

From Classic User Modeling to Scrutable User Modeling

Giorgio Gianforme¹, Sergio Miranda², Francesco Orciuoli² and Stefano Paolozzi¹

¹ Dipartimento di Informatica e Automazione
University of ROMA TRE, Rome, Italy

² Dipartimento di Ingegneria dell'Informazione e Matematica Applicata
University of Salerno, Fisciano, Italy

Abstract. User Modeling still represents a key component for a large number of personalization systems. Maintaining a model for each user, a system can successfully personalize its contents and use available resources accordingly. Ontologies, as a shared conceptualization of a particular domain, can be suitably exploited also in this area. In this paper we explain some concepts about user modeling, particularly focusing on scrutability and its importance in ontology-based user modeling systems.

1 Introduction

In a world where information is increasing constantly, the challenge is not only to make information available to people at any time, at any place, and in any form, but specifically to say the “right” thing at the “right” time in the “right” way. User modeling researchers try to address and solve these issues.

User modeling is one of several research areas that intuitively seems to be winning propositions based on their obvious need and potential “return of investment”. Indeed as systems have become increasingly more complex in terms of large amounts of information to be conveyed and a wide range of task structures, it is important that a system should be able to provide effective guidance and assistance to enable the user to make full use of the available functionality. Moreover, given the variety of users of such systems, the fact that different users will not all have the same problems or needs, and that a user’s level of expertise is likely to change over time, it is desirable that interfaces to complex systems should be able to adapt to and support the requirements of individual users. Thus an information-providing system could tailor the form and content of information provided to users according to its assessment of what the users need or wish to know, while a dialogue module could construct the human-computer interaction according to the users’ preferences.

In this paper we analyze the characteristics of actual user modeling systems and their importance in human-computer interactions. Moreover, we explain the details of a fairly new concept in user modeling called scrutability: the ability of analyzing and eventually modifying a user model by its user. Finally we discuss about the role of ontologies in user modeling also in according to the scrutability concept.

The rest of the paper is structured as follows. Section 2 explains the main concepts of modern user modeling, describing the characteristics of a user model. Section 3 describes the importance of user modeling in the human-computer interaction field. In Section 4 the concept of scrutability and its importance in user modeling are described. Section 5 are devoted to the description of the ontology role in scrutable user modeling. Finally Section 6 concludes.

2 User Modeling Analysis

The term *user modeling* has been applied to the process of gathering information about the users of a computer system and using this information to provide services or information adapted to the specific requirements of individual users (or groups of users).

A “perfect” user model would include all features of the user behavior and knowledge that affect their learning and performance [20]. The information, contained in the user model determines the adaptability of the system. Constructing this kind of model is quite complex task, even for people not only for machines, thus simplified, partial models are used in practice.

The term user model has been used in several different ways in Artificial Intelligence (AI) and Human-Computer Interaction (HCI) and it is important to clarify these differences at the outset. According to software design terminology, the term ‘user model’ usually refers to the designer’s model of the users of the system, either in fairly gross terms, such as their expected level of proficiency, or more specifically in terms of the models users may have of the system and of the procedures involved when interacting with the system.

These models can be identified as the users’ system model and the users’ task model [10]. Generally speaking, designers models of users are implicit and thus are not represented explicitly within a computer system. This type of modeling is usually done once only, at the design stage of a system, so that any unanticipated user characteristics or any changes in user characteristics over time cannot be accommodated without a complete re-design of the system. A more refined system is one which is *adaptable*, that is system where a user can make choices among various options that affect the system’s behavior and can save these choices for future reference in a user profile. However, the most complex (or even sophisticated) kind of system, is an *adaptive* system which automatically acquires knowledge about its users, updates this knowledge over time, and uses the knowledge to adapt itself to the different users requirements.

The two main features of a user model in an adaptive system can be summarized as follows. First of all, the user model can be viewed as a knowledge source of information about users, which contains explicit assumptions on those aspects of a user that might be relevant to the interface of the system. Secondly, a user model could also be used to simulate the user activity, for example, to predict the effect of an action of the user. User models have been employed in this way in intelligent tutoring systems and also to provide anticipation feedback in natural language dialogue systems [5].

