REDEFINITION OF BASIC MODULES OF AN INTELLIGENT TUTORING SYSTEM: THE TUTOR MODULE

Fernando Salgueiro[1,2], Guido Costa[1,2,], Zulma Cataldi[1], Fernando Lage[1], Ramón García-Martínez[2,3]

1. LIEMA. Laboratorio de Informática Educativa y Medios Audiovisuales: liema@fi.uba.ar
2. LSI - Laboratorio de Sistemas Inteligentes.
Facultad de Ingeniería. Universidad de Bs. As. Av. Paseo Colón 850. 1063 Cdad. de Buenos Aires.
3. Centro de Ingeniería del Software e Ingeniería del Conocimiento. Escuela de Postgrado. ITBA.
Av. Madero 399. Anexo 3º Piso. Ciudad de Buenos Aires. Argentina.

Abstract

It has been noticed that during the initial semesters of Computing Engineering that the amount of human tutors is insufficient. The students/tutors ratio is very high and there is a great difference in the acquired knowledge and backgrounds. The idea is that a system could emulate the human tutor and besides provide the student with a degree of flexibility for the selection of the most adequate tutorial type. This could be a feasible solution to the stated problem. But a tutorial system should not only emulate the human tutor but besides it should be designed from an epistemological conception of what teaching Basic Programming means specially in an Engineering course due to the profile and identity of the future engineer.

Keywords**:** Intelligent Advisor Systems, Intelligent Tutoring Systems
Area: Artificial intelligence

1. Introduction

During the past six semesters students have been monitored (through their mid terms and final examinations) in order to determine the reason why some students fail to pass the subject. Despite the fact that the percentage of students that complete the subject, around 30%, for which they are currently working out strategies intended to palliate the situation. [1,2]. Throughout the past three years the use of diverse didactic strategies such as audiovisuals means, discussion groups, learning groups [3] has shown some improvements, these strategies are aimed at students which normally would not have any difficulties. It was for this reason, that the development of an intelligent advisor (using Intelligent Systems) was an appealing idea. This advisor would perform the tutoring activities by adapting different teaching way or strategies. This alternative could be useful specially for those students that need a high degree of one on one tutoring [4]. Modelling of intelligent tutor systems (ITS), implies taking into consideration the three basic modules of the tripartite architecture proposed by Carbonell [5]. At the time of modelling an ITS they should be considered the characteristics of the domain (content), of the behaviour of the student (student model) and of the set of strategies that would be tackled by the tutoring module. In this communication it will be presented a re-modelling of the basic modules of an intelligent advisor in order to satisfy the needs of the students of the subject Algorithms and Programming I (included in the area of Basic Programming of the Computing Department of the Engineering Faculty of the University of Buenos Aires). It has been noticed that in classic intelligent tutors, the control criteria changes according to some optimisation principle, and many times it is just an implementation decision, which means, there is no difference in the way in which the interaction is controlled. Some systems may monitor students' activities, adapting their actions to the answer given, but never by transferring the interaction control. An example of this system is the intelligent substitute for to learning called CIRCSIM. [6; 7; 8]

2. SOLUTION DESCRIPTION

Taking the classic structure of the intelligent tutoring systems as a starting point, the basic components and interfaces of each of the modules will be redefined, some modules will be add to satisfy the needs of both students and teachers, in order to perform the teaching task in an effective way. Many efforts have been taken towards the separation of the three fundamental modules of the intelligent tutoring systems, but subsequent implementations have shown that the domain can not be

separated completely from the tutor and student modules. As a first step it was necessary to analyse the interaction of the modules for a general hypothetical domain, and then all the necessary modifications were made for the particular domain of teaching the Pascal programming language.
An intelligent tutor system behaves as a professor (instructional systems). Normally, it is the one which controls the work flow, and leads the interaction with the student. On the other hand, intelligent advisor systems are oriented to help the student through its learning, the system makes suggestions with the goal that they will help the student to improve his performance (acts as an assistant). The student is the one which leads the work and the interaction, and performs this in the order which he considers the most convenient. The modules can be basically defined as:

2.1 Student module

This module should model the individual characteristics of the student, among then one of the most important is the instant individual knowledge about the domain, Robles [9] defines it as: *"The model of the student reflects how much does the student know about the domain, as well as its cognitive experiences and of learning, from which a diagnostic can be made"*. Besides it should be considered that this module will have to be able to interact with its pair, which a fundamental characteristic of intelligent advisors.

