A heuristic method for performance evaluation of learning in an intelligent tutoring system

KIRAN MISHRA\(^1\)

R.B. MISHRA\(^2\)

Banaras Hindu University
Department of Computer Engineering
Institute of Technology
Varanasi-221005- India
\(^1\)kiran.mishra.bhu@gmail.com
\(^2\)ravibm@bhu.ac.in

Abstract. Evaluating student’s performance is one of the important activities in intelligent tutoring system. This paper presents learning strategy for study and a heuristic method for performance evaluation of learning of student in intelligent tutoring system. A course has been divided into different interlinked courseware. The method is based on the number of attempts with and without hint for success in a courseware and student’s performance in a courseware. Grading such as average, good, excellent has been assigned to the student based on his performance value index. A comparative view of different methods has been presented based on some characteristics such as: computing model, evaluation parameter and evaluation measure. A graphical view of comparison has also been presented.

Keywords: Intelligent tutoring system, performance evaluation, hint, attempt, performance index.

(Received November 17, 2009 / Accepted May 07, 2010)

1 Introduction

Many intelligent tutoring systems (ITS) have been developed for teaching of programming languages i.e. MENO-II [5], PROUST[9], LISP-Tutor [3], ELM-PE[17], C++ tutor [1][10], JAVA tutor [15], [2]. These systems use different learning strategy. PROUST attempted to estimate student’s plan intention while LISP tutor guided student’s left to right, top down attempt by interpreting their code as a correct or buggy solution.

MENO-II and ELM-PE tutor analyze the student’s solution to exercise and provide feedback to identify misconception or missing skills based on the analysis. [6] presented an expert tutoring system (E-TCL) for teaching computer programming languages through WWW. Their system consists of three agents representing server-client relationship, tutoring agent(TA) as a "server", personal assistant agent for teachers(PAA-T), and personal assistant agent for students (PAA-S) as "clients". The PAA-S can communicate with the TA through the WWW to retrieve the tutoring dialog of the command(s) that a student wants to practice, and to access the experiences of other students in the blackboard module while the PAA-T communicates with the TA to add/modify semantic rules of computer programming languages and to check the correctness of the contents of the blackboard database.

[4] presented an ITS (Bits) for computer programming using Bayesian technology. Bits provided remote access to hypermedia-structured learning material which included instruction notes, tests, and examples. Unlike traditional web based education tools, Bits provided the learner with intelligent navigation support, recommendation, and integrated the features of an electronic hypermedia textbook with intelligent tutoring tactics. Bits proposed learning goals and guided users by generating reading sequences for them. [10] asks students to pre-
dict C++ program’s output and identify semantic and runtime errors. [15] intelligently examines the student’s submitted code and determines appropriate feedback based on a number of factors such as cognitive model of the student, the student’s skill level, and problem details.

[13] described a tutoring system for learning computer programming based on a multiple domain multiple-agent environment. It provided a multiple domain learning environment for language learners. The aim of the system was to teach the target domain by using a supporting domain(s) to reinforce the learning of it. [2] developed an agent based intelligent tutoring system for parameter passing in Java programming. This system helps student better understanding of parameter passing mechanisms in Java using problem based technique. [1] provides intelligent feedback to the student and for this purpose it relies on a group of information: the problem statement, problem specification, student’s code, established student’s model, the C++ compilation and the result from C++ run time engine.

Different methods have been used for performance evaluation of student. [18] evaluated performance of student in four levels i.e. global assessment(an overall measurement of student’s ability); procedure-level assessment(measurement for each problem the student is asked to solved); stage assessment(a measurement for each of three physiological stages in a problem); local assessment(a measurement for each variable that has been tutored). [16] presented an approach to hierarchical knowledge representation for the student’s evaluation in propositional logic. The hierarchical evaluation consists in assessing the student’s state of knowledge at several level of granularity.

[8] evaluated student’s performance using evaluation module. The evaluation module evaluates the user’s performance during a learning session, based on the interaction of the user with the system. Student’s mark level ranges from ‘low’ to ‘excellent based on the number of times the user asked for assistance, the number of related example requested by the user and the number of answering attempts made by the user. This paper presents learning strategy for study and heuristic method for performance evaluation of learning of student in intelligent tutoring system. The method is based on the number of attempts for success in a courseware and student’s performance in a courseware.

