

A WEB-BASED INTELLIGENT TUTORING SYSTEM

Zhiping Li, Yu Sun and Minkun Liu

¹Computer Center of Yunnan Normal University, ^{2,3}Institute of Computer Science and Information Technology, Yunnan Normal University, 650092, Kunming, China

Abstract: A formal model of a web-based intelligent tutoring system composed of a user environment and a pedagogical environment is presented, which represents domain knowledge based on ontologies to improve the sharing and reusing of teaching materials. The system constructs the user environment based on users' knowledge levels, learning styles, psychology characteristics, etc. in order to improve the self-adaptability and pedagogical effects of the system. Furthermore, it distinguishes information about a user and what a pedagogical agent knows about the user. Based on the pedagogical agent's knowledge (represented by its cognitive state) about the user, a teaching process is designed for the user. Finally, the running process of the system is discussed in detail to show that the model is practicable.

Key words: intelligent tutoring system, ontologies, web, domain knowledge

1. INTRODUCTION

Intelligent Tutoring Systems (ITSs) is a branch of artificial intelligence and form an advanced generation of Computer Aided Instruction systems [1,2]. The key feature of intelligent tutoring systems is the ability to provide a user self-adapted presentation of the teaching material [9] by using artificial intelligence methods to represent the pedagogical decisions and the information about each user. The emergence of the WWW has increased the applications of such systems [3,4].

Although some tutoring systems have correct information and appropriate teaching strategies, a learner might still lack the motivation to

complete a learning process [10]. The main reason is that these systems apply the same approach to all learners and do not consider users' different knowledge levels, learning styles, and demands. Our tutoring system constructs the user model based on users' knowledge levels, learning styles, psychology characteristics, etc. in order to improve the self-adaptability and pedagogical effects of the system. Furthermore we use ontologies for domain knowledge representation to improve the sharing and reusing of materials; and use multimedia materials obtained from web to improve teaching effects.

Even if some systems take a user's characteristics (e.g. knowledge levels, learning styles, etc.) and needs into account, a gap between the estimation of users' characteristics and the users' real characteristics may make these systems fail to reach the desired effects. To describe this gap, we distinguish information about a user and what a pedagogical agent knows about the user. The pedagogical agent transforms its perceptions about the user (e.g. knowledge levels, learning styles, etc.) into its knowledge. If this transformation is complete (no information loss), then there is no gap between the pedagogical agent's cognition of a user's characteristics and the user's real characteristics. Based on the knowledge about a user, a pedagogical agent designs a teaching process for the user.

The paper is organized as follows. Section 2 presents an overview of the system's architecture. Section 3 presents the user environments formalism of the system. Section 4 presents the representation of domain knowledge and the pedagogical environment formalism of the system. Section 5 describes the runs of the system in detail to show that the model is practicable. Finally, section 7 concludes the paper with conclusion and future works.

2. A WEB-BASED INTELLIGENT TUTORING SYSTEM

In this section, we propose the formal model of a web-based intelligent tutoring system showed in figure 1. The model consists of two parts: user environment (*PE*) and pedagogical environment (*PE*).

Our intelligent tutoring system contains one pedagogical agent with a set of cognitive states which records its cognition or knowledge of the user agent and one user agent with a set of user states (user models) recording basic information such as personal data, user characteristics, etc. about the user. The user agent updates its state after finishing a learning process.

The cognitive process of the pedagogical agent include two steps: (1) dividing user states into disjoint set; the pedagogical agent cannot distinguish two user states in a same partition; and for two user states in a same partition, the pedagogical agent adopt the same teach process; (2)

perceiving the user state; the perceptions of two user states in a same partition is the same. Then, according to what perceived from a user state, the pedagogical agent transforms the perceptions into its knowledge and changes its cognitive states. Since there may be information loss in the perceiving step and transformation process, what the pedagogical agent knows about a user state cannot completely reflect the user state.

