Individualizing a cognitive model of students' memory in Intelligent Tutoring Systems

Maria Virvou, Konstantinos Manos

Department of Informatics
University of Piraeus,
Piraeus 18534, Greece
mvirvou@unipi.gr; konstantinos@kman.gr

Abstract: Educational software applications may be more effective if they can adapt their teaching strategies to the needs of individual students. Individualisation may be achieved through student modelling, which is the main practice for Intelligent Tutoring Systems (ITS). In this paper, we show how principles of cognitive psychology have been adapted and incorporated into the student modelling component of a knowledge-based authoring tool for the generation of ITSs. The cognitive model takes into account the time that has passed since the learning of a fact has occurred and gives the system an insight of what is known and remembered and what needs to be revised and when. This model is individualised by using evidence from each individual student's actions.

Introduction

The fast and impressive advances of Information Technology have rendered computers very attractive media for the purposes of education. Many presentation advantages may be achieved through multimedia interfaces and easy access can be ensured through the WWW. However, to make use of the full capabilities of computers as compared to other traditional educational media such as books, the educational applications need to be highly interactive and individualised to the particular needs of each student. It is simple logic that response individualized to a particular student must be based on some information about that student; in Intelligent Tutoring Systems this realization led to student modeling, which became a core or even defining issue for the field (Cumming & McDougall 2000).

One common concern of student models has been the representation of the knowledge of students in relation to the complete domain knowledge, which should be learnt by the student eventually. Students' knowledge has often been considered as a subset of the domain knowledge, such as in the overlay approach that was first used in (Stansfield, Carr & Goldstein 1976) and has been used in many systems since then (e.g. Matthews et al. 2000). Another approach is to consider the student's knowledge in part as a subset of the complete domain knowledge and in part as a set of misconceptions that the student may have (e.g. Sleeman et al. 1990). Both cases represent an all or nothing approach on what the student knows and they certainly do not take into account the temporal aspects of knowledge, which are associated with how students learn and possibly forget.

In view of the above, this paper describes the student modeling module of an educational application. This module can measure-simulate the way students learn and possibly forget through the whole process of a lesson. For this purpose, it uses principles of cognitive psychology concerning human memory. These principles are combined with evidence from the individual students' actions. Such evidence reveals the individual circumstances of how a student learns. Therefore the student model takes into account how long it has been since the student has last seen a part of the theory, how many times s/he has repeated it, how well s/he has answered questions relating to it.

As a test-bed for the generality of our approach and its effectiveness within an educational application we have incorporated it in a knowledge based authoring tool. The authoring tool is called Ed-Game Author (Virvou et al. 2002) and can generate ITSs that operate as educational games in many domains.

The cognitive model of memory for an average student

A classical approach on how people forget is based on research conducted by Herman Ebbinghaus and appears in a reprinted form in (Ebbinghaus, 1998). Ebbinghaus' empirical research led him to the creation of a mathematical formula which calculates an approximation of how much may be remembered by an individual in relation to the time from the end of learning (Formula 1).

$$b = \frac{100 * k}{(\log t)^c + k} \qquad (1)$$

Where:

- t: is the time in minutes counting from one minute before the end of the learning
- b: the equivalent of the amount remembered from the first learning.
- c and k : two constants with the following calculated values: k = 1.84 and c = 1.25

In the student model of Ed-Game Author the Ebbinghaus calculations have been the basis for finding out how much is remembered by an average student. In particular, there is a database that simulates the mental library of the student. Each fact a student encounters during the game-lesson is stored in this database as a record. In addition to the fact, the database also stores the date and time when the fact was last used, in a field called LastAccessDate. A fact is first added to the memory database when a student is first taught this fact through a lesson. When a fact is inserted in the database, the current date and time is also recorded in the field called TeachDate.

Thus, whenever the system needs to know the current percentage of a student's memory of a fact, the equation (2) is used, which is largely based on the Ebbinghaus' power function. However, equation (2) has been adapted to include one more factor, which is called the Retention Factor (RF). The retention factor is used to individualise this equation to the particular circumstances of each student by taking into account evidence from his/her own actions.

If the system does not take into account this evidence from the individual students' actions then the Retention Factor may be set to 100, in which case the result is identical to the generic calculations of Ebbinghaus concerning human memory in general. However, if the system has collected sufficient evidence for a particular student then when a fact is first encountered by this student the Retention Factor is set to 95 and then it is modified accordingly as will be described in detail in the following sections. The RF stored in the "mental" database for each fact is the one representing the student's memory state at the time showed by the TeachDate field.

$$X\% = \frac{b}{100} * RF \quad (2)$$

Where b is the Ebbinghaus' power function result, setting t=Now-TeachDate.