2.1 User Model Characteristics

Three main characteristics can be considered for the definition of a user model. Namely they are:

- what information about the user is included in the model and how it can be obtained
- how we can model the user representation in the system
- how we can distinguish between a model for individual user and a model for classes of users

We described these aspects in the following.

Information About the User. Information about the user, which is usually saved in the user model, can be divided in static (i.e they remain constant through the learning process) and dynamic (i.e. they can change during the learning process).

The static part of the user model includes user's personal characteristics (e.g. age, gender, level of instruction, etc.), user's capabilities (e.g. background knowledge, cognitive and non-cognitive abilities, etc.) and user's preferences. These characteristics are usually analyzed at the beginning of the learning process, using interviews (e.g. online questionnaires) and different tests [6].

The dynamic part of the user model includes user's knowledge, concepts and skills, learning style, motivation, viewpoints, current goals, plans and believes, learning activities that have been carried out, etc. The information for devising this part of the user model are gathered directly from users (e.g. specification of current goal), through tests and practice (e.g. test results, user history of responses and problem solving behavior), or user's actions (browsing behavior, visited concepts, time spent on page, total session time, selection of links, searching for further info, queries to the help system). This information is constantly collected during the learning process and is also used for updating the user model.

It is also possible to distinguish between explicit information acquisition (EIA) and implicit information acquisition (IIA) in user modeling. EIA usually refers to the addition of new information to the user model on the basis of information gathered from an external source. This external source can be represented, for example, by a user interacting with the system or a user such as a team manager who is reporting the characteristics of an employee or groups of employees to the system. IIA refers to inferring user model information on the basis of the knowledge available within the user model and rules of inference that are applied to this knowledge. This kind of information is usually acquired incrementally. Examples of IIA and EIA are given by user-driven acquisition (in which the system passively acquires information from user actions) and system-driven acquisition (in which the system identifies what information is needed and seeks to acquire it), respectively.

Representation of the User's Information. Different methods can be used for constructing the user model. They may include:

- Bayesian methods,

- Machine learning methods (rule learning, learning of probabilities, instance-based learning),
- Logic-based methods (first order predicate calculus), overlay methods,
- Stereo-type methods [17] [7],
- Specifically developed computational procedures (user’s expertise is calculated from their navigational actions or time spent on documents),
- Specifically developed qualitative rules and procedures (special rules regarding user’s properties or behavior),
- Other general methods (plan recognition)

In real application a combination of two or more methods is frequently applied, especially when different methods are used for initializing and maintaining the user model. This grants more accurate modeling and allows better exploitation of gathered information.

Constructing and Updating the User Model. The methodology for exploiting the user model is, in some way, similar to the user modeling construction methodology. Moreover, we can use well-known methods from artificial intelligence and machine learning (e.g. Bayesian networks, rule learning, instance-based learning, learning of probabilities, logic-based and heuristic), as well as other general techniques and principles (e.g. plan recognition), specifically developed computational procedures or specifically developed qualitative rules (e.g. rules for selecting and evaluating examples, rules for choosing adaptation type and rules for choosing questions).

Classes of Users vs Individuals. Grouping together classes of users, according to shared characteristics, has been a widely applied technique in user modeling and it is related to the topic of stereotypes [17] [15]. Stereotypes provide a way of reducing to some extent the problem of the unique modeling of each individual user as we can use stereotypes to fill in the background information on a person when we know only few details about them [16].

We can also refine the distinction between individual users and group of users in terms of the number of agents to be modeled and the number of models. Generally, we assume that the user model refers to a model of the person who really uses the system. However, a system may also be required to model other persons at the same time [19]. For example a manager may want to ask a system about characteristics of employees.

3 User Modeling in Human Computer Interaction

We have already mentioned that the user modeling problem involved two main research areas, i.e. artificial intelligence and human-computer interaction (HCI). For the purpose of this paper we are more interested in HCI.