2.2 Tutor's module

This module posses the knowledge about teaching strategies and tactics, in order to choose then regarding the characteristics of the student (which are stored in the student module). Robles [9] defines it as: *"the expert or domain model, which deals with the subject or course to be imparted"*.

2.3 Domain's module

This module posse the knowledge of the subject formed by the production rules, stereotypes, etc. From this module the tutor can obtain the knowledge to be imparted.

2.4 Interface module

This is the interface between the ITS and the real student, its function is to present the domain's material and any other didactical material in a proper way. According to Robles: *"The interface, allows the users to interact with the system. Three specific types of users can be distinguished: the Student, the Instructor and the Developer of the system"*. [9]

2.5 Evaluation module

Performs a general evaluation of the system and to create the necessary feedback with the teacher, it should be pointed out that intelligent advisors do not replace a human teacher, but they rather acts as a complement, therefore this module should give the human teacher information which would help him to modify the teaching methods used with the students during classes.

3. Interaction Between Fundamental Modules.

Figure 1 shows the four fundamental modules and the way they interact with each other for any given domain. This approach separates completely the domain of the module from any kind of overlapping with the other modules. Interactions happen in the following way:

3.1 Determination of the Student's Style.

A chart with questions is handed out to the students, in order to categorise them within the available styles of the system. The student should hand in the completed chart and the student model categorises him within the available styles. This task is performed only once.

3.2 Creation of the Knowledge State:

Being based on the domain model, the student model is responsible for the generation of the Knowledge States of the student; which will be updated as the user becomes qualified in tutoring performance with the system as in external tutoring during classes. In the latter case, after classes with a human tutor and with the attendance list the system would generate the new knowledge state of the student.

3.3 Determination of the Pedagogical Style:

The model of the student delivers the style which best suits the user; then the tutor module selects the most adequate pedagogical style, according to the characteristics of each student in order to give classes.

3.4 Planning of the Lesson:

According to the selected pedagogical style the knowledge state of the student is transferred to the student module. Taking into account both the style and the knowledge state a lesson is planned, and the required information for the lesson is handed out to the domain module, following the specific order in which were planned by the tutor module.

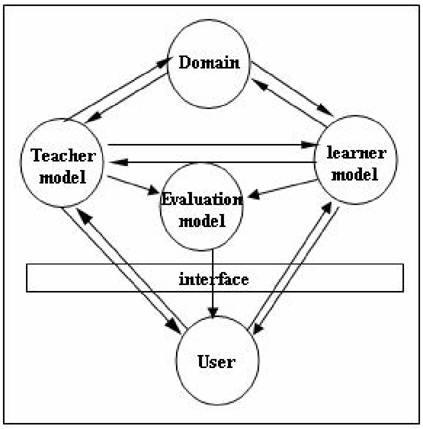

Figure 1. Interaction between the modules of an intelligent tutoring system.

3.5 Pedagogic Session:

The tutor module imparts the planned knowledge during the class according to the selected pedagogical style. This will go through a process which will shape it into a natural language to ease the interaction with the user and it will be presented through an interface, which will arrange the pedagogical elements used for the particular lesson. During the pedagogic session, the tutor module will have general objectives to carry out (to give the solution to a problem, to assure that the student reaches a certain degree of knowledge about the theme, etc), but through the interaction with the user many secondary objectives rise, necessary and not, to achieve the main purpose of the pedagogic session. It would be responsibility of the tutor module to guide the user towards every objective, and to continue the pedagogic session until the goals have been satisfied.

3.6 Evaluation of the Pedagogic Session:

The results of the session are compiled and interpreted by the tutor module. The starting points are questions, exercises, etc. and the obtained result is knowledge (or not) of the imparted subjects. This also updates the heap of objectives to fulfil for the session that is taking place.

