

Teacher Model to Personalize Support for the Implementer Teacher in Authoring an Educational Adaptive Hypermedia

Arnoldo Rodríguez

Universidad de Costa Rica, Turrialba, Costa Rica

Abstract

This paper analyses how a Teacher Model supports the emulation of teaching styles for enhancing decision-making skills. Our general goal is to contribute to making the authoring of adaptive Web-based applications easier. This research also studies how an adaptive hypermedia system can use the Teacher Model to facilitate personalized support.

We pay attention to the Teacher Model as the main source of information that a system uses for making decisions in adaptation and recommendation of examples that matches the teacher's preferences. In our research, the system uses information from the Teacher Model that reduces the complexity of adapting to different teaching styles in order to propose appropriate examples for a particular teacher.

A probabilistic method is used to infer the teaching style of a new teacher, whether the system does not have any information about he and he needs to start using the system. Instead of using a static Teacher Model, we use an evolving model that is applied to track the changes of the teacher's teaching styles.

1. Introduction

There are two approaches in intelligent help: learning support and problem-solving support [1]. The learning support helps the user to extend his knowledge about a subject and problem-solving support helps users find the solution to a problem [2].

In this paper we focus on the second approach and on how we can use a Teacher Model to provide more suitable problem-solving support in a Web-based system, called ARIALE (Authoring Resources for Implementing Adaptive Learning Environments), for teaching decision-making skills in Network Design.

A Teacher Model is the information about human teachers and their personal manners of teaching a subject. A Teacher Model keeps a structured representation of the teacher's knowledge about how to teach the expertise to solve problems in a knowledge domain. It is a model of how someone

skilled in a particular domain represents the knowledge for solving problems in that domain.

In this research, the teacher is an implementer teacher [9], and this concept involves the teaching styles and personality attributes of a human teacher who is an expert in teaching decision-making skills. By implementer teachers, we mean university professors who are using a Web-based authoring tool to create courseware for a subject or course prepared by a designer [14]. These teachers may be experts in the subject, but not necessarily skilled at using Web-tools.

A teacher is an implementer of a course who needs to be helped not only in the different pedagogical strategies involved in teaching decision-making, but also in the usage of an authoring tool for creating Educational Adaptive Hypermedia (EAH). He has to solve a problem: create materials for his courses. In addition, he has experience in teaching and using authoring tools, but his goal is not to become an expert about a specific authoring tool.

Designers create computer-based educational environments, but it is the teacher who adapts these environments for teaching courses according to their particular interests.

Kinshuk establishes that there is a difference between an implementer teacher (represented by a human teacher model) and a designer [9][10]. Usually, a teacher who wants to adapt a system to his needs (implementer teacher) cannot change any parameter of the system.

Research on Intelligent Tutoring Systems (ITS) attempts to create a perfect teacher rather than a teacher's tool [10]. ITS and EAH cannot replace all the functions of a human teacher, consequently we focus on enhancing tools that support the human teacher in the authoring process of an EAH.

The main objective of using these types of systems is designing strategies to support an interactive process which evaluates students' reactions and generates the required feedback for each reaction until the proposed pedagogical goal is achieved.

In addition, ARIALE pays attention to the Teacher Model and its Teacher Model is used to support the decision of the system in adaptation and recommendation of examples for the implementer teacher (problem-solving support) [13][19].

In the following sections we discuss other research related to our subject, the Teacher Model and the teaching style, the classification of a new teacher, the update of the Teacher Model, the use of the Teacher Model to recommend examples and our conclusions.

2. Literature review

Using a Teacher Model to support adaptive hypermedia applications is not a very widespread practice, but based on a teacher's model, the WEAR system [20] helps authors with teaching (strategy), and allows teachers to share exercises with one another.

In other cases, researchers have taken into account the pedagogical knowledge that teachers have about how to teach a specific skill [8][3].

An application of machine learning for user modelling can be seen in adaptive Web site agents and intelligent systems, which learn from visitors' access to a Web site. Research on intelligent profiling and interactive learning facilitates the acquisition of user information without unnecessary effort on the user's part.