Rest of the paper has been organised as follows: Apart from introduction in section 1, section 2 deals with modeling the courseware. Section 3 represents learning strategy. Performance evaluation has been represented in section 4. Section 5 contains experimentation. section 6 contains result, section 7 represents comparison and finally section 8 represents conclusion.

2 Modeling the courseware
We have been taken C++ programming language as subject model. The course has been divided for the whole curriculum for C++ into seven coursewares i.e. CW1-classes and objects, CW2-constructor, CW3-operator overloading, CW4-inheritance, CW5-virtual functions, CW6-managing console io operations, CW7-working with files. The curriculum has been divided into courseware in such a way that CW1 is prerequisite for CW2; CW1 and CW2 are prerequisite for CW3; CW1,CW2 and CW3 are prerequisite for CW4; CW1,CW2,CW3 and CW4 are prerequisite for CW5; CW1, CW2, CW3, CW4 and CW5 are prerequisite for CW6; and CW1, CW2, CW3,CW4,CW5, and CW6 are prerequisite for CW7. For example, classes and objects are necessary to understand constructor and classes and objects and constructor are necessary to understand operator overloading.

3 Learning strategy:
The flow chart for learning strategy has been shown in figure 1 which includes the following steps.

1. Student studies first courseware CW[1]
2. After study his performance is evaluated.
3. If his performance is up to level he can go to CW[2]. If performance is not up to level he has to repeat same courseware (he will get up to three attempts. In 2nd and 3rd attempt he will get hint related to question. If performance is up to level, he will go to next courseware. If performance is not up to level he will repeat same courseware. After three attempts if performance is not up to level he will get specific treatment)
5. If his performance is up to level he can go to CW[3]. If performance is not up to level he has to repeat same courseware (he will get up to three attempts. In 2nd and 3rd attempts he will get hint related to question. If performance is up to level, he will go to next courseware. If performance is not up to level he will repeat same courseware. After three attempts if performance is not up to level he will get specific treatment)
7. If his performance is up to level he can go to CW[4]. If performance is not up to level he has to repeat same courseware (he will get up to three attempts. In 2nd and 3rd attempt he will get hint related to question. If performance is up to level, he will go to next courseware. If performance is not up to level he will repeat same courseware. After three attempts if performance is not up to level he will get specific treatment)


9. If his performance is up to level he can go to CW[5]. If performance is not up to level he has to repeat same courseware (he will get up to three attempts.


11. If his performance is up to level he can go to CW[6]. If performance is not up to level he has to repeat same courseware (he will get up to three attempts. In 2nd and 3rd attempt he will get hint related to question. If performance is up to level, he will go to next courseware. If performance is not up to level he will repeat same courseware. After three attempts if performance is not up to level he will get specific treatment)
Table 1: Examination data of student.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6 (1st)</td>
<td>5 (2nd)</td>
<td>4 (1st)</td>
<td>3 (1st)</td>
<td>5 (3rd)</td>
<td>4 (1st)</td>
<td>3 (1st)</td>
<td>3.5</td>
<td>good</td>
</tr>
<tr>
<td>2</td>
<td>5 (2nd)</td>
<td>6 (1st)</td>
<td>3 (1st)</td>
<td>4 (1st)</td>
<td>5 (2nd)</td>
<td>6 (1st)</td>
<td>4 (1st)</td>
<td>4.2</td>
<td>good</td>
</tr>
<tr>
<td>3</td>
<td>3 (1st)</td>
<td>5 (1st)</td>
<td>4 (1st)</td>
<td>5 (2nd)</td>
<td>3 (1st)</td>
<td>4 (1st)</td>
<td>4 (1st)</td>
<td>3.7</td>
<td>good</td>
</tr>
<tr>
<td>4</td>
<td>5 (1st)</td>
<td>4 (1st)</td>
<td>5 (1st)</td>
<td>6 (2nd)</td>
<td>4 (1st)</td>
<td>3 (1st)</td>
<td>4 (1st)</td>
<td>4.1</td>
<td>good</td>
</tr>
<tr>
<td>5</td>
<td>6 (1st)</td>
<td>6 (2nd)</td>
<td>6 (2nd)</td>
<td>6 (1st)</td>
<td>6 (2nd)</td>
<td>6 (1st)</td>
<td>6 (2nd)</td>
<td>4.8</td>
<td>Excellent</td>
</tr>
</tbody>
</table>

he will repeat same courseware. After three attempts if performance is not up to level he will get specific treatment)


