
Downloaded from: http://eprints.bbk.ac.uk/311/

Usage Guidelines
Please refer to usage guidelines at http://eprints.bbk.ac.uk/policies.html or alternatively contact lib-eprints@bbk.ac.uk.

This is an author-produced version of a paper published in *Expert Systems with Applications* (ISSN 0957-4174). This version has been peer-reviewed but does not include the final publisher proof corrections, published layout or pagination.

All articles available through Birkbeck ePrints are protected by intellectual property law, including copyright law. Any use made of the contents should comply with the relevant law.

Citation for this version:

Citation for the publisher's version:

E. Frias-Martinez¹, G. Magoulas², S. Chen¹, R. Macredie¹

¹Department of Information Systems & Computing, Brunel University, Uxbridge, Middlesex. UB8 3PH UK
²School of Computer Science & Information Systems, Birkbeck College, University of London, London WC1E 7HX U.K.

Abstract.

Adaptive Hypermedia systems are becoming more important in our everyday activities and users are expecting more intelligent services from them. The key element of a generic adaptive hypermedia system is the user model. Traditional machine learning techniques used to create user models are usually too rigid to capture the inherent uncertainty of human behavior. In this context, soft computing techniques can be used to handle and process human uncertainty and to simulate human decision-making. This paper examines how soft computing techniques, including fuzzy logic, neural networks, genetic algorithms, fuzzy clustering and neuro-fuzzy systems, have been used, alone or in combination with other machine learning techniques, for user modeling from 1999 to 2004. For each technique, its main applications, limitations and future directions for user modeling are presented. The paper also presents guidelines that show which soft computing techniques should be used according to the task implemented by the application.

Keywords: User Modeling, Adaptive Hypermedia, Soft Computing, Machine Learning, Data Mining.

1 Introduction

Adaptive Hypermedia (AH) can be defined as the technology that allows customization for each individual user of a hypermedia application (Perkowitz & Etzioni, 2000).

The architecture of an AH system is usually divided in two parts: the server side and the client side. The server side generates the user models from a database containing the interactions of the users with the system and the personal data/preferences that each user has given to the system. These user models, in combination with a hypermedia database, are used by the “Decision Making and Personalization Engine” module to identify user needs, decide on the types of adaptation to be performed and communicate them to an adaptive interface. Figure 1 presents the architecture of a generic AH system.

The process of personalization in an AH system is defined as the ways in which information and services can be tailored to match the unique and specific needs of an individual or a community (Callan et al., 2001). Personalization is about building customer loyalty by building a meaningful one-to-one relationship; by understanding the needs of each individual and helping satisfy a goal that efficiently and knowledgeably addresses each individual’s need in a given context (Rieken, 2000).

In this context, the user model is considered as a set of information structures designed to represent one or more of the following elements (Kobsa, 2001): (1) representation of assumptions about the knowledge, goals, plans preferences, tasks and/or abilities about one or more types of users; (2) representation of relevant common characteristics of users pertaining to specific user subgroups (stereotypes); (3) the classification of a user in one or more of these subgroups; (4) the recording of user behavior; (5) the formation of assumptions about the user based on the interaction history; and/or (6) the generalization of the interaction histories of many users into stereotypes (a stereotype is defined as a set of users that share a common behavior or interest).

The more information a user model has, the better the content and presentation will be personalized. A user model is created through a User Modeling (UM) process in which unobservable information about a user is inferred from observable information from that user; for example, using the interactions with the system (Zukerman, Albrecht & Nicholson, 1999). User models can be created using a user-guided approach, in which the models are directly created using the information provided by each user, or an automatic approach, in which the process of creating a user model is controlled by the system and is hidden from the user. This paper focuses on soft computing techniques to automate the acquisition and creation of user models in AH systems.