A key objective of HCI researches is to make systems more usable, more useful, and to provide users with experiences which are strictly related to their specific background knowledge and goals. The challenge is not only to make information available to people at any time, at any place, and in any form, but specifically to say the “right” thing at the “right” time in the “right” way [18].

Some HCI researchers have been interested in user modeling because there is the potential that user modeling techniques will improve the collaborative nature of human-computer systems [2]. In HCI the different kinds of information represented in a user model are usually concerned with the user's cognitive style and personality factors.

The main objective in HCI is to provide a "dialogue" adapted to the requirements of the user and to the particular task which the user is currently engaged in. Thus, in addition to a user model, a task model is required which describes both the physical and the conceptual aspects of a domain [1]. While the task model is relatively stable, the user model, which includes the user's mental model of the domain and its task structure, is likely to change dynamically as the user interacts with the system. A requirement on user models in HCI is that they should be psychologically relevant, and this aspect is generally validated empirically in experimental studies. This information about users may be used in a predictive way to assist system designers to design a system with user requirements in mind; however, recent work in HCI has been concerned with the automatic adaptation of computer systems to their users and has thus converged with one of the main concerns of AI.

Recent studies relate user modeling with the new research area of "multimodality". By definition, "multimodal" should refer to the use of more than one modality, regardless of the nature of the modalities. However, many researchers use the term "multimodal" referring specifically to modalities that are commonly used in communication between people, such as speech, gestures, handwriting, and gaze. Multimodality seamlessly combines graphics, text and audio output with speech, text, and touch input to deliver a dramatically enhanced end user experience. When compared to a single-mode interface in which the user can only use either voice/audio or visual modes, multimodal applications give them multiple options for inputting and receiving information. The term "modality" is used to describe the different ways of operation within a computer system, in which the same user input can produce different results depending of the state of the computer. It also defines the mode of communication according to human senses or type of computer input devices. In terms of human senses the categories are sight, touch, hearing, smell, and taste. In terms of computer input devices we have modalities that are equivalent to human senses: cameras (sight), haptic sensors (touch), microphones (hearing), olfactory (smell), and even taste. In addition, however, there are input devices that do not map directly to human senses: keyboard, mouse, writing tablet, motion input (e.g. the device itself is moved for interaction), and many others.

In the field of HCI with reference to user modeling it is important to distinguish between the concepts of *adaptive* and *adaptable*. An interesting comparison between these concepts are sketched in [2] and reported in Figure 1 (see also [13] for more details).

4 Ontology and User Modeling: Scrutable Models

In the last years the concept of "scrutability" arises, connecting ontologies and user modeling more deeply. A user modeling system is called *scrutable* if users can not only examine the data in their user models, but also the processes that use that data for personalization.

	Adaptable	Adaptive
Definition	user changes (with substantial system support) the functionality of the system	dynamic adaptation by the system to current user and current task
Knowledge	knowledge is extended	contained in the system; presented in different ways
Mechanisms Required	layered architecture, domain models and domain-orientation, design rationale	models of user, dialogues and tasks, knowledge base of goals and plans, incremental update of models, powerful matching capabilities
Application Domains	information retrieval, end-user modifiability, filtering, design in use	information retrieval, active help systems, differential descriptions, user interface customization
Strengths	users is in control, users know his task best, success model exists, system knowledge will fit better	little or no user effort, no special knowledge of the user is required
Weakness	systems become incompatible, increasing complexity (users need to learn the adaptation component), users must do substantial work	loss of control, users have difficulty developing a coherent model of the system, few (if any) success models exist

Fig. 1. Comparison between Adaptable and Adaptive systems.

Scrutability means that the user should be able to explore the user model and modeling processes. It is fairly clear that the most important reason for scrutable user models is the possibility for users to know what information a system maintains about them.

An important characteristic of this kind of user modeling is that a scrutable user model enables the user to check and correct the model itself. This is a critical argument for scrutability, and it relates to the previous one stating the users right to know what the system stores about them. Often, it is simple to gather user modeling information from the analysis of that part of the users behavior which is observable by the machine.