In general, researchers are interested in modeling the expertise of one human teacher to insert his expert knowledge into an ITS [8][3]. An exception is ARTHUR, a system with a different approach that includes versions of different teachers for the same course and recommends to each student the course that fits his learning style [6]. However, ARTHUR courses are static and adapting contents to other teachers' visions is not possible.

Additionally, only a few researchers focus on topics indirectly related to Web-based intelligent help and supporting teachers.

In the past years, research on EAH and ITS focused on the Students Model and static Pedagogical Models but pioneer efforts related to the Teacher Model are arising [9][20].

3. General structure of the system

ARIALE is a prototype of a Web-based EAH assistant made up of an authoring tool and an Intelligent Help System (IHS).

The help system supports teachers while they are authoring learning activities. ARIALE has a structure of multi-linked concepts and multiple layers.

The architecture of ARIALE includes three main modules, Session Manager, Planner, and Helper, which run most of the processes on the server side.

The data that the system processes are also stored on the server side as part of the following components: Curriculum, Teacher Model, Pedagogical Model, Learning Sessions Base and Resources. More details of the structure of this system can be found in [15] [16].

4. Decisions about teaching and teachers

In ITS and EAH, a Pedagogical Model provides a model of the teaching process, which is adapted to the different needs of each student. Based on the student model, the pedagogical model can make pedagogical decisions about what aspects of the domain knowledge should be presented to the learner, when to review, when to present a new topic, and which topic to present.

The Pedagogical Model is a unique and static way that a particular teacher applies to teach a subject. However, implementer teachers can need to adapt the Pedagogical Model to their particular interests and styles to implement a course. Using an authoring tool, a teacher would be able to adapt contents according to his requirements, limitations, and pedagogical goals.

A teacher should be able to define his pedagogical goals for the activities that he proposes to the students. For example, specifying some criteria to be applied in a particular type of analysis that his students must develop. This view refers to the teacher as a person with particular characteristics and preferences, who wants to adjust or adapt an existing ITS or EAH according to his interests and expertise.

The implementation of a course allows each teacher to adjust it to his pedagogical goals by creating specific learning sessions. Then, a teacher can implement the course in a variety of ways by authoring different sessions.

A session is like a hypertext that can be represented by an all-connected graph [18]. The nodes are the activities, and a teacher can go from one to another by following a sequence, or he can jump from any node to another, according to his interest.

Our approach is also intended to allow the teachers to integrate theory and practice by applying knowledge to solve decision-making tasks in real-world problems using problem-based learning, simulations or role games. According to these strategies, a teacher requires the students to assume the roles of "players" engaged in a competition to optimize the design of a bank network. Two other strategies that complement competition are analysis by comparison and analysis of sensitivity of cases. The analysis of cases allows the student to select and decide between two options through a comparison of the two cases. This is the phase in which the student acquires competence. After this, the student explores an environment, modifies it, and arrives at the proficiency phase.

As part of a sequence of three tasks, comparison and modification of cases are the two initial tasks that converge into the third task: optimization [15]. During the competition (optimization), the student has to master the process to create solutions. By doing this, he acquires expertise.

This process is repeated with each activity, and each time the student acquires new schemas or patterns for making decisions, he accumulates more and more expertise. Throughout these tasks, students can gradually improve their expertise in making decisions about how to design a network.

As a student progresses, he moves on from the level of competence, passing to the proficiency level until he reaches the expertise level, according to Payr's phases [12].

4.1. Modelling the teacher

A Teacher Model is the information about each human teacher and his personal manner to teach a subject. In many cases, this type of information is included in the Pedagogical Model, but we are considering the Teacher Model as a specialization of the user model.

The user model is a set of information describing a particular teacher who is using a system. This model includes information about how to teach the expertise needed to solve problems in the knowledge domain. It is a model of how someone skilled in a particular domain stores such knowledge in his mind. For example, one teacher may prefer to use simple examples and another may prefer case studies that are very complex.

The Teacher Model profiles the characteristics of the teacher according to his identification, knowledge level, learning style, teaching style, preferences, and history of previous activities.

