with construction of basic Filipino sentences. The tutor decides when to advance to the next lesson based on the student's performance in the exercises given in every lesson. These exercises are namely Translation Exercise and Creation Exercise. The exercises were designed as a dialogue type of exercise between the tutor and the student. However, it is only in the Translation Exercise where the tutor will guide the student to attain the correct answer by giving feedback on his or her current answers. The feedback given by the tutor is done in natural language inspired by AutoTutor. This is made possible through the use of template-based Natural Language Generation (NLG), a technique used in generating content in natural language.
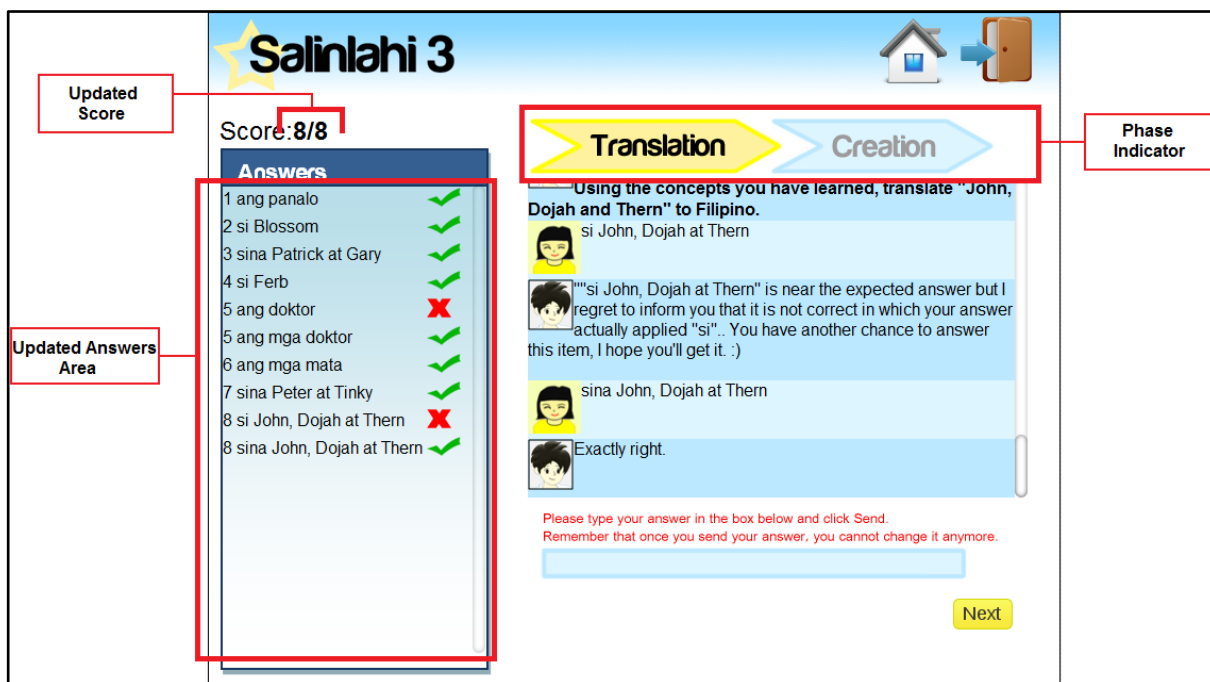
Figure 1. Screenshot of the Translation Exercise

Salinlahi III focuses on teaching basic Filipino grammar whereas the two previous iterations focus on Filipino vocabulary. This change in the domain of knowledge is a solution to accommodate learning needs of older students with a basic knowledge of Filipino vocabulary. These students are assumed to be at the range of fourteen (14) to seventeen (17) years old.
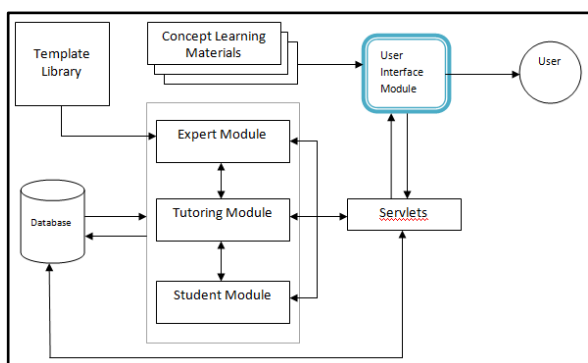
2 System Architecture

Figure 2. Salinlahi III System Architecture

The system architecture of Salinlahi III follows the functional model of an ITS by (Kassim, Kazi, & Ranganath, 2004). Based from this model, Salinlahi III has been designed with four main modules namely, User Interface Module, Expert Module, Tutoring Module, and Student Module.