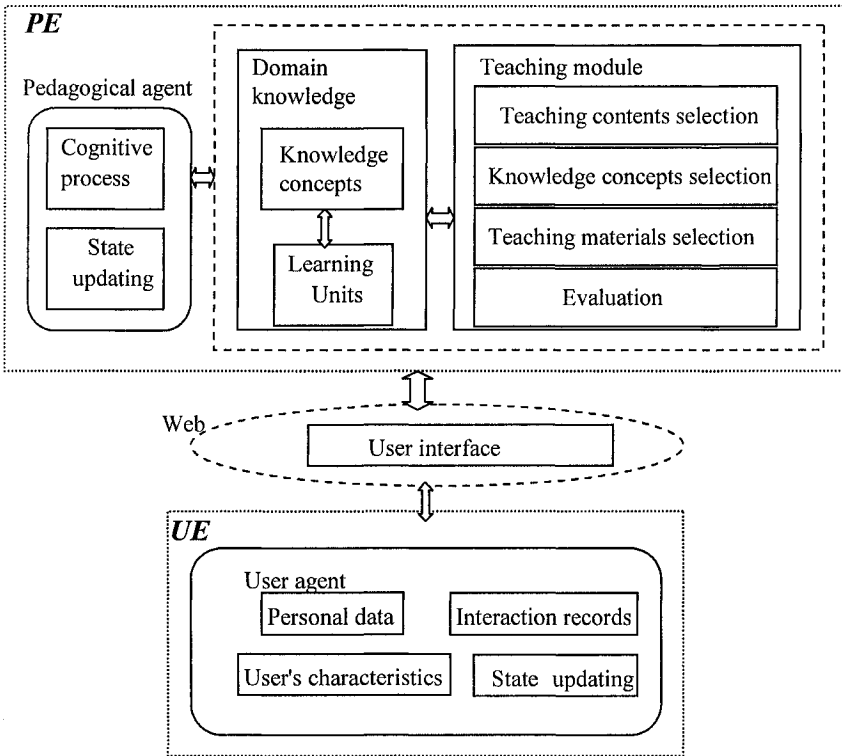


Figure 1: Intelligent Tutoring System

Finally, the pedagogical agent makes use of the domain knowledge and teaching module to design a teaching process based on its cognition or knowledge about the use agent. We discuss in detail the user environment and pedagogical environment in the following sections.

3. USER ENVIRONMENT

In this section, we introduce the user environment (*PE*) and give its formal definition.

Comparing with traditional pedagogical system, the most salience character of ITSs is self-adaptability [5]. ITSs construct user model based on

users' knowledge levels, cognitive abilities, psychology characteristics, etc. and then makes a teaching plan for each user according to its user model. Furthermore, ITSs can instruct a user in person with pertinence; and adjust teaching contents and strategies dynamically in the process of teaching in order to optimize teaching materials and strategies and to improve the self-adaptability and pedagogical effects of the system.

The goal of user model is to guide intelligent tutoring systems to adopt the most suitable teaching content and strategies. Most of web-based systems cannot adjust learning materials on demands [6]. At the same time, even if there is correct information, users may lack the motivation to accomplish the whole learning process, for example, users may dislike current learning styles. These systems haven't considered that users have different levels in knowledge levels, learning styles and demands. Hence, they adopt a same teaching content and teaching methods for all the users.

Based on multiple domains and multiple applications, we study user environment in terms of learning styles, knowledge levels, learning motivations and psychology characteristics, etc.

Formally, a user environment UE is defined as follows:

$UE = \langle U, L, I, vis, perc, trans, inter, update \rangle$, where

- $U = \{u_1, u_2, \dots\}$ is a set of user states; each u represents a user model;
- $L = \{l_1, l_2, \dots\}$ is a set of cognitive states of the pedagogical agent which represent the internal structure of the pedagogical agent, that is, the knowledge of the pedagogical agent about the user agent;
- $I = \{i_1, i_2, \dots\}$ is a set of interaction states of a user agent with an intelligent tutoring system; interaction states contain information recorded from the interaction between a user and a system such as things like, the type and amount of help asked, the correct and wrong answers to exercises, the marks obtained from exercises, etc.
- $vis: U \rightarrow 2^U$, is a user state partition function; it divides the elements in U into disjoint sets such that $\forall u \in U$, we have $u \in vis(u)$. We say vis is transparent if $\forall u \in U$, $vis(u) = \{u\}$, that is, every user state is visible to a pedagogical agent; a pedagogical agent cannot distinguish two user states if they belongs to one partition, that is, it regards the two user states to be the same;
- $perc: vis(U) \rightarrow P$, is a user state perception function, where the set of user state perceptions P consists all the information that is perceived by a pedagogical agent in a user state; we say $perc$ is perfect if it is an injection, that is, a pedagogical agent can perceive all the information that is visible to it;
- $trans: L \times Perc \rightarrow L$, is a cognitive state transformation function; it changes the cognitive state of a pedagogical agent according to its perception of a user state, that is, after a perception process of a user

state, a pedagogical agent translates what it perceives into knowledge and update its cognitive state; if $trans(l, p) = trans(l', p')$ when $p = p'$, then $trans$ is complete, that is, there is no information loss;