It is also possible to classify the teacher's characteristics according to: cognitive factors related to the knowledge; interests and skills of the teachers; as well as affective aspects that imply preferred media for receiving messages, willingness to share and communicate knowledge, or layout details such as the size of the letter to show running text on the screen. Table 1 shows some examples of each type of aspects.

A human teacher model includes personal characteristics of each teacher, information about how he teaches and information about how he acquires more knowledge and experience using an authoring tool.

In our system, for example, a human Teacher Model also keeps the teacher's name, his age, the course he teaches, the type of examples that he prefers to use, the learning sessions that he has authored and the help received.

As a consequence, modelling a teacher also implies to store information about the teacher's preferences, his activities and interests (frequent sessions, exercises), his level of expertise [20], his goals and plans, his experience and skills [5], his teaching style, his prior knowledge about the platform, his discipline subject to teach, his learning ability, and the capabilities of the computer hardware

and software he is using. For example, an implementer teacher can have some learning sessions in use and some additional ones to be finished; his skills to author tasks to compare examples are high but he does not have any ability in defining a competition between students to find the best solution for a problem. This information is different in comparison with another teacher who uses completely finished learning sessions only and who has very high skills performing only tasks related to the analysis of the sensitivity of an environment.

Table 1. Examples of cognitive and affective characteristics of a Teacher Model

Type of characteristic	
Cognitive	Affective
Years of experience teaching this course.	Preference of text, image or video for receiving help.
Class of the sessions used generally by this user (teaching style).	Preference for sharing knowledge.
Experience using computers.	Acceptance or not of animated agents.
Concepts most frequently required from help.	Layout of a page.
Language.	Preference of media to teach.
Background area of the teacher.	Preference for editing existing sessions.

The data for the Teacher Model of a particular teacher can be acquired using two methods: stereotype and overlay model.

A system acquires some data mentioned in the previous paragraphs when a teacher provides data directly or explicitly, for example when he fills in a form with details about his interest and preferences. Other type of data is acquired implicitly from the interaction between the teacher and the system. For example, using a dialogue box, a teacher provides explicit information about his preference using or rejecting a recommendation.

The registration process is the explicit way to supply the profile with data provided by the teacher while filling the specific forms out. Since some teachers might find this process uncomfortable or tedious, we assume that in some cases data will be missing. To overcome this problem, the available teacher's data can be compared with other teachers' data to reuse or copy their data [5]. This is the first method: stereotype.

If a teacher has never interacted with the Website, he does not have any statistics, and then, to solve this "start up" or cold-start problem, the system uses a collaborative filtering method [4] that takes into account the similarities among teachers with similar characteristics. The system believes that teachers with similar characteristics can share teaching styles and it infers that examples used by

the teachers in the selected group can also be used by the new teacher.

The other method is the overlay model, which represents the knowledge of an individual teacher as a record of the knowledge that a user has of the domain concepts. For example, for each concept of the domain there is a counter of the number of times that the teacher consulted this concept [5].

If a teacher accesses a help topic, this event is implicit information included as a part of his overlay model. As a consequence, the Teacher Model is also a repository of two types of the teacher data: the initial information (background, preferences, etc.) that forms the Teacher's Profile, and the history of his performance and pedagogical decisions (Overlay Model).

The information in the Teacher's Profile, does not change frequently, such as the teacher identification, username, background knowledge and some preferences. The Overlay Model changes continuously because it keeps data on the sessions, the times a specific fragment of text has been accessed, and the teacher's preferred class of examples. For example, during the initialization of the Teacher Model for a particular teacher, our system ARIALE compares the teacher's data such as the subject to be taught, the teacher's background and the expertise in teaching the subject with other teachers' data. As a result of that comparison, the teacher is classified in a group of teachers with similar expertise, background and subject to teach, and the system infers that examples used by the teachers in the selected group can also be used by the new teacher. When the teacher has chosen examples by himself, data related to those examples that the teacher prefers is stored in the Overlay Model to support future recommendations of examples. That data allow ARIALE to generate, test and recommend new examples that matches the teacher's preferences about examples.

The difference between stereotype and overlay model is that a stereotype compares a teacher's characteristics to other teacher's characteristics, and an overlay model stores data related to a particular teacher's knowledge.