2.1 User Interface Module

The User Interface Module provides lesson and exercises to the user. It is also responsible for passing the user input to the different system modules.

2.2 Expert Module

The Expert Module does the analysis of the user's input. This module contains two different analyzers for the exercises. The first input analyzer is for the Translation Exercise. This analyzer checks the translation of the student with the corresponding expected translation by the tutor. After analysis, the tutor will come up with its assessment. The second analyzer is for the Creation Exercise. This analyzer's main job is to parse the student's input and check if the student has successfully applied in his or her answer what is being discussed in the current lesson. This module also does different functions such as the computation of the Levenshtein distance, tokenization of strings and the Realiser which prepares the tutor's feedback.

2.3 Tutoring Module

The Tutoring Module handles the processes of the exercises. For the Translation Exercise, the tutor's move is managed. This move, as discussed earlier in Section 2.3, is based from the assessment of the tutor on the student's answer. The assessment is produced in the Expert Module. This module includes (but is not limited to)

keeping track of the chances of the student, managing the number of items given, updating the student model, managing the turns between the student and the tutor, communicating with the Expert Module for analysis, management of the student model, and the production of evaluation after the exercises.

2.4 Student Module

The Student Module contains student model. It is in this module that the tree for the student model is handled. This module is accessed by the Tutoring Module to update the student model based on the student's performance in the exercises. It is also accessed when the evaluation for the student is needed. The evaluation is generated based from the data in the nodes of the student model.

3 System Exercises

In every lesson, students are given exercises to test their learning on the current lesson they are studying which are Translation Exercise and Creation Exercise. This phase where students answer exercises corresponds to the "Response" stage in the Initial- Response-Evaluation (IRE) model mentioned by (Murray, 1999). According to the IRE model, the Response stage is where students are expected to have gained knowledge or skills after being exposed to instructions. Furthermore, in this stage, students are expected to have learned how to translate new knowledge or skills into practice (Murray, 1999).

3.1 Translation Exercise

Students go through Translation Exercise first before the Creation Exercise. In Translation Exercise, the learner is given a certain number of phrases or sentences he or she needs to translate to Filipino in the context of the lesson where it belongs. The exercise begins with the tutor giving instructions to learners followed by giving the first problem. For each problem, the learner is given at most three chances to answer correctly. In case the learner does not get it right at the last chance, the score will be based on his or her last answer. As the learner tries to answer the problem, the tutor in turn gives the appropriate feedback in order to facilitate learning as seen in Figure 1.

After the student enters his or her answer, the tutor immediately reflects and gives feedback based from the student's answer. It can be seen in Figure 3 that the tutor immediately tries to make its move after the student's input. The moves of the tutor are not random. It makes its move based from its assessment on the student's answer. Figure 3 shows the flow on how the tutor's feedback is formed.

Figure 3. Flow on how the tutor comes up with a feedback

3.1.1 The Assessment of Student's Answer

$$Difference\ Value = \frac{LevenshteinDistance}{LengthOfExpectedAnswer}$$

Equation 1. Difference Value

In the Translation Exercise, there are four possible assessments for the student's answer. These are "Correct", "Near the Answer", "Incorrect" and "Cannot be understood". "Correct" is the assessment that the tutor would come up with when the student gave an answer that exactly matches the expected answer of the tutor. "Near the Answer" is assessed by the tutor if it sees that the answer is wrong but have analyzed that the student possibly applied a misconception in his or her answer. "Near the Answer" could also be assessed by the tutor if the computed difference value of the student's answer against the expected answer is less than or equal to .30. The formula of the difference value is shown on Equation 1. In Equation 1, "LengthOfExpectedAnswer" corresponds to the number of characters the string, which is the student's answer, has. The "LevenshteinDistance" corresponds to the Levenshtein distance. It is a string metric used to measure how many insertions, substitutions and deletions are required to transform a given string to a target string (Nielsen, 1994).