Memorise Ability

One important individual student characteristic that is taken into account is the ability of each student to memorise new facts. Some students have to repeat a fact many times to learn it while others may remember it from the first occurrence with no repetition. To take into account these differences, we have introduced the student's Memorise Ability factor (MA). This factor's values range between 0 and 4. The value 0 corresponds to "very weak memory", 1 to "weak memory", 2 to "moderate memory", 3 to "strong memory" and 4 to "very strong memory".

During the course of a virtual-game there are many different things that can give an insight on what the student's MA is. One important hint can be found in the time interval between a student's having read about a fact and his/her answering a question concerning this fact. For example, if the student has given a wrong answer about a fact that s/he has just read about then s/he is considered to have a weak memory. On the other hand if s/he gives a correct answer concerning something s/he had read about a long time ago then s/he is considered to have a strong memory.

Taking into consideration such evidence, the student's MA value may be calculated. Using MA the Retention Factor is modified according to the MA value of the student in the way illustrated in Table 1. As mentioned earlier, every fact inserted in the database has an initial RF of 95.

Table 1: Retention Factor's modification depending on the Memorise Ability

Memorise Ability Value	**Retention Factor Modification**
0	RF\` = RF – 5
1	RF\` = RF – 2
2	RF\` = RF
3	RF\` = RF + 2
4	RF\` = RF + 5

After these modifications the RF ranges from 90 (very weak memory) to 100 (very strong memory), depending on the student's profile. Taking as a fact that any RF below 70 corresponds to a "forgotten" fact, using the Ebbinghaus' power function, the "lifespan" of any given fact for the above mentioned MA may be calculated. So a student with a very weak memory would remember a fact for 3 minutes while a student with a very strong memory would remember it for 6.

Response Quality

During the game, the student may also face a question-riddle (which needs the "recall" of some fact to be correctly answered). In that case the fact's factor is updated according to the student's answer. For this modification an additional factor, the Response Quality (RQ) factor, is used. This factor ranges from 0 to 3 and reflects the "quality" of the student's answer. In particular, 0 represents "no memory of the fact", 1 represents an "incorrect response; but the student was close to the answer", 2 represents "correct response; but the student hesitated" and 3 represents a "perfect response". Depending on the Reponse Quality Factor, the formulae for the calculation of the new RF are illustrated in Table 2.

Table 2: Response Quality Factor reflecting the student's answer's quality

RQ	Modification
0	RF' = X – 10, where TeachDate=Now
1	RF' = X – 5, where TeachDate = Now
2	RF' = RF + (MA + 1) * 3
3	RF' = RF + (MA + 1) * 4

When a student gives an incorrect answer, the TeachDate is reset, so that the Ebbinghaus' power function is restarted. When a student gives a correct answer, the increase of his/her Retention Factor depends on his/her profile and more specifically on his/her Memorise Ability factor. In particular, if the student's RQ is 2 and s/he has a very weak memory then the RF will be increased by 3 points (extending the lifespan of the memory of a fact for about a minute) while if s/he has a very strong memory the RF will be increased by 15 (extending its lifespan for over 6 minutes). These formulae for the calculation of the RF give a more "personal" aspect in the cognitive model, since they are not generic but based on the student's profile.

In the end of a "virtual lesson", the final RF of a student for a particular fact is calculated. If this result is above 70 then the student is assumed to have learnt the fact, else s/he needs to revise it.

Conclusions

In this paper we described the part of the student modelling process of an ITS authoring tool that keeps track of the students' memory of facts that are taught to him/her. For this reason we have adapted and incorporated principles of cognitive psychology into the system. As a result, the educational application takes into account the time that has passed since the learning of a fact has occurred and combines this information with evidence from each individual student's actions. Such evidence includes how easily a student can memorise new facts and how well she can answer to questions concerning the material taught. In this way the system may know when each individual student may need to revise each part of the theory being taught.

References

Cumming G. & McDougall A.: Mainstreaming AIED into Education? *International Journal of Artificial Intelligence in Education,* Vol. 11, (2000), 197-207.

Ebbinghaus, H. (1998) "Classics in Psychology, 1885: Vol. 20, Memory", R.H. Wozniak (Ed.), Thoemmes Press, 1998

Matthews, M., Pharr, W., Biswas G. & Neelakandan, (2000). "USCSH: An Active Intelligent Assistance System," *Artificial Intelligence Review* 14, pp. 121-141.

Sleeman, D., Hirsh, H., Ellery, I. & Kim, I. (1990). " Extending domain theories: Two case studies in student modeling", *Machine Learning,* 5, pp. 11-37.

Stansfield, J.C., Carr, B., & Goldstein, I.P. (1976) Wumpus advisor I: a first implementation of a program that tutors logical and probabilistic reasoning skills. At Lab Memo 381. Massachusetts Institute of Technology, Cambridge, Massachusetts.

Virvou M., Manos C., Katsionis G., Tourtoglou K.(2002)**,** "Incorporating the Culture of Virtual Reality Games into Educational Software via an Authoring Tool", *Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC 2002, Tinisia).*