To understand the importance of this models feature, let us consider the following scenario: a user allows another person to use his/her system account: the inferences are no longer about the individual we intended to model. The models accuracy can also be affected by people giving the user advice: someone might direct the user to type a series of complex commands. This could result in a user model with an inaccurate picture of the users behavior. If users can easily check the user model, they can correct it.

Making a model scrutable ensures that the user can see and control the bases of the systems inferences about them. Anyway, there is a potential problem in allowing users to see and alter the user model. Indeed, they might just decide to lie to the system and create a model of themselves that they would like to be accurate: perhaps they would like the system to show them as experts. Another typical situation is the presence of a "curious" user who might simply like to see what happens if he/she modifies his/her user model. It is clear that there is a concrete possibility that users may corrupt their user model, therefore it must be taken into account in designing the representation and support tools of a system.

However this possibility of user models corruption, it is not a sufficient reason to prevent users from controlling the systems model of them.

There are, obviously, differences between machines and people. These differences introduce the possibility for user-adapted systems to be especially effective in some interactions. In particular, the processes controlling an user-adapted computer system are normally deterministic. On the contrary, in human-to-human interaction, the processes behind actions may not be accessible. Although the machine cannot really know the users beliefs, it should be possible for the user to know the machines beliefs, especially the beliefs contained in a explicit model of the user. In collaboration (or co-operative) systems, the scrutability of the user model can play an important role in helping the user understanding the systems goals and viewing the interaction. Moreover, we can reduce user misunderstandings, due to expectations and goals held by the system and not appreciated by the user.

Scrutability can be the basis for adaptable systems. Indeed, we can expect to enhance the quality of the collaboration if both the user and system can be aware of the beliefs each holds about the other: an accessible user model can help the user be aware of the machines beliefs.

In the case of teaching systems, there are additional reasons for scrutability. Indeed, we can observe that it is possible to change the user model (usually called a student model in this context) to become a valuable basis for devising deeper learning [9]. In this context, the important role of a scrutable user model is to represent what the learner knows and does not know at any stage. If the model is externalized by the system, presented in a way that enables the student to reflect on and evaluate their own knowledge, the student is more motivated in developing a deeper understanding. Then the user may be motivated to learn more. The discussion to this point has emphasized the possibility of a teaching system making an externalized user model part of its teaching. There is also a learner-initiated view.

It can be observed that in a not-scrutable system, the users exploration and curiosity always corrupt the user model. This means that scrutability might enable the user to explore the system without compromising the accuracy of the user model.

Finally, scrutability becomes especially important when user models are very large, i.e. consisting of hundreds or even thousands of components (such as large ontologies describing user characteristics and behavior). Inference can be something that becomes very complex due to the number of components involved, creating considerable challenges to maintain the scrutability of the user model. There is the need to explore ways to structure and visualize user models to make the user modeling process in these situations more explicit and understandable.

5 The Role of Ontologies in User Modeling

Ontologies, defined by Gruber as “a shared conceptualization of a particular domain” are a fundamental element of the new generation Web, the Semantic Web [3]. Although ontologies have their foundations in the field of artificial intelligence, they have also attracted the attention of other fields such as knowledge management, information retrieval, and e-Learning.

Ontologies have also been recently applied to the field of user modeling (as proved by several researches [14] [4]) In these systems, ontologies provide an important link between the domain content, user models, and adaptation.

There is also a role for ontologies in providing structure to a user model. In fact, when the user model does not have a suitable structure to organize the components in the model, an ontology should be extremely useful. This is true essentially for two reasons: first of all, ontology can provide such a graph structure and, second, that structure should make sense to the user, in terms of the meanings of the concepts modeled [8].

Ontologies can also be used for reasoning. Indeed, moving from the knowledge of a subset of facts within the domain, new facts can be produced by inference. We can use inference on user models with an ontological structure to find out new facts about the user.

