The Design of a ‘Motivating’ Intelligent Assessment System

Katerina Georgouli
Technological Educational Institute (T.E.I.) of Athens, Department of Informatics, Agiou Spiridona, 12210 Egaleo
kgeor@teiath.gr

Keywords: *Intelligent Tutoring Systems, Adaptive Assessment, Motivation, Problem Solving.*

Abstract. This paper intends to present the design of an intelligent assessment system, which attempts to assess the student in arithmetic word problem solving. During assessment, the system keeps track of the aptitudes, which the student shows, concerning the answers he/she gives to special types of problems and in the same time it observes aspects of the student’s motivational state. More precisely, motivation aspects deal with a) the effort the student shows in solving the different types of problems and b) with the independency and the confidence that characterize his/her behaviour. The system tries to adapt itself according to the above information, in order to motivate the student, offering her the appropriate help and the possibility to follow an individualized way through the objective items of the assessment.

1. Introduction

Many attempts have been made in the field of Artificial Intelligence (AI) to implement acceptable by the schoolteacher systems [1,2]. The first and still foremost contribution of AI to education is the so-called intelligent tutoring system (ITS). In particular, ITSs are computer-based learning systems which attempt to adapt themselves to the needs of learners and are therefore the only such systems which attempt to ‘care’ about what the student knows, wants to do, is able or unable to understand, tries to avoid, etc [3]. For a tutoring program to be classified as ‘intelligent’, it must have ‘human-like’ tutoring capabilities, like being able to adjust the content and delivery of the lesson to the needs of the student by analysing responses and behaviour. This is usually done, by tracing the path of the student’s understanding through the curriculum.

On the other hand, one of the main concerns in Education is to make the instruction an interesting and engaging experience for the student. In fact, very little research has been done in motivational aspects of instruction in ITSs, although many of them use multimedia facilities to motivate the student. The need for basic computational models of motivation and the use of the design-based approach of artificial intelligence to lay down such models is advocated by many of the researchers in the area. There already exist a few useful models [5,6], characterized by their efforts to organize into some structure variables that have been shown empirically to affect student motivation, including some of the inner conceptions of the student. In the cognitive perspective, student motivation is defined in terms of the individual's commitment and persistence in choices of plans and actions. Keller [7] in his ARCS Model defines four components that influence the motivation of a learner: attention, relevance, confidence and satisfaction. The information needed for a motivation diagnosis is selected by questionnaires, verbal communication, self-report, expert systems and sentic modulation [8]. Lepper et al. [4] were between the first researchers who suggested that some additions should be made to a computer tutor in order to provide it with an ability to detect a student's motivational state. Computer diagnosis of motivation is usually done by adding specialized motivational components in the standard ITS architecture. Del Soldato et al. [9], more recently, has suggested some additions that should be made to a computer tutor in order to be able to detect a student motivational state. Particularly, she added two new modules to the traditional ITS architecture: a motivation modeller and a motivational planner. In this way, her system was able to detect the student's motivational state, concerning his effort and confidence, exploiting the pattern of standard reactions.

In this paper, we present the design of an intelligent assessment system able to adapt itself to the student's aptitudes and motivational state. The system concerns students who show remarkably low performance in a domain and who need to be faced in a special way. The design follows the prevailing expert system and student modelling approach and has in general the architecture of an ITS. In section 2, we consider the knowledge representation and in section 3, the architectural ideas, which permit the system to detect the student's aptitudes, to diagnose her motivational state and to react appropriately. In section 4, we focus on the student modelling aspects, which guide the adaptation of the system. In section 5, we present in brief the results of a preliminary evaluation of the prototype system that has been implemented according to the proposed design. The results we present here, concern mainly the accuracy of the provided by the system student profile and the reliability of those aspects of the student model that guide the adaptation of the system.