The Teacher Model is the main source of information that the system uses for making decisions about supporting implementer teachers.

4.2. The Teacher Model in ARIALE

The Teacher Model used in ARIALE includes information of the teacher data, as the initial characteristics of the teacher (background, preferences, etc.) that forms the Teacher's Profile, and the history of his performance and pedagogical decisions, namely the Overlay Model, as Figure 1 depicts [14].

As seen previously, the Teacher Model stores two types of teacher data in the Teacher Profile: Demographic and Background knowledge. The data in both types is detailed as follows [15]:

Demographic

Age.
Gender.

Background

Teacher's self-evaluation of his skill or experience teaching the selected subject.

Number of times the teacher has taught this type of course.

Teacher's background discipline or area of expertise.

Teacher's skills using computers.

Teacher's experience using Web-based technology for teaching.

Teacher's experience using this Web site.

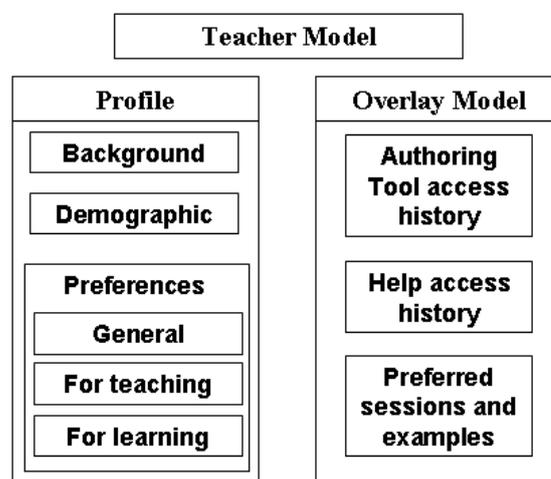


Figure 1. The components of the Teacher Model

The Teacher Model also organizes teachers' information related to the Overlay Model, as Teaching Styles, Learning Styles, Preferences and Performance Statistics. Table 2. shows information of other teacher's characteristics.

Table 2. Other teachers' characteristics

Group	Description
Performance using the system.	Statistics about the interaction of the teacher with the system.
Learning style.	Preferences for learning.
Teaching style.	Preferences for teaching.
Preferences.	Attitude toward sharing examples, communication, type of help, etc.

Our Teacher Model allows the system keep data about the teacher's experience, stores statistics about his performance in authoring learning activities and about his experience with help.

5. The evolving teaching style

As students have different learning styles [7], teachers work in different manners and they have particular tendencies in teaching, for example formal authoritarian, demonstrative, suggestive, facilitative, collaborative, delegative, problem solving, case-based, progressive, sequential and others.

While there is not a particular definition, a teaching style can be described by the teaching methods that a teacher uses, for example, his personal characteristics, the way in which a group of teachers deals with some type of subjects, or the general patterns of classroom behavior such as audiovisual or text based pedagogical strategies, examples, simulations and demonstrations used to teach.

A teacher can follow the Pedagogical Model but adapting it by using particular examples and goals. The examples and pedagogical goals that a teacher uses to teach a subject are his teaching style. Then, this data can be included in a Teacher Model that stores the teaching style in terms of the demonstrative, case-based, and audiovisual methods and the characteristics of the examples that a teacher uses.

The examples used by the teacher illustrate the way he thinks and the way he teaches. This information delineates his knowledge (expertise) about teaching the subject. If the teacher changes this preference, the system has to learn this change in order to adapt and provide appropriate support. A teacher's style changes when the most frequent class of example used by that teacher changes. The focus is on his current style, not on his global style, because the objective is to reflect the teacher's latest tendency. This teaching style is not static because it evolves as the teacher's examples changes. For instance, if a teacher has three sessions classified as "class 2", and then starts creating a session with examples from "class 4" and accumulates four sessions with this class, the style will automatically change from "class 2" to "class 4". After recognizing a change in the teacher's style, the system will recommend sessions that contain examples matching the new style.

Each time a teacher finishes a session, the system updates the information about the most frequently used class of examples in the corresponding category. For example, in a category "Beginner", most teachers prefer examples from the class of example 1; thus, the teaching style of this category is class 1. The system stores data related to these changes and doing this the systems learns and improves its ability to deliver future support. The data stored in the Teacher Model can be used as a basis to chose and recommend examples which characteristics are similar to the characteristics of the examples that a teacher uses.