The problem of UM can be implemented using an automatic approach because a typical user exhibits patterns when accessing a hypermedia system and the set of interactions containing those patterns can be stored in a log database in the server. In this context, machine learning techniques can be applied to recognize regularities in user trails and to integrate them as part of the user model. Machine learning encompasses techniques where a machine acquires learns knowledge from its previous experience (Witten & Frank, 1999). The output of a machine learning technique is a structural description of what has been learned that can be used to explain the original data and to make predictions. From this perspective, data mining and other machine learning techniques make it possible to automatically create user models for the implementation of AH services. Pierrakos et al. (2003)
SC technologies provide an approximate solution to an ill-defined problem and can create user models in an environment, such as a hypermedia application, in which users are not willing to give feedback on their actions and/or designers are not able to fully define all possible interactions. Human interaction is a key component of any hypermedia application, which implies that the data available will usually be imprecise, incomplete, and heterogeneous. In this context, SC seems to be the appropriate paradigm to handle the uncertainty and fuzziness of the information available to create user models (Pal, Talwar & Mitra, 2002). The elements that a user model captures (goals, plans, preferences, common characteristics of users) can exploit the ability of SC to mix different behaviors and to capture human decision processes in order to implement a system that is more flexible and sensible in relation to user interests. Different techniques provide different capabilities. For example, Fuzzy Logic provides a mechanism to mimic human decision-making that can be used to infer goals and plans; Neural Networks a flexible mechanism for the representation of common characteristics of a user and the definition of complex stereotypes; Fuzzy Clustering a mechanism in which a user can be part of more than one stereotype at the same time; and NeuroFuzzy systems a mechanism to capture and tune expert knowledge which can be used to obtain assumptions about the user. These techniques can be used to construct a user model by themselves or in combination with traditional machine learning techniques.

This paper will explore the development of user models using soft computing techniques from 1999 to 2004, focusing on the main journals and conferences for UM, mainly: User Modeling and User-Adapted Interaction Journal; Expert Systems with Applications; International Conference on User Modeling; International Conference on AH; IEEE Transactions on Neural Networks; Workshop of Intelligent Techniques for Web Personalization (part of ICAI-International Joint Conference of Artificial Intelligence); and International Workshop on Knowledge Discovery on the WEB (WEBKDD, part of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining). The paper’s intentions are (1) to give an up-to-date view of Soft Computing techniques to UM and highlight their potential advantages and limitations, and (2) to give basic guidelines about which techniques can be useful for a given adaptive application.

The organization of the paper is as follows. The paper first gives a taxonomy of user models using two parameters: granularity and task implemented. After that, each SC technique is briefly introduced, giving also examples of applications for UM, and highlighting its pros and cons in the field of UM. Next we develop guidelines for how to choose a soft computing technique to create a user model according to the needs of the AH application that is going to be implemented. The conclusion section closes the paper by presenting future directions.


User models are classified in this work based on how human behaviors are represented as models, and their purpose. To this end, two main dimensions are considered: (1) the granularity of the model, a model can be created for each individual user (content-based modeling) or for clusters of users (collaborative modeling); and (2) the type of task for which the model is going to be used. Below, we define four basic types of tasks: (i) Prediction, (ii) Recommendation, (iii) Classification and (iv) Filtering. Prediction is the capability of anticipating user needs using past user behavior. A basic assumption is made with this approach: a user’s immediate future is very similar to his/her immediate past. In the literature this is traditionally presented as content-based filtering. Recommendation is the capability of suggesting interesting elements to a user based on some extra information not based on the past behavior of the user, for example, from the items to be recommended or from the behavior of other users. In this context, recommendation is what in the literature is known as collaborative filtering. Classification builds a model that maps and Ernaki and Vaziriannis (2003) present a review of how traditional data mining techniques can be applied to UM and the general architecture of such systems. Nevertheless, traditional machine learning techniques have some limitations for modeling human behavior, mainly the lack of any reference to the inherent uncertainty that human decision-making has. This problem can be partially solved with the introduction of Soft Computing (SC) for UM. SC is an innovative approach to building computationally intelligent systems that differs from conventional (hard) computing in that it is tolerant of imprecision, uncertainty and partial truth. The guiding principle of soft computing is to exploit the tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low solution cost (Sinha, Gupta & Zadeh, 2000). SC consists of several computing approaches, including neural networks, fuzzy set theory, approximate reasoning, and search methods, such as genetic and evolutionary algorithms (Jang, Sun & Mizutani, 1997).
or classifies data items into one of several predefined classes. Filtering is defined as the selection of a subset of items that are interesting to a user from the original set of items.

In general, any of the previous tasks can be implemented using knowledge stored in the different user model elements (Kobsa, 2001). For example, a filtering task can be implemented using knowledge represented by some user preferences, or by classifying the user in a stereotype (or in more than one stereotypes). A prediction task can be implemented using knowledge captured by user’s goals but also by classifying the user in a stereotype, etc. In the following subsections we present a number of SC techniques and give examples of AH systems that employ the particular technique, specifying the task implemented and the granularity of the model.