- $inter: I \rightarrow I$, is an interaction state transformation function; it updates the interaction records of a user agent with a system;
- $update: U \times I \times E \rightarrow U$, is a user state updating function; it changes the state of a user agent according to the pervious user state; the evaluation of the user's performance in a learning process (of a teaching content); and the interaction records between the user and the system.

A user state used to record information about the user is vital for the system's user-adapted operation. It includes the following information:

(1) Personal data: The personal data concerns information necessary for the creation and management of the user's account (e.g. name, email). It is used for the identification of the user.

(2) User's characteristics: cognitive abilities (e.g., attention, memory, etc.); learning styles (e.g., preferences of multimedia type such as text, image, etc.); experience concerning the use of computer and intelligent tutoring system and availability of Internet resources; expectations (e.g. the knowledge and skill that a use desires to learn; and teaching strategies preferences.

(3) Knowledge level such as novice, beginner, intermediate, advanced, etc. of the sub-domains and the whole domain.

(4) Interaction records which record the interaction of a user with a system.

User characteristics and knowledge levels directly affect the teaching process whereas most of the interaction information indirectly.

4. PEDAGOGICAL ENVIRONMENT

In this section, we discuss the pedagogical environment (PE) and give its formal definition.

The pedagogical environment provides the knowledge infrastructure in order to tailor presentation of the teaching material according to the cognitive states of the pedagogical agent. It dynamically produces personalized teaching web pages (learning units) to optimize teaching effect.

First, we discuss the representation of domain knowledge based on ontologies. The introduction of ontologies is to overcome the general problem in existing teaching systems [7]. At the same time, web introduces rich multimedia teaching materials by texts, images, audios, videos, examples, etc. to make the dynamic interaction between systems and users and between users available [8]. Besides effective communication, dynamic interaction improves the abilities of problem solving and learning.

Hence, we construct the representation frame of domain knowledge based on ontologies to improve the sharing and reusing of teaching materials; and we fuse the rich teaching materials provided by web into the frame. We adopt three-level frame to represent domain knowledge: subject level (the subject content of learning and teaching); knowledge concept level (knowledge concepts and the relations between these concepts such as part-of, is-a and prerequisite, etc.); and teaching material level (multimedia materials, e.g. text materials, image materials, example materials, exercise materials, video materials and audio materials). In the above three level frame, a subject corresponds to a set of knowledge concepts used to teach the subject; a knowledge concept corresponds to a set of multimedia materials used to teach the concept. We extract part of the materials from the multimedia materials according to user's learning styles, etc. and organize these materials based on some specific teaching strategies to form a learning unit of the knowledge concept. This learning unit is presented to users when teaching the knowledge concept.

Due to the differences among users' teaching strategy preferences, learning styles, memory and knowledge levels, etc. and the richness of materials, same knowledge concept may correspond to several learning units.

Then, we discuss the process of teaching. In the process of teaching, the pedagogical agent must perform the following tasks: (i) Select teaching contents (sets of knowledge concepts) to be taught according to the cognitive state of the pedagogical agent; (ii) Select a knowledge concept that will be taught currently according to the relations between the knowledge concepts (of a teaching content) and the evaluation of the learning effect of the previous knowledge concept; (iii) Select a learning unit to be presented from the set of learning units corresponding to the knowledge concept selected in (ii) according to the cognitive state of the pedagogical agent; (iv) Evaluate the user's performance and update the user's state.

Selection of a set of knowledge concepts in (i) is based on the user's knowledge of the domain, links between knowledge concepts, and user's desired detail level of the presented educational content. Evaluation of the user's performance updates the inferable user characteristics and may create a feedback for tasks (iii) and (iv).