The following algorithm shows the process that updates the teaching style.

```

Procedure Updating Teaching Style ()
  IF a session is finished THEN
    Classify the index example of the session
    Increase the teacher's counter for the
    selected class of example

  IF the current teacher's preferred class == the
  class of the last classified session THEN
    Do not update the teacher's preferred class
  ELSIF teacher's preferred class counter < the
  teacher's counter for the last selected class of
  example THEN
    teacher's preferred class = the last selected
    class of example

  Increase the counter of the same selected class of
  example in his category

  IF the current teacher's preferred class == the
  class of his category THEN
    Do not update the category class
  ELSIF the current category class counter < the
  counter of the same selected class of example in
  his category THEN
    current category class counter = the same
    class of example that the teacher had
    selected.
}

```

This method is important because it recognizes that a teaching style is not determined by an initial diagnosis, but rather that this style is dynamic and evolves over time. Every time this algorithm is executed, the system might acquire new knowledge about the teaching style of the teacher.

6. Classifying a new teacher

In this research, the Teacher Model includes information that comes from the registration process, the interaction between the teacher and the system, and the examples that teachers use for teaching decision-making on Network Design [14].

The creation of the Teacher Model of a particular teacher involves a process that starts in the registration phase, continues with the processing of the explicit information provided by the teacher to create a profile, and finishes by classifying the teacher as a Beginner, Intermediate or Advanced teacher.

Once a Teacher Model is built, data stored in it also allows the system to identify the teacher's interest and preferences, store and update new information or provide information that the system needs to make decisions.

ARIALÉ assumes that a new teacher can reuse the type of examples most often used by other teachers in his category. For example, teachers of similar number of times teaching a Network Design

course, who have the same background and experience teaching the subject, can have similar preferences to use a particular type of examples of network topologies. Of course, there is a failure risk because they can use different examples, but we took the risk of failing to start recommending examples, instead of making a simple random choice or a fix option for every beginner [14]. This assumption is more likely than a random method because we are taking into account similarities between teachers rather than assuming they are absolutely different.

Many methods can be used to model the teachers' characteristics, especially for the initialization of a Teacher Model, such as classification. Classification maps teacher's data into one of several predetermined classes.

To classify teacher's characteristics and to assign the new teacher to the category in which the teacher's data are most suitable, our system uses a naïve Bayesian classifier, a common technique in user modeling. Once the teacher belongs to a category, the most frequently used class of examples in this category is used to recommend sessions and examples to the new teacher.

The following example of a new teacher has to be classified in one of three categories (Beginner, Intermediate or Advanced). We associate each characteristic with a number for future identification in Table 3.:

1. Age = 45
2. Gender = Male (M)
3. Self-evaluation = Beginner (B)
4. Times teaching this course = 1
5. Teacher's background = Network design (ND)
6. Teacher's skills = Medium (M average)
7. Teacher's experience using Web-based technology = Never (N)
8. Teacher's experience using this Web site = Never (N)

Generally, people until 30 years have the highest use of Internet, after that the use of information technology becomes lower with the pass of the time; finally, the period over 55 years age is of very low Internet use, for example. We have defined these rules that organize data about age in three ranges:

If teacher's age ≤ 30 , then his age range is 1.

If teacher's age > 30 AND ≤ 55 , then his age range is 2.

If teacher's age > 55 , then his age range is 3.

This new teacher's characteristics are summarized as follows in Table 3. [14]:

Table 3. A new teacher's attributes and values

Teachers	Teachers' Characteristics								Class
	1	2	3	4	5	6	7	8	
A new teacher	2	M	B	1	N D	M	N	N	?