2.1 Fuzzy Logic

Fuzzy Logic (FL) defines a framework in which the inherent ambiguity of real information can be captured, modeled and used to reason under uncertainty. A key concept in FL theory is the notion of the fuzzy set. A fuzzy set expresses the degree of membership of an element in that set. When compared to traditional binary or multi-valued logic, in which the degree of truth takes values from a discrete finite set, in fuzzy logic the degree of truth can take continuous values between [0,1]. This characteristic allows capturing the uncertainty inherent to real data. An introduction to FL can be found in Klir & Yuan, (1995) and Yan, Ryan & Power (1994).

FL is not a machine learning technique; nevertheless, due to its ability to handle uncertainty, it is used in combination with other machine learning techniques in order to produce behavior models that are able to capture and manage the uncertainty of human behavior. Some examples of these combinations are Fuzzy Clustering, Fuzzy Association Rules, and Fuzzy Bayesian Networks. Another alternative is to capture user models with a machine learning technique (possibly with some kind of representation of uncertainty) and use FL inference to implement the personalization engine. A traditional FL inference system processes knowledge in three steps (1) fuzzifies the input data; (2) conducts fuzzy inference based on fuzzy information; and (3) defuzzifies the fuzzy decisions to produce the final outcome. FL in UM does not necessarily realize all of the three steps, but maybe only a subset of them.

Typically FL has been employed in recommender systems. In these applications, FL provides the ability to mix different user preferences and profiles that are satisfied to a certain degree. An example of fuzzy inference used for recommendation is Nasraoui & Petenes (2003), which uses user profiles obtained with hierarchical unsupervised clustering. In Ardioso & Goy (1999) fuzzy logic is used to model user behavior and provide recommendations using this fuzzy behavior model. Although, strictly speaking, there is no actual fuzzy inference involved, the stereotypes that characterize users are modeled using membership functions, and the recommendation process is done using a fuzzy AND operator. Schmitt, Dengler & Bauer (2003) present a system designed to recommend products in an e-commerce site, according to how well this product satisfies user preferences. The score of an item (according to how much that item matches user interests) is done using an OWA (Ordered Weighted Averaging) operator. This family of operators allows the representation of fuzzy logic connectives and the aggregation
Table 2.
Characteristics of some Neural Networks-based User Modeling applications.

<table>
<thead>
<tr>
<th>Application</th>
<th>Training Data</th>
<th>Outcome</th>
<th>T</th>
<th>I-G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bidel et al. (2003) Classification and tracking of user navigation.</td>
<td>Data generated from an online encyclopedia.</td>
<td>A labeled approach to the problem produces better accuracy.</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>Sas et al. (2003) Prediction of user’s next step in a virtual environment</td>
<td>30 users performed exploration and searching within the environment.</td>
<td>Very accurate predictions of the next step.</td>
<td>R</td>
<td>G</td>
</tr>
<tr>
<td>Beck &amp; Wooff. (1998) Construction of a student model for an intelligent tutoring system.</td>
<td>Data collected by the tutoring system</td>
<td>NN-based recommendation to each individual.</td>
<td>R</td>
<td>G</td>
</tr>
<tr>
<td>Shavlik &amp; Eliaﬃ (2001) Adaptive agents that retrieve and extract information by accepting user preferences in the form of instructions.</td>
<td>Instructions given directly by the user and user rated web pages.</td>
<td>Facilitates creating intelligent agents combining user instructions with machine learning.</td>
<td>F/P</td>
<td>I</td>
</tr>
<tr>
<td>Rohl et al. (2003) Three step recommendation model based on collaborative filtering that combines NN with case-based reasoning.</td>
<td>Movilens data sets (GroupLens Research Project, Univ. of Minnesota) containing ratings of movies.</td>
<td>The new algorithm gives useful recommendations to each user.</td>
<td>R</td>
<td>G</td>
</tr>
<tr>
<td>Changchien &amp; Lu (2001) On-line recommendation system for e-commerce sites based on customer and products fragmentation.</td>
<td>Sample of sales records from a Database.</td>
<td>Recommendation knowledge can promote internet sales.</td>
<td>R</td>
<td>G</td>
</tr>
</tbody>
</table>

of different user preferences. Kuo and Chen (2004) design a recommendation system for electronic commerce using fuzzy rules obtained by a combination of fuzzy neural networks and genetic algorithms. FL has also been used for filtering (Vrettos & Stafylopatis, 2002). In this case, FL provides a soft filtering process based on the degree of concordance between user preferences and the elements being filtered. Table 1 summarizes relevant studies and applications of FL for UM. The columns detail the application, the data, the results obtained, the type of task (T) for which the SC technique was used, i.e. Prediction (P), Recommendation (R), Classification (C) and Filtering (F), and (I-G) if the system created a model for each Individual (I) or for Groups of users (G).