2. Knowledge Representation

In our system, the domain knowledge is split into small "chunks" of knowledge, called *Learning Units* (LU), which are linked to one another through semantic associations of the type, *is-part-of.* Learning units that represent the different types of problems of the domain are called *basic learning units (BLUs)* and are grouped, according to their common characteristics, to major groups called *Classes of Learning Units (CLU),* which in their turn are grouped to major classes of CLUs, called *Major Classes of Units (MCLUs).* This way, MCLUs, CLUs and BLUs establish a semantic network of learning units, defining the declarative domain knowledge of the system. The declarative domain knowledge is stored in the form of frames in the expert system.

Counters associated to specific slots of the LUs contain information about the student's performance to each LU. The values of these counters reflect the extent to which the system believes the student has mastered each LU. Another set of counters is about motivational aspects related to the LU. Motivational knowledge deals with the effort the student shows in finding the right answers in certain types of problems and the overall confidence and independency she shows during assessment. All the above-mentioned counters compose only a part of the student model. The rest of it concerns the student's aptitudes. This kind of knowledge deals with "meta-cognitive" skills and contains declarative knowledge about the student's observed through assessment individual characteristics. Some of these characteristics are efficiency in manipulating specific CLUs, ability in calculations and estimation of the expected result etc. The concepts used in this kind of knowledge are closer to the general terms that teachers often use to evaluate students.

Aptitudes and motivational knowledge is the only kind of declarative knowledge that is accessed by the system in order to decide whether to adapt itself or not to the student's individual characteristics. The above described object oriented knowledge representation permits the existence of components and procedures that can be used from different modules of the system.

C*urriculum knowledge* is about the conceptual network of the different BLUs the problems of which the student has to solve during the assessing process. Curriculum knowledge expresses: a) the sequencing in which the problems of each BLU might be presented through assessment and b) the transformations that are permitted to take place in this sequencing in order the student to be able to follow an individualized way through the conceptual network of the domain knowledge. The curriculum knowledge is stored in a dynamic list called *presentation scenario (PS)*. The PS consists of *Curriculum Units (CUs),* each of which belongs to a different BLU. When the system adapts itself to the student the sequencing of the CUs is changed, forming this way a new PS. The PS in effect at the beginning of each assessing session is called *starting presentation scenario (SPS)* and the one in effect at each assessing moment is called *current presentation scenario (CPS).* The system provides a default PS as SPS but also a new SPS might be designed through the authoring environment of the system.

Each CU is described through a set of attributes which characterize a) the placement of the CU in the PS, b) whether a CU is important and must be presented anyway to the student or it is not important and might be bypassed, c) whether the association between the CU and the next CU at the sequencing is 'loose' and can be *broken* when the system decides to bypass the CU or to move it into another place in the PS or whether it is 'strict' and must be never changed (prerequisite CU).

3. The architecture of the system

Typically, an assessment system is an educational system that should be able to identify as quickly and as accurately as possible the gaps in the student's knowledge of the subject domain and to check the reasons for that. Computer-based assessment, in order to be able to adapt itself to the student individualities, like an expert human tutor does, should be mainly able a) to decide about how to present the assessing material and what to assess next, following the curriculum sequencing [10] and b) to take under consideration the motivational state of the student [11]. Curriculum sequencing is essentially a control path through the objective items of the assessment and is usually explicitly predicted when designing an assessment test. The design of our system, in order to overcome the rigidity of the curriculum's explicit design, supports an internal curriculum sequencing transformation mechanism, lying on the student' s aptitudes and her motivational state.

The architecture of the system has a tripartite model [3,12], which includes the standard ITS architecture as a subset (Figure 1).