Using a training dataset, the new teacher is classified by applying the following equation (1), that uses m-estimate of probabilities [11]:

$$P(a_i / v_j) = \frac{nc + mp}{n + m} \quad (1)$$

where:

- a_i is the value of each attribute a_i tested for each class v_j . For example, an instance of a_i can be 2.
- n is the total number of instances in each class v_j . In this case of class Beginner, $n = 9$.
- n_c is the number of instances with attribute a_i and class v_j . This is the number of times that the value of the attribute a_i appears in each class v_j . For example, the value "M" appears one time in the Advanced category, in this case $n_c = 1$.
- p is a priori estimate for $P(ai/vj)$. We assume that the probabilities of all attributes are equiprobable and each attribute can have three possible values and $p = 1/3$.
- m is a constant used to avoid the possible consequences that could arise if $n_c = 0$ (in this case the calculation would be zero). We use $m = 8$ as a constant because we have eight attributes in the examples of our training dataset.

After the calculation of the probabilities of each attribute, the new teacher is classified in the category "Beginner" because the classifier predicts that this class has the highest probability for this teacher. Once the teacher is classified, the system has new information with which to support the teacher with recommend examples

7. Updating and learning

After the new teacher authors his first learning session, the information from the teacher's category is not taken into account anymore for recommendations. Over time, as each teacher acquires more experience using the Web site, teaching the subject and authoring his own learning sessions and examples, the system switches from the stereotypes and collaborative filtering to take into account each teacher's own examples. The system collects data related to his performance while monitoring the teacher-system interaction. These data refer only to pertinent variables that the system uses to make decisions in adapting recommendations. Variables that the system monitors include the contents (examples) that the

teacher uses in each session [14]. Using an overlay model and content-based filtering, the system gets a more accurate personalization of each teacher automatically, by monitoring the examples that each teacher uses.

After recognizing a change in the teacher's style, the system will recommend sessions that contain examples matching the new style. A teaching style is not determined by an initial diagnosis, but rather this style is dynamic and evolves over time. The changes in the teacher's teaching style are updated in the corresponding Teacher Model and, doing this, the system also learns a particular teaching style [14].

The following algorithm describes the process used to update the Teacher Model in the server side when a Web page is unloaded:

```

Procedure Update Teacher Model ()
  Send information about interaction to the server side
  Update teacher's overlay model (history statistics)
  Generate a new plan to follow
  IF a session is finished THEN
    Update the teacher's teaching style
    Update the teacher's category
  }

```

8. Support based on a Teacher Model

In our research, the system uses information from the Teacher Model to make decisions about updating the teacher's style. A function of our Teacher Model is to allow the system to adapt to different teaching styles in order to recommend appropriate problem-solving support (selecting the class of examples to be recommended). For example, information in the Teacher Model allows ARIALE to recommend some network topologies from a case base [14].

If the teacher does not like an example recommended from a case base, the system can generate a new example according to the teacher's teaching style. The generation, testing and recommendation of examples, that is a type of automatic creation of courseware material, depends on the information contained in the Teacher Model. If the generator creates an example that is out of the scope that the teacher prefers, the example is aborted and a new one is generated [13]. Figure 2 shows the interface of the recommendation of an example (a network topology).

In addition, to help teachers to find and access specific and personalized information, we designed a process based on the knowledge existing in the Teacher Model. The process to find and personalize specific information starts when the teacher gives explicit information to the system during the registration.