Although FL is an ideal technique for modeling human reasoning, it faces some challenges in real-world applications. The main one is related to the fact that it possesses no mechanism for learning from data. This implies that the knowledge of the application domain has to be explicitly given by the designer. Moreover, it also has an impact on the definition of other model parameters like membership degrees and fuzzy operators, which are in general application dependent. Neuro-fuzzy systems, which will be discussed in another section, have emerged as an approach to alleviate these challenging situations.

2.2 Neural Networks

A Neural Network (NN) is an information processing paradigm that is inspired by the way of biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. Comprehensive introductions to Neural Networks can be found in (Faussett, 1994) and (Haykin, 1999). NNs are able to derive meaning from complicated and/or imprecise data and to extract patterns that are too complex to be noticed by other computational techniques. No initial knowledge about the problem that is going to be solved is needed. These characteristics make NNs a powerful method to model human behavior and an useful technique to create user models for hypermedia applications.

NNs have been extensively used for user modelling, mainly for classification and recommendation in order to group together users with the same characteristics and create profiles. Some examples are Bidel, Lemoine & Piatt (2003), which uses NN to classify user navigation paths and Hsieh (2004) that classifies bank users. Self Organizing Maps (SOM) is a type of unsupervised NN that has also been extensively used for recommendation, because it transforms highly dimensional data into a two dimensional grid, grouping elements with the same characteristics. Goren-Bar et al. (2001) use SOM to classify documents based on a subjectively predefined set of clusters in a specific domain. Rohl, Oh & Han (2003) also use SOM to create a recommendation system for movies and Changchien & Lu (2001) use SOM to create a recommendation system for e-commerce.


Table 3.
Characteristics of some Genetic and Evolutionary- based User Modeling applications.

<table>
<thead>
<tr>
<th>Application</th>
<th>Training Data</th>
<th>Outcome</th>
<th>T</th>
<th>KG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min et al. (2001)</td>
<td>Profiling behavior of e-commerce customers.</td>
<td>GA are useful for the discovery of profiles of e-commerce customers.</td>
<td>R</td>
<td>G</td>
</tr>
<tr>
<td>Romero et al. (2003)</td>
<td>Discovering prediction rules from student usage information to improve web courses.</td>
<td>The rules produced are better than traditional rule extraction algorithms.</td>
<td>R</td>
<td>G</td>
</tr>
<tr>
<td>Fan et al. (2000)</td>
<td>Personalization of search engines using automatic term weighting.</td>
<td>GA Automatic weighting improves the retrieval performance quite dramatically.</td>
<td>F</td>
<td>G</td>
</tr>
<tr>
<td>Shin &amp; Lee (2002)</td>
<td>Application of Gas to bankruptcy prediction modeling.</td>
<td>Preliminary results show that rule extraction approach using Gas for bankruptcy prediction modeling is promising.</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>Lee &amp; Tsai (2003)</td>
<td>User modeling for enhance the quality of web searches.</td>
<td>Experimental results show that the framework developed is efficient and useful to enhance the quality of web search.</td>
<td>R</td>
<td>I</td>
</tr>
</tbody>
</table>

NNs have also been used for recommendation in (Sas, Reilly & O’Hare, 2003), which predicts the next step for a given user trajectory in a virtual environment, in (Beck et al., 2003) and (Beck & Woolf, 1998) which model student behavior for an intelligent tutoring system and in (Buczak, Zimmerman & Kuparati, 2002) which uses a NN to fuse the recommendations given by a set of personalized TV recommenders.

Nevertheless, due to their ability to capture any kind of knowledge, they have also been used for filtering and prediction tasks, like in Shepperd, Waters and Marath (2002) and Shavlik and Eliassi (2001) respectively. Table 2 summarizes some applications of Neural Networks for UM.