From what has been discussed above, we give the definition of pedagogical environment $PE = \langle K, C, E, csel, ksel, teach, eval \rangle$, where,

- $K = \{k_1, k_2, \dots\}$ is a set of knowledge concepts;
- $C = \{c_1, c_2, \dots\}$ is a set of learning units;
- $E = \{e_1, e_2, \dots\}$ is a set of evaluation states of learning units;
- $csel: L \rightarrow 2^K$, is a teaching content selection function; it selects the set of knowledge concepts to be taught according to the cognitive state of the

pedagogical agent (especially about user's domain knowledge levels and expectations);

- $k\text{sel}: L \times E \times c\text{sel}(L) \rightarrow K$, is a knowledge concept selection function; it selects the most suitable knowledge concept to be taught currently according to the cognitive state of the pedagogical agent; the relationships between knowledge concepts and the evaluation of the learning effect of the previous taught knowledge concept, for example, in teaching process, if there is a priority order between two knowledge concepts, then the system will not teach the user the next knowledge concept unless the evaluation of the learning effect of the previous taught knowledge concept reaches the lowest standard.
- $\text{teach}: L \times K \rightarrow C$, is a learning unit selection function; it selects the most suitable learning unit to be presented according to the cognitive state of the pedagogical agent (about user's cognitive abilities, learning styles, knowledge levels, etc.) and the knowledge concept to be taught;
- $\text{eval}: L \times I \times C \rightarrow E$, is a learning unit evaluation function; it evaluates user's performance in a learning unit according to the cognitive state of the pedagogical agent; the learning unit to be taught; and the interaction records between user and the system (the type and number of learning units accessed, the type and amount of help asked, the correct and wrong answers to exercises, the marks obtained from exercises, etc.).

5. RUN OF THE SYSTEM

A web-based intelligent tutoring system $S = \langle UE, PE \rangle$ consists of user environment and pedagogical environment and a set of its global states $S = \{s_0, s_1, \dots\}$ is a sub-set of $U \times L \times C \times E$. A global state s composed of a user state u , a cognitive state l , a learning unit c , and an evaluation state e , that is, $s = (u, l, c, e)$. The goal user state $s_g = (u_g, l_g, c_g, e_g)$ represents the goal state that is expected by a user agent and a pedagogical agent; and $s_1 = (u_1, l_1, c_1, e_1)$ is an initial global state composed of an initial user state u_1 , an initial cognitive state l_1 , an initial learning unit c_1 and an initial evaluation state e_1 .

The run of an intelligent tutoring system can be viewed as a sequence of global states. A sequence (s_1, s_2, \dots) of S is a run of system $S = \langle UE, PE \rangle$ if and only if:

- the initial global state $s_1 = (u_1, l_1, \text{teach}_1(l_1, k\text{sel}_1(l_1, e_0, c\text{sel}_1(l_1))), e_1)$; where $l_1 = \text{trans}_1(u_1, \text{perc}_1(\text{vis}_1(u_1)))$; $e_1 = \text{eval}_1(l_1, i_1, \text{teach}_1(l_1, k\text{sel}_1(l_1, e_0, c\text{sel}_1(l_1))))$; and e_0 is the evaluation of the domain knowledge level of user agent;
- for every k , if $s_k = (u_k, l_k, c_k, e_k)$, $s_{k+1} = (u_{k+1}, l_{k+1}, c_{k+1}, e_{k+1})$, then $l_{k+1} = \text{trans}_k(u_k, \text{perc}_k(\text{vis}_k(u_k)))$; $u_{k+1} = \text{update}_k(l_k, i_k, \text{eval}_k(l_k, i_k, \text{teach}_k(l_k,$

$k_{sel_k}(l_k, e_{k-1}, c_{sel_k}(l_k)))); c_{k+1} = teach_k(l_k, k_{sel_k}(l_k, e_{k-1}, c_{sel_k}(l_k)));$ and $e_{k+1} = eval_k(l_k, i_k, teach_k(l_k, k_{sel_k}(l_k, e_{k-1}, c_{sel_k}(l_k))))).$

The run of system S can be described as follows: first, a pedagogical agent observes an initial user state u_1 in its cognitive process (using partition function vis , perception function $perc$) and change its state to l_1 ; then the pedagogical agent does the following tasks: (1) Select the set of knowledge concepts used to represent a teaching content according to its state via function c_{sel} ; (2) Select the knowledge concept to be taught via function k_{sel} ; (3) Select a learning unit to teach the user agent via function $teach$ and get c_1 ; and (4) Evaluates the performance of the user agent after the learning unit has been taught via function $eval$ and get e_1 . Thus we get the initial global state of the system s_1 .