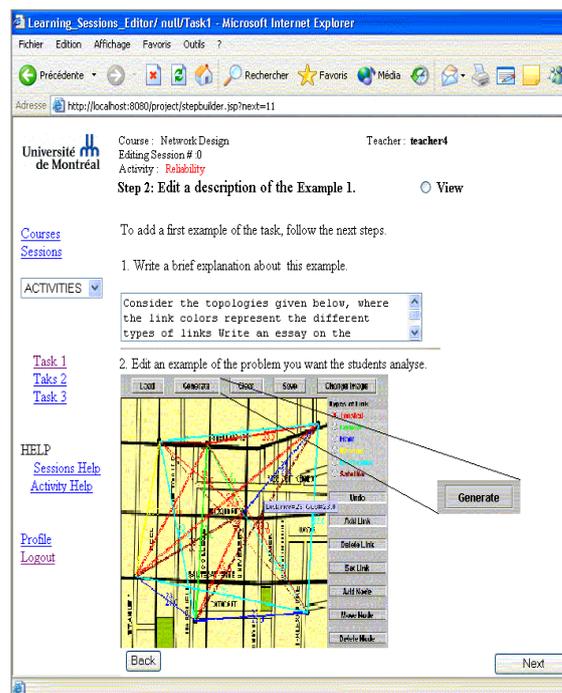


Figure 2. The recommendation of examples depends on the Teacher Model

After the creation of the Teacher Model, every time the teacher authors a learning session, a cycle of interaction and changes is repeated to provide adaptive help. In this cycle, the system takes information from the Teacher Model to define the teacher's skills, plans and preferences to create Web pages with resources adapted to the teacher's characteristics. For example, setting the teacher's skills means that the system stores information about the current teacher's skills in the working-memory of the system, to support its decisions. Defining a plan is coding the short-term activity, tasks and steps that the teacher has to perform in the format for representing a plan [18]. The teacher's preferences, such as the teacher's language, are also included in the working-memory of the system to support its decisions about contents and specific resources to be retrieved.

In ARIALE, a teacher authors a learning session by changing values in a given step. These changes are the teacher's feedback that the system receives. Once the teacher has finished the edition of the step, a new Web page is loaded, the Teacher Model is updated with new information acquired during the authoring process and the interaction in a new Web page starts again, taking into account the last changes in the Teacher Model. In this interactive process, the teacher interacts with the authoring tool and with the help system and both parts generate feedback to support the decision of each other [17].

9. Conclusions

This research tries to improve the knowledge about acquisition, updating and management of user models for adaptive and more personalized Web-based systems. Based on a Teacher Model, we are using a recommender that reduces the complexity of finding an appropriate example for a particular teacher.

Regarding the teacher's aspects, the concepts of human teacher model and implementer teacher are basic to model teachers' data in ARIALE.

In the section devoted to the Teacher Model, we modeled the preferences of a human teacher and his teaching style (type of examples that he uses most frequently). The model also includes information about teacher's background, preferences, and progress in interacting with the system. This model supports the decisions that ARIALE makes to help teachers. The recommendation of cases and other types of help depends on the information contained in the Teacher Model.

A probabilistic method is used to infer the teaching style of a new teacher, whether the system does not have any information about his and he needs to start using the system. Instead of using a static Teacher Model, ARIALE includes an evolving model that is applied to track the changes of the teacher's teaching styles.

This view of the Teacher Model is helpful, because classical approaches in EAH suggests that each teacher must follow a static Pedagogical Model; instead, our Teacher Model is a knowledge base that allows our system to flexibilize and variate the Pedagogical Model.

10. Acknowledgements

This research was supported by the University of Costa Rica, the Consejo Nacional para Investigaciones Científicas y Tecnológicas, and the Ministerio de Ciencia y Tecnología de Costa Rica. This work is also supported partially by the National Sciences and Engineering Research Council of Canada (NSERC), the Fonds Québécois de la Recherche sur la Nature et les Technologies (FCAR), and Valorisation Recherche Québec (VRQ) of the governments of Canada and Quebec.

11. References

[1] J. Aberg, *Live Help Systems: An Approach to Intelligent Help for Web Information Systems*, Ph.D. Thesis, Department of Computer and Information Science, Linköpings universitet, Sweden, 2002.

[2] P. Brusilovsky, and E. Schwarz, "User as Student: Towards an Adaptive Interface for Advanced Web-Based Applications", A. Jameson, C. Paris and C. Tasso (eds.),

Proceedings of the Sixth International Conference on User Modeling, Springer-Verlag, Italy, 1997, pp.177-188.

[3] B. du Boulay, and R. Luckin, "Modelling Human Teaching Tactics And Strategies For Tutoring Systems", *International Journal of Artificial Intelligence in Education*. 12, 2000, pp. 235-256.

[4] M. Eirinaki, and M. Vazirgianis, "Web Mining for Web Personalization", *ACM Transactions on Internet Technology (TOIT)*, Vol. 3, No. 1, 2003, pp. 1 – 27.

[5] M. Encarnação, "Multi-level User Support through Adaptive Hypermedia: A Highly Application-Independent Help Component", *Proceedings of the International Conference on Intelligent User Interfaces IUI'97*, USA, 1997, pp.187-194.

