© 2016 IJSRSET | Volume 2 | Issue 6 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Intelligent Tutoring System using CBM

Anagha A. Ralegaonkar[1], Dr. S. B. Thorat[2], Dr. P. K. Butey[3]

[1]Research Scholar, Department of Electronics & Computer Science, RTMNU, Nagpur, Maharashtra, India
[2]Director, SSBES's, Institute of Technology & Management, Nanded-431602, Maharashtra, India
[3]HOD & Associate Professor, Computer Science, Kamla Nehru Mahavidyalaya, Nagpur, Maharashtra, India

ABSTRACT

The Internet provides an infrastructure that supports unprecedented communication capabilities and collaboration opportunities. In the field of education, it supports collaboration between various domain experts and teachers in designing novel approaches to teaching and co-operation among teachers who can share instructional material. An intelligent tutoring system is computer software designed to simulate a human tutor's behavior and guidance. Intelligent Tutoring Systems (ITSs) are designed with using Artificial Intelligence (AI) techniques and called cognitive tutors or Knowledge Based Tutoring Systems which can guide learners to progress in the learning process at their best and to facilitate instruction. Intelligent Tutoring Systems have revolutionized online education by providing individualized instruction tailored towards each student. Constraint-based tutors model instructional domains at an abstract level, a novel approach that simplifies the development of ITSs. This paper describes structure of ITS and deals with the CBM for development of Intelligent tutoring systems.

Keywords: Constraint based modelling (CBM), Intelligent Tutoring Systems (ITS), Pedagogical Module, Report Module

I. INTRODUCTION

It is well known that one-on-one human tutoring is much more effective than traditional classroom instruction, An Intelligent Tutoring Systems is educational software containing an Artificial Intelligence component. The software tracks the student's work, tailoring feedback and hints along the way. By collecting information on a particular student's performance, the software can make inferences about strengths and weaknesses, and can suggest additional work. The goal of the research in the area of Intelligent Tutoring Systems (ITS) is to build computer-based tutors that achieve the effects of learning individually with a human tutor. ITSs have been shown to be effective in increasing student's performance and motivation levels compared with traditional instructional methods.

ITS must be able to achieve three main tasks:

- ✓ Accurately diagnose a student's knowledge level using principles rather than programmed responses.
- ✓ Decide what to do next and adapt instruction accordingly.
- ✓ Provide feedback.

II. METHODS AND MATERIAL

Architecture of an ITS

In the proposed ITS consist of six emerging subsystems, namely:

Student Module
Knowledge Module
Tutor Module
Pedagogical Module
Report module
User Interface Module

Two major issues related to an ITS are "what to teach" and "how to teach". The student module and knowledge module deals with the "what to teach" part, whereas the tutor module and pedagogical module are concerned with "how to teach" part. This proposed module also deals with "how to report" part by using report module and user interface module.

The knowledge Module is also important as it is the representation of the domain knowledge. The most important part of an ITS is the tutor module. This module is the centre of the whole system. It communicates with the other modules and does the entire decision making. The structure of the ITS is shown in the figure 1.

A. Student Module

The student module contains descriptions of student knowledge or behaviors, including his misconceptions and knowledge gaps [1, 10]. In other words, the cause of collecting information about student is to determine the education level of student and the most suitable learning method for him. The collected information is stored in data base and whenever necessary it can be retrieved and used.

ITSs serve two basic aims [5, 6, 11, 14, and 18]:

1. To form a learning program adaptable according to the student
2. To be a guide to solve student's problem.

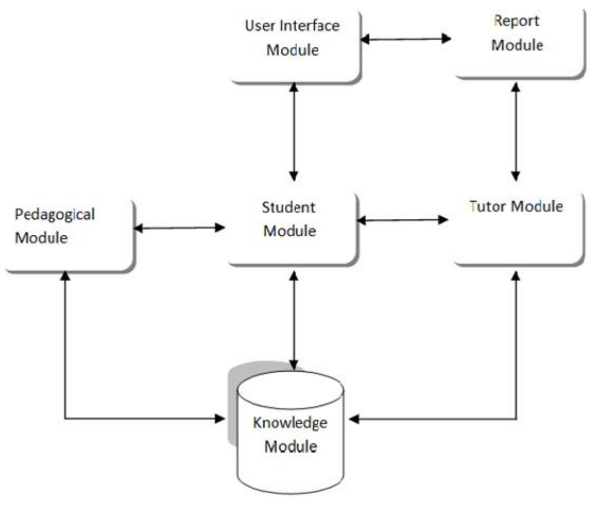

Figure 1. Structure of ITS

B. Knowledge Module

The Knowledge module references an expert or domain model consist of a description of the knowledge or behaviors that represent expertise in the subject-matter. It is the module in which main information and tutorial information that are going to be taught resets. We can say that the designing of better knowledge module helps more in proper tutoring and assessment. It is to say, when a student module is identified, the data base suitable for this module should be able to present to the user easily.

C. Tutor Module

A mismatch between a student's behavior or knowledge and the expert's presumed behaviors or knowledge is signaled to the tutor module, which subsequently takes corrective action, such as providing feedback or remedial instruction with the help of pedagogical module. This module contains the information which includes tutoring strategies and tactics which are stored in knowledge module. Tutor module provides necessary information so that tutoring aims can be achieved. Besides, it will answer students' questions properly and will present the needed help when they solve a problem or perform their skills. It must have a mechanism which will be able to determine what kind of help to be presented.

D. Pedagogical Module

The pedagogical module is the driving engine of the teaching system and is closely linked to the student module. It designs and controls instructional interactions with the student for their better understanding. It uses the student model and knowledge model to make its pedagogical decisions. The pedagogical module forms and updates the student model and offers hints when the student is struggling, supplies advice, support and explanations, selects a new topic, etc.

E. Report Module

This module generates report related with the performance of the student as well as produces status of knowledge acquainted by the student. This module also support tutor to update the information about his student. A report module visualizes the knowledge module and the Student Module to analyze the various learning parameters of the learner such as the allotted time for studying a specific content in order to generate report of assessment with the help of tutor module. Based on the strategy of the tutor module, the learner at any time is able to choose the learning contents himself which leads to the updating of the Student Model

F. User Interface Module

The user interface module provides the means for the student to interact with the ITS, usually through a graphical user interface and sometimes through a rich simulation of the task domain the student is learning [5, 10, 13, and 18].

Student modeling is widely accepted as one of the central problems in ITSs. If a system is to be effective, it must reason about student's knowledge and adapt its actions to the needs and abilities of each individual student. Many SM approaches have been suggested over the years. Some of them are suited to a particular domain, or a teaching strategy. The research presented here focuses on effective and computationally tractable student modeling. We adopted Constraint-Based Modeling, a student modeling approach recently proposed by Ohlsson and have had extremely good experiences with it [15].

III. RESULTS AND DISCUSSION

Constraint-Based Modeling

Constraint-based tutoring is now a mature and successful student modeling approach to provide adaptive learning environments. Constraint-based modeling uses abstraction to avoid the student's misconceptions. Constraints represent only correct knowledge in terms of pedagogically significant states; each constraint maps to a set of solution states that share the same domain principle. A constraint-based ITS can therefore react in the same way (e.g. by displaying the same feedback message) for any solution that violates a given constraint.

Constraints have three components: a relevance condition, a satisfaction condition and the feedback message. The relevance condition (Cr) describes (in terms of problem/solution features) when this constraint is applicable. The satisfaction condition (Cs) then specifies additional tests to be applied to the solution to check its correctness. This way, constraint violations allow an ITS to react at the right time, and also govern the instruction to be delivered. The feedback message attached to the constraint tells the student that his/her solution is wrong, points out why it is wrong, and reminds the student of the corresponding declarative knowledge (i.e. the domain principle that is violated by the solution).

Constraints are used to represent the knowledge by the tutor model also they serve to check student's knowledge. In our proposed system When a student submits a solution, a ITS analyses it using the constraints, relevant constraints are identified and their satisfaction conditions determine whether they have been satisfied or violated. Pedagogically CBM determines the content of instruction provided. If there are errors in a student's action, the ITS will present feedback provided by the violated constraints. The form of this feedback is shaped by the underlying learning theory: it should tell the student what domain principle he/she has violated how it was violated by the student's solution and reiterate the correct domain principle. For e.g. in our proposed system KITWEB, Knowledge based Interactive Tutoring system for WEB application testing, to check weather the student user understands concepts related with the functionality testing the system allows its student user to test the text box for field Name, the constraint written is

Cr: The student is trying to test the text box associated with the Name field

Cs: student should enter valid combinations of alphabet to check the functionality of the text box.