13. If his performance is up to level he can go to CW[7]. If performance is not up to level he has to repeat same courseware (he will get up to three attempts. In 2nd and 3rd attempt he will get hint related to question. If performance is up to level, he will go to next courseware. If performance is not up to level he will repeat same courseware. After three attempts if performance is not up to level he will get specific treatment.)

14. Performance in CW[7] is evaluated. If performance is up to level his final score will be calculated based on the equation (1) given in following section. If performance is not up to level he will repeat CW[7] up to three attempts. After three attempts if performance is not up to level he will get specific treatment.

4 Performance Evaluation

In [11] We have made analytical (A), reasoning (R) and descriptive (D) type questions from each courseware for evaluating the performance of student. Although there has been more 4 divisions of the type of questions such as AR(Antalytical-Reasoning), RD (Reasoning - Descriptive), AD (Analytical - Descriptive), and ARD (Analytical - Reasoning - Descriptive).

Student’s performance is up to level, if he gets greater or equal to 1 (1 is threshold value) in analytical(A), and in reasoning(R) and in descriptive(D) and his average of marks in A, R and D is greater or equal to 1. If this condition is not met his performance is not up to level. Based on this performance and number of attempts performance index can be calculated using following formula i.e. equation (1).

\[ PI = \frac{1}{q} \sum_{i=1}^{7} P_i \times W_n \]  

In equation (1) PI is performance index. q = number of courseware, here q=7 because there are seven courseware. Pi is performance of the student in ith courseware. Performance is summation of marks obtained by student in analytical, reasoning and descriptive type question-naire. For example if student got 2 in analytical, 2 in reasoning and 2 in descriptive in two attempts in a courseware, his performance will be 2+2+2=6 and Wn (weight of marks) will be 2/3.

Wn is the weight of marks obtained by student. It can be 1, 2/3 or 1/3. It depends on the number of attempts in which student’s performance is up to level. If his performance is up to level in first attempt, Wn will be 1. If performance is up to level in 2nd attempt Wn will be 2/3. If his performance is up to level in three attempts Wn will be 1/3.

Based on the value of performance index “excellent”, “good”, and “average” grade have been given to the student. Following rules have been used for grading in the scale of 1 to 6:

Rule 1: If PI is between 1 to 2.5 grade= “Average”

Rule 2: If PI is between 2.51 to 4.5 grade= “good”

Rule 3: If PI is between 4.51 to 6 grade= “excellent”

5 Experimentation

In the experimentation we have taken the students of BTech I and II semesters of our department. Questionnaire of subject topic category i.e. analytical (A), reasoning (R), descriptive (D), was given to the student in each courseware. Examination data have been given in table 1.
Table 2: Comparative view of our method with other’s method.