[6] J. Gilbert, and C. Han, "Adapting instruction in search of a significant difference". *Journal of Network and Computer Applications*, Volume 22, Issue 3, Austria, 1999, pp. 149-160.

[7] S. Graf, Kinshuk, and C. Ives, "A flexible mechanism for providing adaptivity based on learning styles in learning Management Systems", *10th IEEE International Conference on Advanced Learning Technologies*, ICALT, 2010 pp.30-34.

[8] N. Heffernan, *Intelligent Tutoring Systems have Forgotten the Tutor: Adding a Cognitive Model of Human Tutors*, Dissertation, School Of Computer Science, Carnegie Mellon University, USA, 2001.

[9] Kinshuk, "State-of-the-art in Adaptive Learning: Keynote Address Pedagogies & Communication", *Proceedings of the International Conference on Open & Online Learning (ICOOL 2003)*, University of Mauritius, 2003: <http://icool.uom.ac.mu/2003/papers/file/keynote/kinshuk.zi> p. (Access date: 23 March, 2004)

[10] Kinshuk, H. Hong, and A. Patel, "Extended ITS Framework with Human Teacher Model", C. H. Lee, S. Lajoie, R. Mizoguchi, Y D. Yoo & B. du Boulay (Eds.) *Enhancement of Quality Learning Through Information & Communication Technology (ICT)*, *Proceedings of ICCE/SchoolNet 2001 Conference*, Korea, 2001, pp. 1259-1262.

[11] E. Meisner, *Naive Bayes Classifier example*, 2003: <http://www.cs.rpi.edu/courses/fall03/ai/misc/naive-example.pdf>. (Access date: 6 October, 2003)

[12] Payr, S., "The Virtual University's faculty: An overview of educational agents", *Applied artificial intelligence*, Vol. 17, Germany, 2003, pp. 1-19.

[13] A. Rodríguez, "Generation, Testing and Recommendation of Teaching Materials Using Classification Learning", *International Journal of Advanced Media and Communication (IJAMC)*, Vol. 2, Número. 3, Switzerland, 2008, pp. 308 – 324.

[14] A. Rodríguez, "Teacher Model supporting an educational adaptive hypermedia for an implementer

teacher”, *Proceedings of the Canada International Conference on Education (CICE-2011)*, Canada, 2011, pp. 531-536.

[15] A. Rodríguez, “Teacher Model for an educational adaptive hypermedia that supports the teaching of decision making”, *Proceedings of the International Technology, Education and Development Conference (INTED-2011)*, Spain, 2011, pp. 91-100.

[16] A. Rodríguez, E. Aïmeur, and F. Vázquez-Abad, “Training Teachers In: Teaching Decision-Making Skills”, *Proceedings of the II International Conference on Multimedia and Information & Communication Technologies in Education (MICTE 2003)*, Spain, 2003, pp.1964-1968.

[17] A. Rodríguez, E. Aïmeur, F. Vázquez-Abad, “E-Learning for complex Decision-Making with the support of a Web-based Adaptive ITS”, *Proceedings of the International Conference on Knowledge Engineering and Decision Support (ICKEDS'2004)*, Portugal, 2004, pp. 47-54.

[18] A. Rodríguez, E. Aïmeur, F. Vázquez-Abad, “Adaptive Help Techniques to Reduce the Teachers’ Cognitive Overload”, *Proceedings of the International Conference on Computers in Education (ICCE2004)*, Australia, 2004, pp. 1741-1750.

[19] F. Vázquez-Abad, A. Rodríguez, and E. Aïmeur, “Training the Teacher: A New Approach for Authoring ITSs for Teaching Decision-Making”, *Proceedings of the International Conference on Open & Online Learning (ICOOL 2003)*, University of Mauritius, 2003: <http://icool.uom.ac.mu/2003/papers/file/Rodriguez.pdf>. (Access date: 26 May, 2004)

[20] M. Virvou, and M. Moundridou, “Adding an instructor modelling component to the architecture of ITS authoring tools”, *International Journal of Artificial Intelligence in Education*, 12, 2002, pp. 185–211.