As stated in the previous sections, user models typically consist of different components, each representing a single piece of information or belief about the user. Although the user model may be very large with hundreds or even thousands of components, it is very important that the system maintains scrutability, particularly in reasoning about the data, whether it is performed by the user modeling system or other users who might have access to data from the user model.

Therefore, as previously stated in Section 4 it would be greatly useful for users to be able to explore possible inferences that can be made about them by an agent from their user model or better a partial version of this model. This is important because even though the partial user model may only contain a subset of data, many inferences can be made about the user on the basis of an underlying ontology.

This is true especially in case a user wants to keep private some elements of his model (e.g. personal details and contact information). A system can still provide an adequate picture of the user by making inference, even with partial information about the user himself. A similar situation is sketched in Figure 2. In this case, the user has chosen only a portion of his model to be shared (top right of the figure). On the other side the system may allow the user to visualize possible inference that can be made about him from this information (bottom right of the figure).

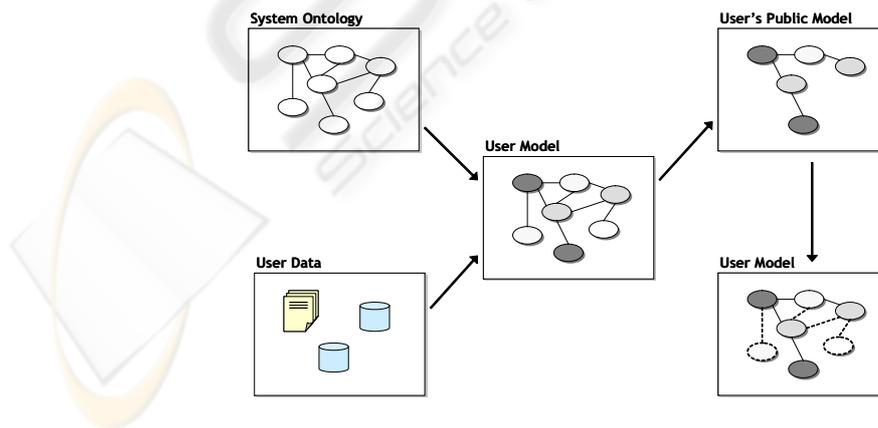


Fig. 2. An illustration of a user model.

Ontological inference is particularly important in environments where evidence feeds into the user model at very different *granularities*. A typical example can be found in the e-Learning systems, where course materials typically teach fine-grained concepts that contribute to higher level learning goals. It is important to be able to model both coarse and fine grain concepts in the learner model, so that learners can see their overall learning progress and goals through the coarse grained concepts, and also determine what elements of work contribute to these high level goals through the fine grained concepts [12].

In this way, we can exploit ontologies for both scrutability of the user model and also reasoning about users. Using ontologies for scrutable modeling raises several issues to be considered (as recently observed [11]):

- *Construction*. Creating ontologies is not a trivial activity. In fact, we require a source of knowledge of the domain we are trying to model (either through domain experts or documentation). By this process we can identify the concepts and relationships that make up the ontology, and a way to represent the ontology.
- *Enhancement*. When we create an ontology, it may not contain all the concepts and relationships necessary for the the particular objective. It is imperative that ontologies were easily and quickly updatable; this, however, introduces the challenge of being able to update (enhance) the ontology without compromising its integrity.
- *Interface*. An effective interface for viewing and exploring the ontology is obviously required. Moreover we need convenient tools for aforementioned construction and enhancement processes.

6 Conclusions

As a demonstration of the growing interest in ontology-based user modeling, a lot of researches and tools have been developed in the last years. Therefore, User Modeling still represents a key component for a large number of personalization systems. In this paper, after a review of the main concepts of user modeling, we have discussed about the importance of ontologies in building user models, reasoning about them and finally supporting *scrutable* user modeling system.