"Incorrect" on the other hand is assessed by the tutor if the answer is also wrong. However, it only becomes the assessment when the student applied a misconception and the computed difference value exceeds the threshold which is .30. "Cannot be understood" is the assessment given by the tutor when it sees that the student's answer is wrong, no misconception was applied and the computed difference value exceeds the threshold.

3.1.2 The Move of the Tutor

The move that the tutor should make is based from the assessment of the student's answer. The four moves of the tutor are *Explain*, *Warn*, *State* and *Produce*. All of these moves have different types of feedback. *Explain* will be the move used

when the learner's input is "Incorrect" or "Near the Answer." *Warn* will be the move used when the learner's input is "Cannot be Understood". This would tell the user to enter a more sensible input. *State* will be the move used when the learner's input is "Correct". *Produce* will be the move to be used to give an exercise to the learner.

3.1.3 The Feedback of the Tutor

The feedback of the tutor is based from the move that the tutor has to make. The feedbacks are produced using template-based NLG. Under each of the four moves are different templates for the feedback of the tutor. The type of feedback to be given by the tutor also takes into account the number of chances left in the current item. An example of a feedback can be seen on Figure 1. In that scenario (see Figure 1) the tutor used an *Explain* move. The type of feedback under this move to be used by the tutor would be based from the assessment and the number of chances left. The tutor's feedback which is '*"si John, Dojah at Thern" is near the answer but I regret to inform you that it is not the correct answer in which your answer actually applied "si". You have another chance to answer this item, I hope you'll get it. :)* '. Notice in this scenario, the tutor informed the student his or her answer is wrong but is near to the expected answer. The tutor also tried to show why the answer was wrong by pointing out the misconception applied by the student. Also, since it is the student's first time to get a wrong answer in that item, the last part of the message indicates that the tutor tries to give the student a positive feedback to boost up his or her confidence.

3.2 Creation Exercise

In the Creation Exercise, the student is asked by the tutor to construct or create his or her own Filipino phrases or sentences. The phrases or sentences that need to be created would be based from the lessons or sub lessons discussed in the concept learning materials. The tutor would randomly pick a sub lesson and ask the user. Unlike the Translation Exercise, in the Creation Exercise, the tutor does not give immediate feedback on the student's input. The student would only see an overall evaluation after the exercise.

3.2.1 Reflection in Action

The Translation Exercise was designed based on "reflection in action" by (Schön, 1987). According to (Schön, 1987) "reflection in action" is a case where people "reflect in the midst of action". It shows here that it is during the task that the person reflects on what he or she is doing. In accordance to this, the Translation Exercise was designed to help the student to reflect in his or her performance while taking the exercise. The tutor gives feedback to guide the student and make him or her rethink about his or her answer while there is still a chance. There is an immediate feedback whose goal is to let the student immediately reflect and react.

3.2.2 Reflection on Action

The Creation Exercise was designed with basis on "reflection on action" by (Schön, 1987). (Schön, 1987) stated that "We may reflect on action, thinking back on what we have done in order to discover how our knowing-in-action may have contributed to an unexpected outcome." The Creation Exercise is designed to be taken after the Translation Exercise. This is to allow the student to think on his or her previous actions previously committed during the Translation Exercise. In the Creation Exercise, the student can think back on his or her actions and reflect on it based from the feedback of the tutor at that time. Compared to the Translation Exercise, the tutor during the Creation Exercise does not give an immediate feedback to the student's input. This is to promote self-reflection on the student's part and have a recall about the tutor's feedback earlier (in the Translation Exercise). Whatever phrase or sentence the student has created, this could most likely show how much he or she really knows, without the help of the tutor.

4 System Evaluation

Tests were done to determine whether the system was able to achieve its main objective, which is to teach Filipino grammar to secondary level heritage language learners. Experts evaluated the system to determine whether the feedback and user interface is appropriate for the target users. Experts also evaluated the system to determine whether it can be considered as a potential educational software. Students also tested the first lesson of the system to know whether there was an increase or decrease in their learning. The increase or decrease in learning may also determine the potential of Salinlahi III as an educational software.