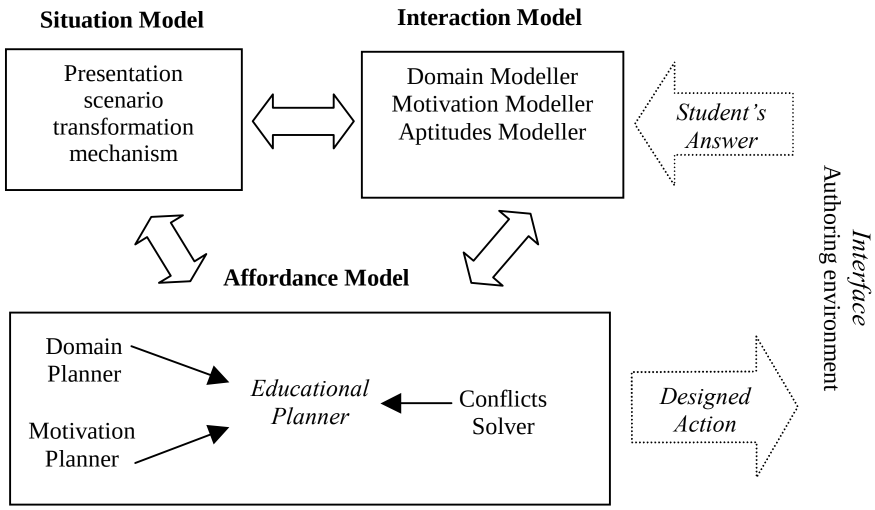

Figure 1. The architecture of the intelligent assessment system

3.1. The Affordance Model

The Affordance Model is a part of the system, which contains pedagogical expertise about the assessing strategy in use in the form of productive rules. It is a multi-part component, composed of the domain planner, the motivation planner and the conflicts solver. These three components send proposals to the educational planner who takes the final educational decisions.

The domain planner detects the current state of the learner's knowledge, according to the domain student model, and proposes the appropriate next step, traversing the curriculum sequencing in a progressive manner in the direction of the existing ultimate goal. Its main characteristic is than it is more than a typical planner, taking also into consideration student's detected weaknesses and aptitudes.

The motivation planner takes into account the student's motivational state and decides: a) about whether to advance or not in the traversal of the PS and b) the appropriate help to offer in order to maintain the motivational student's state, as suggested by Lepper, Woolverton, Mumme, & Gurtner [4].

The educational planner is responsible to reconcile output from the two first planners, to call the conflicts solver to solve any existing conflict between the domain planner and the motivation planner advices and to decide about: a) to advance or not in the PS, b) when and what kind of help to provide and c) whether to change or not the sequencing in the PS. This latter responsibility is the one that permits the system to adapt itself according to the student model, offering the student the possibility to follow an individualized way through the conceptual network of the different BLUs.

3.2. The Situation Model

The Situation Model is responsible to maintain the CPS through the assessment process. A *presentation scenario transformation mechanism (PS-TM)* is included to it in order to retrieve the knowledge about the curriculum sequencing in a specific assessment moment and to carry out the needed transformations in it during adaptation. The PS-TM realizes the ordered by the educational planner transformations, when the importance of the CUs and the semantic associations between the different CUs permit it, redirecting the 'loose' semantic associations between the different CUs of the PS.

3.3. The Interaction Model

The Interaction Model uses three different modellers, which attempt a 'closed' student modelling based on the aptitude and motivational aspects of the student's behaviour.

Table 1. Example of domain modeller's behaviour

Student's performance	*Diagnosis*	*Update of counters*	*Classification of given answer*
She gave up		increase give-up counter	
Right intermediate problem steps	Correct result		Right («belief»100%)
	Wrong result	Increase give-up counter	Right («belief»80%)
Wrong intermediate problem steps	Correct result according to the wrong steps	Further diagnosis is needed	Almost wrong («belief» will be given after studying the aptitudes model)
	Correct result according to the right steps	Increase inattention counter	Almost right («belief» will be given after studying the aptitudes model)
	Wrong result	Further diagnosis is needed	Wrong («belief»100%)

The domain modeller uses the overlay student modelling technique comparing the student's answers with the ones the system already knows as correct. In order to decide about the correctness or not of a given answer, the modeller takes also under consideration the aptitude characteristics of the student, giving according to them a percentage of belief to its judgment. The modeller analyses the student's answers in order to build a model of what the student knows. The analysis is based on the diagnostic rules, in order to classify an answer as "right", "almost right", "almost wrong" or "wrong". The student's performance in the different LUs is measured as a function of the number of presented problems and the number of each type of responses given for them, accompanied by a certainty factor expressing the belief of the system to the above estimation (see Table 1 for an example). The approach the domain-based modelling mechanism adopts is very simplistic, but a more sophisticated one without affecting the basic architecture of the system can easily replace it. In fact, we already experiment with such an approach based on fuzzy neural networks with promising results.