Despite the number of successful applications, NNs still face some important limitations in UM. The main ones are the training time needed to produce a NN-based model (which in certain cases can be measured in the order of many hours or even days) and the amount of data needed. The training time may cause inconvenience when creating models dynamically. Although there are techniques able to retrain NNs, in UM the NN is usually retrained from scratch when, for example, more information comes available, a new user or a new document is added on the AH system. More research in the field of incremental NN learning is needed. Another drawback of NNs is the limited interpretability of their decisions. While other techniques, to different extents, can be interpreted and manually changed, NNs knowledge representations are not easily interpretable, and as a result their application is avoided in cases where human-understandable user models are needed.

2.3 Genetic and Evolutionary Algorithms

Genetic Algorithms (GAs), (Goldberg, 1989), and Evolutionary Algorithms (EAs), (Schwefel, 1995) are search algorithms based on the mechanics of natural selection. They begin with a set of potential solutions called the population. Solutions from one population are taken and used to form a new population, which are closer to the optimum solution to the problem at hand. The idea behind this process is the survival of the fittest.

In general, GAs and EAs have been used for Recommendation in the form of rules, which can capture user goals and preferences, because they perform a global search and cope better with attribute interaction than algorithms used in data mining, where the search is more local. Examples of this approach are (Romero, Ventura & de Bra, 2003) for student modeling, (Min, Smolinski & Boraty, 2001) for profiling of e-commerce customers and (Lee & Tsai, 2003) for capturing users preferences for improvement of web searches.

They have also been applied for filtering (Fan, Gordon & Pathak, 2000) and for classification, as in (Shin & Lee, 2002), which uses GAs to model bankruptcy prediction of companies. Table 3 summarizes relevant applications of Genetic and Evolutionary algorithms for UM. This approach is suitable for searching vast, complex, and multimodal problem spaces but may have some limitations with respect to its potential for dynamic modeling and its computational complexity.

2.4 Fuzzy Clustering

The task of clustering is to structure a given set of unclassified instances (data vectors) by creating concepts, based on similarities found on the training data. A clustering algorithm finds the set of concepts that cover all examples verifying that: (1)
Table 4.

Characteristics of some Fuzzy Clustering-based User Modeling applications.

<table>
<thead>
<tr>
<th>Application</th>
<th>Training Data</th>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lampinen &amp; Koivisto (2002)</td>
<td>Obtain application profiles from network traffic</td>
<td>FCM produced better results than SOM. A method for the comparison of both solutions is also introduced.</td>
</tr>
<tr>
<td></td>
<td>data to manage network resources.</td>
<td>R G</td>
</tr>
<tr>
<td>Nasraoui et al. (1999)</td>
<td>A new algorithm (CARD) to mine user profiles</td>
<td>CARD is very effective for clustering many different profiles in user sessions.</td>
</tr>
<tr>
<td></td>
<td>from access logs is proposed.</td>
<td>R G</td>
</tr>
<tr>
<td>Joshi et al. (2000)</td>
<td>Two algorithms to mine user profiles: FCM dd and</td>
<td>Both algorithms extract interesting user profiles. FCM is not able to handle noise as effectively as FCTMdd.</td>
</tr>
<tr>
<td></td>
<td>FCTMdd.</td>
<td>C G</td>
</tr>
<tr>
<td>Krishnapura et al. (2001)</td>
<td>Web access log analysis for user profiling using</td>
<td>RFCMdd is very effective for clustering of relational data.</td>
</tr>
<tr>
<td></td>
<td>R FCMdd (Robust Fuzzy c-Medoids).</td>
<td>C G</td>
</tr>
<tr>
<td>Shin &amp; Sohn (2004)</td>
<td>Segmentation of customers to find properly graded</td>
<td>Fuzzy clustering analysis is a robust approach to model customers.</td>
</tr>
<tr>
<td></td>
<td>stock market brokerage/commission rates.</td>
<td>C G</td>
</tr>
</tbody>
</table>

The similarity between examples of the same concepts is maximized and (2) the similarity between examples of different concepts is minimized. In a clustering algorithm the key element is the concept of distance used to obtain the similarity between two items of the training set. In non-fuzzy or hard clustering, data is divided into crisp clusters, where each data point belongs to only one cluster. In Fuzzy Clustering (FC), the data points can belong to more than one cluster, and each data point is associated with a set of membership grades that indicate the various degrees this point belongs to the different clusters. The most widely used fuzzy clustering algorithm is the Fuzzy C-Means (FCM) Algorithm (Bezdek, 1981). There are other algorithms, which are basically variations of the original FCM, like the Fuzzy c-Medoids Algorithm (FCMdd) or the Fuzzy c-Trimered Medoids Algorithm (FCTMdd) (Krishnapuram et al., 2001).