4.1   Internal Evaluation

The internal evaluation focused on seeing the system as a potential educational software. The experts who evaluated the system came from different fields – User Interface (UI), development of teaching tools, second/foreign language teaching (experts from this field teaches languages not only Filipino), education, Filipino teaching. The experts evaluated the system based on UI design, feedback, and as a potential educational software

Two experts evaluated the system based on the UI design. They did a test over the system and were later given a questionnaire. The questionnaire is based from the general principles of interface design that was developed by (Nielsen, 1994). The experts gave an average grade of 4.17, which is between excellent and good. Since they evaluated based on general principles in designing the user interface, this means that it is acceptable for all types of target users. However, these experts gave the following recommendations: (1) it would be helpful if there will be a status that would show that the page is loading or that the tutor is typing a message. (2) Instructions should be provided for first time users. (3) The number of chances during the exercise should be made known to the users. In addition, (4) the system should provide a way to retrieve the password once the user forgets it.

An interview was conducted with an expert who evaluated the system based on the tutor's feedback. The results of the interview with the expert show that the tutor's feedback is appropriate for the learners. The expert verified that the messages are gender-neutral and appropriate for the age range of the target users. In addition, the approaches of the tutor are acceptable and overall the tutor sounds human. However, the expert stated that if the answer falls for the assessment "Cannot be Understood", the tutor must inform the student to refer back to the lesson (e.g. "*Your answer is not within the framework of the discussion. Please refer to the review lessons*").

Three experts evaluated the system as a potential educational software. They were also given a questionnaire after testing the system. These experts gave an over-all average rating of 4.3, which is between excellent and good. With this grade and the average grade under all the criteria given, the system is believed to have passed or has a potential to be an educational software. However, few recommendations was made: (1) the numbers given in the exercises should be in

Filipino and not in Spanish, (2) the learning materials should be in Filipino (e.g. using "*saging*" instead of "*banana*"), and (3) add more Filipino words to the system. One of the system's limitation is having a small collection of words. Because of this, the system may have a wrong assessment of the learner's answer only because the Filipino word used by the learner is not found in the system's collection of words.

4.2   External Evaluation

The external evaluation was done in order to assess how the system affected the learner specifically in determining its performance in teaching students. During the said evaluation, a total of twenty-seven students (27), six coming from the De La Salle University and twenty-one (21) came from MIT International School, went through a series of tests illustrated in Figure 2. Students who have not finished the series of tests were excluded from the analysis.

Figure 2. Flow of External Evaluation

Students were grouped into two: experimental and controlled end-user testing. Students who went through experimental end-user testing were allowed to interact with the system and use it on their own while those who went through controlled end-user testing were supervised. Results from both groups were compared to observe whether supervising the students while they use the system affects their scores in the post-test. This was important since previous iterations of Salinlahi did controlled end-user test only. The addition of the uncontrolled group gives a more realistic scenario where most likely there would be no one to supervise the users aside from the system itself and some accompanying documents.

In pre-test, post-test, and vocabulary test, students were asked to translate English words and phrases into Filipino. The scores in pre-tests and post-tests were used to determine whether students' performance increased after using the system, or in other terms, they had a positive learning gain. Thereby learning gains can be measured using the paired t-test. According to (Shier, 2004), "a paired t-test is used to compare two population means where you have two samples in which observations in one sample can be paired with observations in the other sample".

Table 1 shows the scores of the students for the pre-test and post-test under the experimental group. As seen in Table 1, scores of twelve (12 – approximately 67%) students increased, five of them did not show any increase at all and the score of one student decreased. Using paired t-test on these values, results have shown that there is a significant increase in the scores as determined by the value of p which is 0.003. The values obtained in the paired t-test are shown in Table 2.

Student No.	Pre-Test Score (X)	Post-Test Score (Y)	Difference (Y-X)
1	0	4	4
2	1	4	3
3	5	7	2
4	4	8	4
5	3	6	3
6	2	4	2
7	2	2	0
8	6	8	2
9	0	0	0
10	2	4	2
11	0	2	2
12	3	7	4
13	1	2	1
14	2	2	0
15	2	2	0
16	4	3	-1
17	0	2	2
18	0	0	0

Table 1. Students' pre-test and post-test scores for experimental end-user testing

Label	Value (X)
n	18
Mean Difference	1.666667
Standard Deviation	1.57181
Standard Error of the Mean	0.370479287
t	4.498677054
Degrees of Freedom	17
p-value	0.003

Table 2. Paired t-test values for experimental end-user testing

On the other hand, Table 3 shows the scores of the students for the pre-test and post-test under the controlled group. As seen in Table 3, scores of twelve (6 – approximately 67%) students increased, two of them did not show any increase at all and the score of one student decreased. Using paired t-test on these values, results have shown that there is a significant increase in the scores as determined by the value of p which is 0.0325. The values obtained in the paired t-test done for this group are shown in Table 4.