The motivation modeller generates its part of student model observing the student's characteristic reactions, which specify her motivational state, e.g. the requirements for help, the giving ups, her persistence to give an answer, etc. The motivation modeller focuses on three motivational aspects, as proposed by del Soldato et al. [9], namely effort (or persistence), confidence and independence. Effort refers to how a task was achieved, confidence relies mostly on the student's beliefs on his efficiency to solve the problem, and independence relies on the perceived feeling of needing or not needing help in order to complete the solution steps. Effort is the only motivational aspect that is measured separately for each BLU and CLU in the system. Confidence, effort and independence are characterized as "low", "average" or "high", incremented or decremented in large or small steps during each interaction, according to the rules of the motivation modeller (see Table 2 for an example).

Table 2. Rules of effort[1]modelling

Performance	**Steps**	**Help**	**Effort**
Gave up	none		None
	few	no	Effort to low
		yes	Decrease of effort
	many	no	Increase of effort
		yes	Effort to average
Answered		no	Increase of effort
		yes	Effort to average
Out of time	few	no	Effort to average
		yes	Effort to low
	many	no	Effort to high
		yes	Increase of effort

In our design, additionally to the domain-based modeller and the motivation modeller, a third modeller is considered based on student's general aptitudes issues, the *aptitudes modeller*. The aptitudes modeller generates its student model studying the overall student performance and attempts to identify the student's strengths and weaknesses like preferences and specially good performance at specific CLUs, ability to manipulate problems with difficult semantic structure or big numbers, ability in calculations and in estimation of the expected result etc. In Table 3, aptitudes modeller reconciles information about the student's performance, give-ups and the effort she shows and concludes about the student's preference to a specific CLU.

In next section, we will study more precisely the student modelling aspects, which guide the adaptation of the system.

[1] Effort is expressed as a function of the student's persistence to solve the presented problem even, if he has to ask for help.

Table 3. Rules of preference modeling

<table>
<tr><th>Performance</th><th>Effort</th><th>Give-ups</th><th>Preference</th></tr>
<tr><td>Very good</td><td>Average or high</td><td>None</td><td>Big</td></tr>
<tr><td rowspan="4">Good</td><td rowspan="2">Average or high</td><td>None</td><td>Big</td></tr>
<tr><td>Few</td><td>Medium</td></tr>
<tr><td rowspan="2">Low</td><td>None</td><td>Medium</td></tr>
<tr><td>Many</td><td>Little</td></tr>
<tr><td rowspan="2">Quite good or bad</td><td>Average or high</td><td></td><td>Little</td></tr>
<tr><td>Low</td><td></td><td>No preference at all</td></tr>
</table>

4. 'Motivating' Aspects of the System

Motivation is offered to the student in two different ways.

First, the motivation planner takes into account the student's motivational characteristics and advises the educational planner about whether: a) to offer help or not, in order to maintain the motivational student's state and b) to advance or not in the traversal of the CPS.

Table 4. Examples of decisions of the conflicts solver

Proposals of Domain modeller	Proposals of motivation planner	Decisions of Conflict solver
The student performs well. Present a more difficult type of problem	Present again the same type of problem in order to help the student become more confident	Present a new problem of the same type of problems
The student performs badly. Present a new problem of the same type of problems	Present again the same type of problems in order to help the student to become more confident	Present a new problem of the same type of problems giving the student hints to overcome her bad performance
Wrong answer to the problem. Give help based on the diagnosis of the mistake and present a more difficult type of problems	The student must become more confident, experiencing the feeling of success	Present a new problem of the same type of problems giving the student information about the mistake she did in the last problem
Wrong answer to the problem. Give help based on the diagnosis of the mistake	The student must become more independent	Present a motivating message instead of help.