References

1. D. Benyon, J. Kay, and R. C. Thomas. Building user models of editor usage. In E. Andre, R. Cohen, W. Graf, B. Kass, C. Paris, and W. Wahlster, editors, *Third International Workshop on User Modeling - UM92 (Proceedings)*, pages 113–132. IBFI (International Conference and Research Center for Computer Science), 1992.
2. Gerhard Fischer. User modeling in human–computer interaction. *User Modeling and User-Adapted Interaction*, 11(1-2):65–86, 2001.
3. Thomas R. Gruber. Toward Principles for the Design of Ontologies used for Knowledge Sharing. *Formal Ontology in Conceptual Analysis and Knowledge Representation*, 1993
4. Dominik Heckmann, Eric Schwarzkopf, Junichiro Mori, Dietmar Dengler and Alexander Kröner. The User Model and Context Ontology GUMO Revisited for Future Web 2.0 Extensions. In *Proceedings of the International Workshop on Contexts and Ontologies: Representation and Reasoning (C&O:RR)*, Denmark, 2007.

5. Anthony Jameson. Impression monitoring in evaluation-oriented dialog. the role of the listener's assumed expectations and values in the generation of informative statements. In *Proceedings of IJCAI'83*, Karlsruhe, 1983. International Joint Conferences on Artificial Intelligence.
6. Anthony Jameson. User-Adaptive Systems: An Integrative Overview. Tutorial presented at UM99, International Conference on User Modeling, Banff, Canada, 1999.
7. Aleksandar Kavcic. The role of user models in adaptive hypermedia systems. In *Electrotechnical Conference, 2000. MELECON 2000. 10th Mediterranean*, vol. 1. 2000.
8. Judy Kay and Andrew Lum. Exploiting Readily Available Web Data for Scrutable Student Models. In *Proc. Proceedings of the 12th International Conference on Artificial Intelligence in Education, AIED 2005*, pages 763–765, Taiwan, 2006.
9. Judy Kay and Lichao Li. Scrutable Learner Modelling and Learner Reflection in Student Self-assessment. In *Intelligent Tutoring Systems. ITS 2006*, pages 338–345, Amsterdam, The Netherlands, 2005.
10. Muneo Kitajima and Peter G. Polson. A computational model of skilled use of a graphical user interface. In *CHI '92: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 241–249, New York, NY, USA, 1992. ACM.
11. Andrew W K Lum. *Light-weight Ontologies for Scrutable User Modelling*. PhD thesis, 2007.
12. G. McCalla and J. Greer. Granularity-Based Reasoning and Belief Revision in Student Models. In *Student Modelling: The Key to Individualized Knowledge-Based Instruction.*, Berlin Heidelberg Springer-Verlag. 25, pages 39–62, 1994.
13. Andreas Nrnberger and Tanja Falkowski. Adaptive user support in information retrieval systems. In *Information-Mining und Wissensmanagement in Wissenschaft und Wirtschaft*, pages 111–125, 2004.
14. Liana Razmerita and Miltiadis D. Lytras. Ontology-Based User Modelling Personalization: Analyzing the Requirements of a Semantic Learning Portal. In *Emerging Technologies and Information Systems for the Knowledge Society, First World Summit on the Knowledge Society, WSKS 2008*, vol.1, pages 354–363, Athens, Greece, 2008.
15. Elaine Rich. Stereotypes and user modeling. In A. Kobsa and W. Wahlster, editors, *User Models in Dialog Systems*, pages 35–51. Springer, Berlin, Heidelberg, 1989.
16. Elaine Rich. Users are individuals: individualizing user models. *International Journal of Man-Machine Studies*, 18:199–214, 1983.
17. Elaine Rich. User modeling via stereotypes. pages 329–342, 1998.
18. Cristian Rusu, Virginia Rusu and Silvana Roncagliolo. Usability Practice: The Appealing Way to HCI. In *Proceedings of First International Conference on Advances in Computer-Human Interaction, ACHI 2008*, pages 265–270, Sainte Luce, Martinique, France.
19. K. Sparck Jones. Realism about user modeling. In A. Kobsa and W. Wahlster, editors, *User Models in Dialog Systems*, pages 341–363. Springer, Berlin, Heidelberg, 1989.
20. Etienne Wenger. *Artificial intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.