Student No.	Pre-Test Score (X)	Post-Test Score (Y)	Difference (Y-X)
1	0	2	2
2	0	4	4
3	6	7	1
4	6	8	2
5	0	5	5
6	6	8	2
7	7	6	-1
8	8	8	0
9	7	7	0

Table 3. Students' pre-test and post-test scores for controlled end-user testing

Label	Value (X)
n	9
Mean Difference	1.666666667
Standard Deviation	1.936491673
Standard Error of the Mean	0.645497224
t	2.581988897
Degrees of Freedom	8
p-value	0.0325

Table 4. Paired t-test values for controlled end-user testing

5 Conclusions and Recommendations

Salinlahi III is the third iteration in a series of systems that were developed to teach the Filipino language to heritage language learners. These learners are those who grew up oversees, but did not learn Filipino formally, but may be exposed to it through their Filipino parent(s).

Based on the internal and external evaluations, Salinlahi III has a potential to be an educational software. It got a high average score on the expert evaluations based on UI design, feedback, and as an educational software. The students who used the system got a positive learning gain, which means that the system was able to achieve its objective. However, since these students are residing in the Philippines, different results may arise once the system is tested on its actual target users.

Salinlahi III's future works may include having a conversational tutor that would be able to understand other user's input. This can be done by adding Natural Language Understanding techniques and tools to the current system. Another would be to include sound into the system. Currently, the system does not support voice recognition as it is also not able to produce the tutor's response through sound. Having the system support voice recognition will aid in making the tutor more conversational especially during the exercises.

References

Brusilovsky, P., Schwarz, E., &Weber, G. (1996). Elm-art: An intelligent tutoring system on world wide web. In (pp. 261–269). Springer Verlag.

Graesser, A., Lu, S., Jackson, G., Mitchell, H., Ventura, M., Olney, A., et al. (2004). Autotutor: A tutor with dialogue in natural language. Behavior Research Methods, 36, 180-192. Available from http://dx.doi.org/10.3758/BF03195563 (10.3758/BF03195563)

Kassim, A. A., Kazi, S. A., & Ranganath, S. (2004). A web-based intelligent learning environment for digital systems. International Journal of Engineering Education, 20, 13-23.

Lê, Q., & Lê, T. (2007) Evaluation of educational software: theory into practice. In: Technology and Teaching. Nova Science Publishers, New York, UK. ISBN 978-1-60021-699-2

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. , 10 . Available from http://sascha.geekheim.de/wp-content/uploads/2006/04/levenshtein.pdfnumber=journal=Soviet Physics Doklady, publisher=American Institute of Physics., pages=707--710

Massey, L. D., Psotka, J., & Mutter, S. A. (1988). Intelligent tutoring systems : lessons learned / edited by joseph psotka, l. dan massey, sharon a. mutter, advisory editor, john seely brown [Book]. L. Erlbaum Associates, Hillsdale, N.J. :.

Murray, T. (1999). Authoring Intelligent Tutoring Systems: An Analysis of the State of the Art. International Journal of Artificial Intelligence in Education, 10, 98-129.

Nielsen, J. "Heuristic Evaluation". In J. Nielsen and R.L. Mack, editors, Usability Inspection Methods, John Wiley, 1994.

Polson, M. C., & Richardson, J. J. (Eds.). (1988). Foundations of intelligent tutoring systems. Hillsdale, NJ, USA: L. Erlbaum Associates Inc.

Regalado, R. V. , Antay, M. R., Fernandez, L., Cheng, C., & Jumarang, L. (2010). Building web-based Filipino language learning tool for heritage learners. Philippine Information Technology Journal, Vol. 2, Issue No. 2, pp 54-57.

Shier, R. (2004). Statistics: Paired t-tests. Retrieved from http://mlsc.lboro.ac.uk/resources/statistics/Pairedttest.pdf

Schön, D. A. (1987). Educating the reflective practitioner (Vol. 58) (No. 4). Jossey-Bass.