For example, the motivation planner proposes to insist in presenting the same type of problems when the student has very low confidence and needs to be encouraged. Solving the same problem and succeeding to give a right solution will help the student to experience the feeling of success and become more confident. The planner, in order to help the student to improve her independence and persistence, proposes the refusal of a help request urging her to try harder. When there are conflicts between the proposals of the motivation planner and those of the domain planner, conflicts solver is asked to take the final decisions (see Table 4 for some examples).

Second, the educational planner decides whether to adapt or not the sequencing in the CPS to the student's aptitudes and motivational state. That is, the educational planner decides to re-sequence the CUs in the CPS when the student shows a big preference in problems belonging to a specific CLU and in the same time he has a very low overall performance and confidence, and needs to be encouraged. Then, all the CUs belonging to BLUs of this CLU are grouped and moved in the first places of the rest of the sequencing that has to be presented. For example, if there are types of problems (BLUs) that have as context (CLU) 'money' or 'volume' and the low-confident student performs much better in solving of problems of money than of volume, then from now and then all types of problems that have as context 'money' will always be presented before the equivalent types of problems having as context 'volume'.

5. Evaluation of the prototype

A prototype assessment system has been implemented based on the design model presented in this paper. For the implementation, we used the GC Lisp V programming language, GoldWorksIII expert system environment (from GoldHill Inc.) and Visual Basic VI (from Microsoft). The system, called ASSA (Adaptive System of Student Assessment), assesses the ability of low-attaining pupils to solve simple word arithmetic problems of addition and subtraction and runs on Windows95/98 (from Microsoft).

Two experts in the design of educational software, three teachers of elementary school, five expert teachers in special education and a cognitive scientist have attempted a preliminary evaluation of the prototype system. The evaluation process used here is more akin to the evaluation of Expert Systems, involving the empirical testing of the knowledge base against the judgmental accuracy of experts and ground-truth measures of accuracy. The evaluators experimented with the system in the laboratory under the designers' supervision. Only three pupils have been used at this preliminary evaluation stage. In next paragraphs, we present the followed evaluation procedure and the evaluation results, which concern mainly the reliability of these aspects of the student model, which guide the adaptation of the system, the accuracy of the provided by the system student profiles, produced by different performance histories and the stability of the assessment results.

During evaluation, the adaptive characteristics of the system and the student modelling aspects, which guide the adaptation, were firstly explained in details to the expert evaluators. Next, the evaluators were urged to design a presentation scenario of their convenience and to use it as a SPS for all their subsequent experiments. Then, the subjects were asked to experiment with the assessing process in several sessions, producing different characteristic performances and motivation behaviours at each of them. For each experiment, the student profile was demonstrated to the experts, who had to comment about its accuracy. When the experiments were over the subjects were asked to remark about the efficiency of the motivational factors used in student modelling, the reliability of the measurements of student's individual characteristics (motivational and aptitudes) and the stability of the assessment results.

All evaluators approved the overall adaptive performance of the system and the reliability of the measurements during student modelling. Nevertheless, those experts, who studied the system from the cognitive point of view, reserved themselves to denote convinced about the ability of the system to perform a thorough assessment aiming at a detailed student model. They asked for a more detailed diagnosis of wrong answers and a more appropriate design of hints and provided help. The cognitive scientist expressed her doubts about the efficiency of the motivational factors used in student modelling and argued that knowledge about motivation diagnosis may be elicited based on theories of motivation, observations or 'common sense', but in order to test the validity of this knowledge a number of experiments must be devised.

Although the number of pupils was very small in order to draw accurate conclusions, the assessment results were proved to be stable, remaining always the same when they were produced by similar assessment histories and the produced student profiles have been characterized as accurate, according to the rules of the three distinct modellers and the motivational factors in use. Our intent is to proceed with an evaluation in real class conditions, using a satisfactory number of pupils in order to obtain more accurate results.

6. Conclusion/Future Work

This paper presents the design of an adaptive to the student's individualities and motivational state assessment system, based on expert system and student modelling techniques. The system is able to detect the current state of the student's achievement, her aptitude characteristics, her motivational state, and react with the purpose of adapting the curriculum sequencing to the student's individual strengths and motivational characteristics.