If s_1 is not in accordance with the goal state s_g (e.g. user agent haven't finish the learning of all the knowledge concepts in a teaching content; the performance of user agent haven't reach the expectations of the user agent or the pedagogical agent, etc.), then repeat the above process to get the next global state s_2 until there is some s_j such that it is in accordance with the goal state s_g .

6. CONCLUSION AND FUTURE WORKS

A formal model of a web-based intelligent tutoring system is presented with the following properties:

- ✧ The system represents domain knowledge based on ontologies to improve the sharing and reusing of domain knowledge.
- ✧ The system constructs the user environment based on users' cognitive abilities, knowledge levels, learning styles, psychology characteristics, etc. in order to improve the self-adaptability and pedagogical effects of the system.
- ✧ The model distinguishes the information about a user and what a pedagogical agent knows about the user agent.
- ✧ The running process of the system is defined and discussed in detail to show that the model is practicable.

The system discussed in this paper has only one user agent and the extension of the model to include multiple users are part of our future works. Furthermore, the lacking of face-to-face communication in web-based intelligent tutoring systems reduces users' learning motivation. Hence, the utilizing of the communication, interaction and self-adaptability of agents to improve the interaction between users and between users and the system in order to strengthen the learning motivations of users and to improve teaching effects is also our future works.

NOTES

The work is partially supported by the Natural Science Fund (NSF) of Yunnan Province (No.2004F0017Q), the key project of NSF of Yunnan Province (No. 04F00062), and the Education Bureau Youth Fund of Yunnan Province (No. 03Y312D, No.04Y769D).

REFERENCES

- [1] J.M. Fritz, Provides Intelligence in Web-based Tutors. *North American Web Developers Conference*, October 1998.
- [2] G. Katerina, The design of a 'motivating' intelligent assessment system, In *Proceedings of Lecture Note in Computer Science* 1839, pages 208-218, 2002.
- [3] R. Reddy: Three open problems in AI. *JACM* 50(1): 83-86, 2003.
- [4] M. Specht, R. Oppermann, ATS-Adaptive Teaching System: a WWW-based ITS, In U. Timm (Eds.), *Proceedings of Workshop Adaptivität und Benutzer modellierung in Interaktiven Software systemen*: ABIS 1998.
- [5] P. Brusilovsky, Adaptive and Intelligent Technologies for Web-based Education. In: Rollinger, C., Peylo, C. (eds.) *Kunstliche Intelligenz, Special Issue on Intelligent Systems and Teleteaching*, 1999.
- [6] N.Manouselis & D.Sampson: Dynamic Knowledge Route Selection for Personalised Learning Environments using Multiple Criteria Analysis, In *Proceedings of the 20th IASTED International Multi-Conference on Applied Informatics*, 2002.
- [7] J. Lim, Theoretical bases on web-based instruction from a viewpoint of interactivity, In *Journal of Educational Technology*, 5(1):29-54,1999.
- [8] Georgouli, K.: Modelling a Versatile Mathematical Curriculum for Low- attainers. In *Proceedings of the 8th Panhellenic Conference in Informatics*, pages 463-472, 2001.
- [9] Vassileva, J.: Dynamic Courseware Generation. In *Journal of Computing and Information Technology*, 5: 87-102,1997.
- [10] Angelides, M., Garcia, I.: Towards an Intelligent Knowledge Based Tutoring System for Foreign Language Learning. In *Journal of Computing and Information Technology*, 1:15-28, 1993.
- [11] Aïmeur, E., Frasson, C., Dufort, H. "Co-operative Learning Strategies for Intelligent Tutoring Systems". In *Journal of Applied Artificial Intelligence*, 14 (5): 465-490, 2000.
- [12] M. Wooldridge and A. Lomuscio. A Computationally Grounded Logic of Visibility, Perception, and Knowledge, in *Logic Journal of the IGPL*, 9(2):273-288, 2001.