3.7 Updating the knowledge map:

Once the results of the evaluation about the imparted lesson are processed, the student module updates the Knowledge State of the user; this can modify the predictions about the learning type in which the user was classified, his beliefs about the domain and may obtain statistics. The systems may act as a tutor, or as a classmate; where the student module uses the characteristics of classmate or assistant, in this case its main objective is to adapt to the actions and answers of the students.
The evaluation module plays two fundamental roles, firstly provides initial information to the student module about its knowledge state. Secondly, presents reports about the state of the system, whether it is individual (by student) or general (by group of students), so that the human teacher would be able to make decisions that could improve the general performance of the group.

4. BASIC COMPONENTS OF THE TUTOR MODULE.

The new definition of the tutor module consists of two large modules as shown in Figure 2, and that detailed as stated below:

4.1 Module of the Pedagogic Styles.

As it is shown this module consists of a database of pedagogic styles that are available in the system, and the selection methods and its characteristics. The usable procedures must contain the necessary information and pedagogic tools in order to impart a pedagogic session in a way that the student user would benefit the most (as an example, for a socratic session the system must store the algorithm for the resolution of problems, which will be stockpiled in the domain module and for a magisterial style it must have tools to show slides, videos, sounds, etc; which would be used to enhance the student experience [12] [16] [17].

4.2 A Lesson Planner:

According to the selected pedagogic method, the knowledge state of the student; the content generator diagrams the lesson in order to achieve its objectives, which will be stored in the objectives heap. If during the lesson new secondary objectives rise, or the system establishes that previous objectives should be achieved in order to meet the general objective of the session) these are added to the heap and are tried to be fulfilled before the main objective is reached. The output is transformed from internal knowledge representation structures into a natural language through a special module, which is dedicated to perform this task, enhancing the user experience. The sub-module of natural language not only does generate the natural language for the session, but also obtains the relevant concepts asked to the user and which are later evaluated.

4.3 Determination of the Pedagogic Style:

The model of the student delivers the style which best suits the user; then the tutor module selects the most adequate pedagogical style, according to the characteristics of each student in order to give classes [13] [14] [15].

4.4 Planning of the Lesson:

Based on the selected pedagogic style the knowledge state of the student is transferred to the student module. Taking into account both the style and the knowledge state a lesson is planned, and the required information for the lesson is handed out to the domain module, following the specific order in which were planned by the tutor module [10] [11] [12] [18].

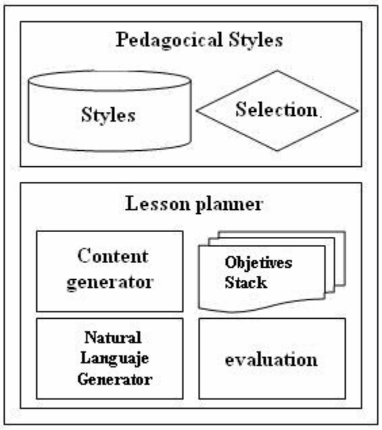

Figure 2. Basic components of the tutor module.

4.5 Pedagogic Session:

The tutor module imparts the planned knowledge during the class according to the selected pedagogical style. This will go through a process which will shape it into a natural language to ease the interaction with the user and it will be presented through an interface, which will arrange the pedagogical elements used for the particular lesson.With a given pedagogic session, the tutor module will have general objectives (as it may be to give the solution to a problem, to reach a certain degree of knowledge about a theme of the curricula, etc), but during the interaction with the user secondary objectives may appear, and they can be necessary or not to reach the main purpose of the pedagogic session. It would be responsibility of the tutor module to guide the user correctly towards every objective, and to continue the pedagogic session until all the goals have been satisfied [10].

4.6 Evaluation of the Pedagogic Session:

The results of the session are compiled and interpreted by the tutor module. The starting points are questions, exercises, etc. and the obtained result is knowledge (or not) of the imparted subjects. This also updates the heap of objectives to fulfil for the session that is taking place.