During assessment the system tackles the objectives to be assessed, in a systematic way according to their sequencing in the presentation scenario, which represents the assessment curriculum. The presentation scenario sequencing might be dynamically changed according to the educational rules and the aptitude and motivational characteristics of the student. At the end of each assessment session, the different kinds of knowledge, which derives from the firing of the rules, are formatted and provided to the teacher as the student's learning style profile. This kind of information is believed to be very useful to the teachers especially to those who have not enough time or experience to follow a similar assessment procedure.

The system has been validated by a number of experts with promising results. First results have shown that the system is able to adapt its assessing strategy to the student's cognitive strengths and to implement motivational tactics in a satisfactory level. Although the implementation of a domain-based student model was not of first priority in our work the need of a more efficient student modelling method emerged and the need of the enrichment of the pedagogical rules, concerning the provision of help was obvious too.

Resulting from the evaluation results, our future work will be to extend our research in the area of fuzzy neural networks for student modelling. This method has been proved to be efficient enough, offering successful student models. We intend to implement it in our prototype system and to study the differences between the new and the old version of ASSA in relation to the produced student profiles and the corresponding adaptive behaviors. Next move will be to adopt the most efficient of the two student modeling methods and design a new web-based assessment system in a more sophisticated domain.

References

1. du Boulay, B., Luckin, R., and del Soldato, T.: The Plausibility Problem: Human Teaching Tactics in the 'Hands' of a Machine. International Conference on Artificial Intelligence in Education: Open Learning Environments: New Computational Technologies to Support Learning, Exploration and Collaboration, Le Mans, France, IOS Press (1999).
2. Koedinger, K., Anderson J.: Intelligent Tutoring Goes to School in the Big City". *International Journal of Artificial Intelligence in Education* 8 (1997) 30-43.
3. Self, J.: The defining characteristics of intelligent tutoring systems research: ITSs care, precisely. *International Journal of Artificial Intelligence in Education 10 (1999) 350-364.*
4. Lepper, M. R., Woolverton, M., Mumme, D., & Gurtner, J.: Motivational techniques of expert human tutors: Lessons for the design of computer-based tutors. Computers as cognitive tools. S. P. Lajoie, & S.J Derry, (eds), Hillsdate, NJ: Erlbaum Associates (1993) 75-105.
5. Aubé, M.: Towards Computational Models of Motivation: A much needed foundation for social sciences and education. *International Journal of Artificial Intelligence in Education*, 8(1) (1997) 43-75.
6. Viau, R.: La motivation en contexte scolaire, Saint-Laurent:ERPL(1994).
7. Keller, J. M.: Strategies for stimulating the motivation to learn. *Performance and Instruction Journal*, 26 (1987) 1-7.
8. de Vicente, A., & Pain, H.: Motivation Diagnosis in Intelligent Tutoring Systems. Proceedings of *ITS '98,* Springer-Verlag, (1998) 86-95.
9. del Soldato, T., & du Boulay, B.: "Implementation of Motivational Tactics in Tutoring System". *International Journal of Artificial Intelligence in Education* 6 (4) (1995). 337-wing378.
10. Gouli, E., Kornilakis, H., Papanikolaou, K., Grigoriadou, M.: Adaptive Assessment improving interaction in an Educational Hypermedia System. In: Avouris, N., Fakotakis, N. (eds): Advances in Human-Computer Interaction I. Proceeding of the Pan Hellenic Conference with International Participation in Human-Computer Interaction, (2001) 217-222.
11. Dowling, C., and Kaluscha, R.: Prerequisite Relationships for the Adaptive Assessment of Knowledge. Artificial Intelligence in Education. G. J. Charlottesville, V.A. Association for the Advancement of Computing in Education (AACE) (1995) 43-50.
12. Prentzas J., Hatzilygeroudis I. and Koutsojannis C.: A Web-Based ITS Controlled by an Expert System, Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT 2001), Madison, Wisconsin, USA (2001) 239-40.