If this latter condition is not met, appropriate feedback can be given, such as, "You are not going in proper way. Please enter alphabets A-Z or a-z".

If the later condition is met system produces feedback "WELL DONE" by using the effective pedagogical agent.

An interesting side-effect of this approach is that CBM is silent to situations it has no knowledge of it. Therefore it is described as CBM's approach, if there are no constraints violated by a particular solution, it is found to be correct.

CBM also requires little computational effort for constraint matching, since checking Cr and Cs is essentially pattern matching.

IV. CONCLUSION

The approach of constrained-based student modeling was proven to be successful to provide adaptive learning environments by individualizing instructions and a promising method for representing domain and student models in our proposed Intelligent Tutoring System. The computational overhead that was previously needed in other student modeling techniques was greatly reduced this way. Instead of providing an extensive bug list or diagnosis model, a more simple constrained-based error list is needed.

V. REFERENCES

[1] M. Ayop, K. Chaellappan, M. A. Nazlena," Intelligent Tutoring Tool for Digital Logic Design Course", In Proc. Of Intetnational Conference on Electrical and Electronic Technology (IEEE TENCON 2001), Singapore, Agusut 2001, pp. 19-22.

[2] H.S. Nwana, Intelligent Tutoring Systems: An Overview, Artificial Intelligence Review, vol. 4, 1990, pp. 251-277. International Journal of Innovation Management and Technology, Vol. 2, No. 5, October 2011 428

[3] AWeb-based Bayesian Intelligent Tutoring System for Computer Programming C.J. Butz, S. Hua, R.B. Maguire Department of Computer ScienceUniversity of Regina Regina, Saskatchewan, Canada S4S 0A2 Email: {butz, huash111, rbm}@cs.uregina.ca

[4] "Andes Physics Tutoring System", International Journal of Artificial Intelligence in Education, 2005.

[5] J. pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, 1988.

[6] W.L.Johnson, Pedagogical agents for Web-bases learing in: Proceedings of First Asia-Pacific Conference on Web Intelligence, Maebashi City, Japan, 2001.

[7] http://www.ai.mit.edu/~murphyk/Bayes/bnsoft.html

[8] Designing Intelligent Tutoring Systems:A Bayesian Approach, ICEIS 2001 - Artificial Intelligence and Decision Support Systems

[9] Hossam Meshref, An Intelligent Tutoring System for Logic Circuit Design Problem Solving, International Journal of Computer Applications (0975 – 8887) Volume 35– No.2, December 2011

[10] J. Vassileva, Dynamic course Generation on the WWW, Proceedings of the workshop ITS's on the WWW 8th World Conference of the AIED Society, Kobe, Japan, August 1989, pp. 18-22.

[11] D. C. Merrill, B. J. Reiser, M. Ranney, J. G. Trafton, Effective Tutoring Techniques: A Comparison of Human Tutors and Intelligent Tutoring Systems, The Journal of the Learning Sciences, 1992, pp. 277-305.

[12] C. Frasson, E. Aimeur, Designing a Multi-strategic ITS for Training in Industry, Elsevier Science Computers in Industry, 1998, vol. 37, pp. 153-167.

[13] R. Nkambou, C. Frasson, g. Gauthier, A New Approach to IT Scurriculum and Course Authoring: The Authoring Environment, Elsevier Science Ltd. Pergamon Computers Educ, great Britian, 1998,vol. 31, pp. 105-130.

[14] J. A. Self, The Distinctive Characteristics of Intelligent tutoring Systems Research: ITSs Care, Precisely, International Journal of Artificial Intelligence in Educations, 1999, pp. 350-364.

[15] M. A. Mark, J. E. Greer, Evaluation Methodologies for Intelligent Tutoring Systems, Journal of Artificial Intelligence in Education , 1999, 4 (2/3): 129-153.