4.7 Updating the knowledge map:

Once the results of the evaluation about the imparted lesson are processed, the student module updates the Knowledge State of the user; this can modify the predictions about the learning type in which the user was classified, his beliefs about the domain and may obtain statistics.

Besides the normal system interaction, when it is working as a tutor, the system can functions as a classmate, where the student module acquires characteristics of a classmate or assistant, in this case

its main objective is to adapt to the actions and answers of the students; but it can never control the interaction.
The evaluation module plays two fundamental roles, firstly provides initial information to the student module about its knowledge state. Secondly, presents reports about the state of the system, whether it is individual (by student) or general (by group of students), so that the human teacher would be able to make decisions that could improve the general performance of the group.

5. Conclusions

The redefinition of modules, into sub modules and interfaces intends to solve the general problems that affect intelligent tutoring systems for several years. Despite the classical structure of intelligent tutor systems remains intact, the circuits are made easier and the interaction between modules is enhanced, this new approach implies greater flexibility and adaptation to the applications that would make use of it.

6. Future Work

In the present it is being developed the modules of an integral intelligent tutoring system, which will use the proposed redefinitions of the classical structure of the student and tutor modules and the evaluating system using the appropriate development tools and methodologies.
Once the implementation and integration of the modules is finished, it will be necessary to validate the data with a test group, which will use the advisor during the practice in the Faculty Laboratories. These data will be compared with the information gathered until that moment and conclusions will be drawn, indicating the feasibility of the implementation and latter effective use of an intelligent advisor system as a complement to the theory classes of the subject Algorithms and Programming I of the Faculty of Engineering of the University of Buenos Aires.

7. REFERENCES

[1] Madoz, C; DeGiusti, A. Vinculación de un curso interactivo multimedial. CACIC97. La Plata. UNLP. 1997
[2] Bruno, O. Los problemas de aprendizaje de algoritmos. Comunicación personal. UTN. 2003
[3] Souto, M. Hacia un Didáctica de lo grupal. Miño y Dávila 1995
[4] Perkins, D.: La escuela inteligente. Gedisa 1995
[5] Carbonell, J. R. AI in CAI: An artificial intelligence approach to computer assisted instruction. IEEE transaction on Man Machine System. V 11 n.4, p 190-202. 1970.
[6] Kim, Nakhoon. CIRCSIM-Tutor: An Intelligent Tutoring System for Circulatory Physiology Ph.D., Illinois Institute of Technology, 1989
[7] Kim, Jung Hee Natural Language Analysis and Generation for Tutorial Dialogue. Ph.D., Illinois Institute of Technology, 2000
[8] Cho, Byung. Dynamic Planning Models to Support Curriculum Planning and Multiple Tutoring Protocols in Intelligent Tutoring Systems.
[9] Robles, B. Asesores inteligentes para apoyar el proceso de enseñanza de lenguajes de programación. tesis de grado. ITSM. 1993.
[10] Wenger, E. Artificial intelligence and tutoring systems. Computational and Cognitive Approaches to the Communication of Knowledge. Los Altos C. A. Morgan and Kaufman. 1987
[11] Hume, G. Using student modeling to determine when and how to hint in an intelligent tutoring system. Illinois Institute of Technology.
[12] Hume G., Michael, J; Rovick, A. y Evens, M., Hinting as a tactic in one-on-one tutoring. Journal of Learning Sciences. 1995
[13] Norman, D.: Perspectiva de la ciencia cognitiva. Paidos. 1987
[14] Gardner, H. Las inteligencias múltiples. La teoría en la práctica. Paidós. Barcelona. 1993
[15] Gardner, H. La nueva ciencia de la mente: Historia de la psicología cognitiva. Paidós. Barcelona. 1987
[16] Cruz Feliú, J. Teorías del aprendizaje y teorías de la enseñanza. Trillas- 1997
[17] Bruner, J. Actos de significado. Más allá de la revolución cognitiva. Madrid: Alianza. 1991
[18] Schunk, D. Teorías de la Educación, Prentice Hall. 1997.