<table>
<thead>
<tr>
<th>Author</th>
<th>Computing Model</th>
<th>Evaluating Parameter</th>
<th>Evaluation measure</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our method</td>
<td>Heuristic method</td>
<td>Number of attempts, Hints</td>
<td>Average, good, excellent</td>
<td>JSP</td>
</tr>
<tr>
<td>Zhou et al. (1999)[18]</td>
<td>Heuristic method</td>
<td>Number of errors the student made while solving the problem, Hints</td>
<td>Poor, good</td>
<td>LISP</td>
</tr>
<tr>
<td>El-Khouly et al. (2000)[6]</td>
<td>Heuristic method</td>
<td>Number of questions which had been asked to the student, correct answer</td>
<td>Not specify</td>
<td>Java, HTML</td>
</tr>
<tr>
<td>Tchetagni et al. (2002)[16]</td>
<td>Mathematical method</td>
<td>Number of solved exercises, average time to solve exercises, number of tutor interventions(hint)</td>
<td>Not specify</td>
<td></td>
</tr>
<tr>
<td>Hatzilygeroudis et al., 2004 [8]</td>
<td>Algorithmic method</td>
<td>Number of times the user asked for assistance (hint), the number of related examples requested by the user, the number of answering attempts made by the user</td>
<td>Low, high, average, excellent</td>
<td>Not specify</td>
</tr>
<tr>
<td>Paranuthui (2005) [12]</td>
<td>Heuristic method</td>
<td>The amount of time spent per lesson, the amount of (uninterrupted) time spent per session, the number of times the student went back, the frequency of help requests(hint), average time spent on a given “page” during a lesson plan</td>
<td>Learning effective(yes/no)</td>
<td>Not specify</td>
</tr>
<tr>
<td>Ferguson et al. (2006)[7]</td>
<td>Mathematical method</td>
<td>Number of problems the students answered correctly, the number of problem the student attempted to answer (did not skip), Hint</td>
<td>Not specify</td>
<td>Not specify</td>
</tr>
<tr>
<td>Stankov et al. (2007)[14]</td>
<td>Algorithmic method</td>
<td>Problem difficulty, Hint</td>
<td>insufficient mark,</td>
<td>ASP, ODBC etc.</td>
</tr>
</tbody>
</table>

There are 10 columns and 6 rows in table 1. Column 1 represents student number, column 2, 3, 4, 5, 6, 7 and 8 represents performance of student in CW1, CW2, CW3, CW4, CW5, CW6 and CW7 respectively. Information i.e. 1st, 2nd and 3rd in brackets in these columns represents attempts in which student’s performance is up to level in the courseware. Column 9 represents PI(Performance Index) of the student calculated by the system using equation (1). Column 10 represents grading assigned based on PI by the system to the student.

6 Result

Results and intermediate results of the system have been shown in this section in figures 2, 3, and 4. Figure 2 shows that student starts his study from CW1 by clicking on sequential study. Figure 3 shows student’s performance in analytical, reasoning and descriptive and average performance in a courseware. Figure 4 shows student’s overall performance (performance in CW1 to CW7), performance index and grade obtained by the student. Column 1 in figure 4 shows courseware, column 2 shows number of attempts in which student’s performance is up to level, column 3, 4 and 5 shows marks obtained by student in analytical, reasoning and descriptive respectively. Total marks obtained by student based on number of attempts have been shown in column 6.

Figure 2: Student starts his study by clicking on sequential study.
7 Comparison

In this section we have represented a comparative view of our method with other’s methods. A comparative view of our method with other’s method has been shown in table 2.

Figure 5 represents a qualitative evaluation of intelligent tutoring systems. We have been taken 10 ITSs out of which 40 percent ITSs have been used heuristic method, 30 percent ITSs have been used mathematical method and 30 percent ITSs have been used algorithmic method for performance evaluation. 90 percent ITSs have been used hint and 40 percent ITSs have been used number of attempts as parameter for performance evaluation. 40 percent ITSs have been used qualitative evaluation measure (Q Measure) i.e. low, good, and excellent for performance evaluation. One ITS has been used other measure (Learning effective Yes/No ) while 50 percent ITSs have not been used(No measure) any qualitative evaluation measure for performance evaluation.

8 Conclusion

A learning strategy for study and a heuristic method for performance evaluation of learning in an intelligent tutoring system has been developed and presented in this paper. It is observed from the table2 and the graph (figure 5) that heuristic methods have been deployed highly in comparison to the mathematical and algorithmic methods. Number of attempts with hint is more favorable and recommendable to the students. Qualitative measure such as average, good and excellent can also be put in the category of very low, low, medium, high, very high mapping in the range of 0-10. Most of the ITSs are web based services deploying JSP, ODBC, HTML etc. Taking into account much more related researches, and number of students in experimentation the table and the graph can be extended. There is a scope of further research using ANN and Bayesian network.

References


