Support Cognitive Processes in Intelligent Tutoring Systems

Alke Martens (martens@informatik.uni-rostock.de)
Department of Computer Science, University of Rostock,
Albert-Einstein-Str.21, 18059 Rostock, Germany

Abstract

Computer based learning with Intelligent Tutoring Systems (ITS) has become a meaningful supplement to traditional lectures at school and at universities. A highly flexible and adaptive tutoring process model is required as a central component of ITS to adequately support cognitive processes of the learner, e.g. the process of knowledge application and the process of diagnostic reasoning. Whereas most of the ITS focus only the second aspect, the tutoring process model provides facilities to support both processes. It continuously and dynamically adapts the training case’s contents and the available set of actions to a learner model and to the training case’s development.

1. Introduction

Computer based learning provides a meaningful supplement to traditional lectures and courses. Located at the borderline of Artificial Intelligence (AI) and Cognitive Science, Intelligent Tutoring Systems (ITS) have always combined insights of both disciplines to construct learning environments that are able to adequately support cognitive processes in knowledge acquisition. Having its roots in computer science, our work adopts insights from cognitive science to develop a flexible and adaptive tutoring process model. It is tested in the ITS ‘Docs ’n Drugs’ (Martens & Bernauer & Illmann, 2001; Docs ’n Drugs, 2000). Application domain is case-based learning in clinical medicine.

The tutoring process model modifies the classical ITS architecture: it is the new centre of the ITS as it is responsible for interaction with the other components and with the learner. It steers and controls the flow and adaptation of the training case - hand in hand with a learner model that continuously protocols the learner’s progress. The tutoring process model, realized as a formal model, is domain independent and highly flexible and thus not restricted - neither to case-based learning nor to clinical medicine.

In the following we will provide a short look into case-based learning. The tutoring process itself and its role in the ITS is described in the third section. An approach is sketched that allows making assumptions about the learner’s progress based on the tutoring process model.

2. Case-based Learning

At the University of Ulm, case-based learning with a web-based ITS is integral part of the medical curriculum. Cognitive processes in medicine have been focus of research, e.g. in the work of Patel, Kaufman and Arocha (1995). They have concluded that two main cognitive processes are distinguishable in medical work, i.e. the process of knowledge application and the process of diagnostic reasoning. Whereas Patel described these two processes as vantage points for research in cognitive science, we have adopted them for case-based learning. Our assumption is that, according to Patel et.al., two main cognitive processes should be supported with a case-based ITS. The learner should have the possibility to train the process of knowledge application in the training domain, i.e. choosing appropriate steps in a correct order in close to real life situations; and using the acquired knowledge to draw and continuously adapt his diagnoses.

A closer look at other domains, e.g. law, biology and modeling and simulation (Martens & Uhrmacher, 2001), has led us to the assumption that the same two processes should also be supported in case-based training in these domains. In problem-based learning, e.g. the ITS Andes (VanLehn et.al. 2002), similar assumptions have been drawn.

3. Tutoring Process Model and Tutor Agent

The classical architecture ITS consists of: the expert knowledge model, the pedagogical knowledge model and a learner model. Sometimes a diagnosis model and an exercise generator are embedded. The framework of interacting models must be complemented by a set of components responsible for steering the flow and adaptation of the contents. These components are application specific and not reusable. In our work, we have changed the focus of the architecture towards a centralized tutoring process model (Volz & Martens & Seitz, 2002). This tutoring process model interacts with the classical ITS models, it is responsible for adaptation – it substitutes the application specific components.

A formal tutoring process model consists of a set of contents and a set of actions. From the perspective of the learner, these are the displayed pages and the actions in a navigation bar, respectively. In a medical training case, the page provides the learner with information about the patient or about results of examinations, whereas the actions denote steps in the treatment process. Contents and actions contain preconditions and effects. Contents as well as the amount of actions available in each step are adapted to the information in the learner model. Every time the learner selects an
action, the learner model protocols it. The tutoring process adapts the contents of the next display, protocols the according effects and selects the set of next possible actions.

In the training case, two kinds of interactions are possible for the learner. One is the selection of next actions to perform. The other is a kind of classical checkpoint, e.g. marking regions in pictures or answering questions. Reasoning about the learner’s skills in the classical checkpoints is straightforward. Reasoning about the sequence of actions is more complicated. In case-based training scenarios in clinical medicine, no one ‘optimal’ action exists, but a set of actions, all of which are equally correct. Thus, the learner should not be guided back to an ‘optimal’ path in the training case, but in contrast should have the possibility to see the consequences of his steps. This can be realized with the highly flexible structure of the contents and actions and the protocol.

Help or feedback should also take place in an indirect manner. Via restricting the amount of actions, the tutoring process provides an implicit support. This is additionally realized via decreasing or increasing the difficulty of the displayed pages. A wrong decision in the treatment process, i.e. a wrong action chosen by the learner, will lead to an adapted content, depending on the effect of the wrong decision. Nonetheless, the training case should be logically coherent.

Whereas the tutoring process provides the main features for an adaptive training case, it lacks a component for reasoning. The problem here is that different level of erroneous decisions has to be distinguished and require different kind of evaluation or help. We introduce a tutor agent into the ITS (Martens & Uhrmacher, 2002). The tutor agent has knowledge about the optimal path in the training case. It is equipped with planning facilities and has access to a plan library. Using the precondition-effect structure of the actions, small sub-plans can be generated and stored in the plan library. The tutor agent uses these sub-plans to evaluate the learner’s progress. It is able to distinguish different kinds of erroneous decisions made by the learner. Its observations can be used for a final evaluation and for a proactive behavior, if the learner continuously makes mistakes in the treatment process.

4. Conclusion

In this paper, we have assumed that two cognitive processes should be supported in case-based learning. According to Patel, Kaufman and Arocha (1995), in clinical medicine, the cognitive processes of knowledge application and of diagnostic reasoning can be distinguished. Analysis of case-based training scenarios in some other application domains has led us to the assumption, that these two processes also occur in other areas and thus should generally be supported in case-based training. Whereas most ITS in ill structured domains like clinical medicine only focus the training of diagnostic reasoning, we have developed a tutoring process model that allows to support both processes.

Being highly flexible, the tutoring process model allows the dynamic adaptation of training cases to the learner during the learner’s work with the training case. With the combination of tutoring process and tutor agent, we are able, to reason about the learner’s progress and his success in the training case’s contents, with respect to his progress in knowledge application and also with respect to his progress in diagnostic reasoning.

Main drawback of our approach is the large amount of authoring work that is required to provide this kind of training cases. The realization of an appropriate authoring system for ‘Docs ´n Drugs’ currently takes place. However, it remains a challenge, if all the mentioned features shall be realized. The tutoring process model can be re-used in the context of authoring, e.g. as a feature for checking the training case’s consistency. The tutor agent might change role to support authoring. Both aspects, however, have not been realized yet.

An evaluation of ‘Docs ´n Drugs’ is currently taking place.

References