For UM, there are two kinds of interesting clusters to be discovered: usage clusters and page clusters. Clustering of users tends to establish groups of users exhibiting similar browsing patterns. Such knowledge is especially useful for inferring user demographics or interests in order to perform market segmentation in e-commerce applications or provide personalized Web content or notifications to the users. On the other hand, clustering of pages will discover groups of pages having related content. This information is useful for Internet search engines and Web assistance providers.

FC for UM is mostly used for recommendation and classification tasks. (Lampinen & Koivisto, 2002), (Nasraoui et al., 1999) and (Nasraoui & Krishnapuram, 2000) are examples of applications that implement a recommendation task using FC. Examples of classification tasks are (Joshi et al., 2000), (Krishnapuram et al., 2001) and (Shin & Sohn, 2004). Table 4 summarizes a range of studies and applications of FC for UM.
The main problem that clustering techniques face is how to define the concept of distance that is going to be used. In general some knowledge of the problem is needed to define the concept of distance in the best available way.

When applied to user modelling this problem is even harder due to the nature of the data available: interactions, user preferences, pages visited, etc., which may not be available in numerical form. Different techniques to characterize web user behavior using numerical vectors have been proposed (Joshi et al., 2000, Mobasher et al., 2000), but in one way or another, the representations lose part of the semantics that the original data had. Also, user models developed so far using fuzzy clustering do not fully use the fuzzy nature of the technique in order to create more flexible and adaptive systems. As a consequence, more work is needed to create meaningful ways of mixing the different personalization features and techniques associated with each one of the clusters in which a user can be included when using fuzzy clustering, and evaluate the benefits of this approach.

2.5 Neuro-Fuzzy Systems

Neuro-Fuzzy Systems (NFS) use NNs to learn and fine tune rules and/or membership functions from input-output data to be used in a Fuzzy Inference System. With this approach, the drawbacks of NNs and FL, the black box behavior of NNs and the problems of finding suitable membership values for FL are avoided. NFS automate the process of transferring expert or domain knowledge into fuzzy rules (Jang & Sun, 1995) and (Jang & Sung, 1997) describe in more detail the basic concepts of NFS. One of the most important NFS is ANFIS (Jang, 1993), which has been used in a wide range of applications (Bonisone, Badami & Chiang, 1995). NFS are especially suited for applications where user interaction in model design or interpretation is desired. NFS are basically FL systems with an automatic learning process provided by NN.

The combination of NN and fuzzy sets offers a powerful method to model human behavior which allows NFS to be used for a variety of tasks. Lee (2002) and Stathacopoulos Grigoriadou & Magoulas (2003) use a NFS for recommendation in an e-commerce site and in an on-line course, respectively. Drigas et al. (2004) provide another example of recommendation task. In this case, jobs are assigned to unemployed people considering user and enterprises profile data. Magoulas, Papanikolau & Grigoradou (2001) use NFS to implement classification/recommendation system with the purpose of planning the contents of a web-course according to the knowledge level of the student. George and Cardulillo (1999) use NFS for prediction tasks within a simulated aircraft control. Table 5 summarizes studies and applications of NFS for UM.

NFS have been designed to retain the positive aspects of NN and FL, nevertheless it still maintains some of the limitations of both approaches, mainly the training time needed and application for dynamic modeling.

3 Guidelines for the Selection of Techniques

The preceding discussion has demonstrated that each technique captures different relationships among the data available and expresses it using different data structures. In this section, we present guidelines to help decide which technique to use when developing an AH application.

Table 6 summarizes the characteristics of the techniques presented along seven dimensions. The first four dimensions capture

<table>
<thead>
<tr>
<th>Application</th>
<th>Training Data</th>
<th>Outcome</th>
<th>T</th>
<th>UG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lee (2001)</td>
<td>Mobile web shopping agent that finds products that suit user needs using a</td>
<td>A test is implemented using a product database with 200 items and 8</td>
<td>R</td>
<td>I</td>
</tr>
<tr>
<td></td>
<td>NFS and FL.</td>
<td>categories.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stathacopoulos et al. (2003)</td>
<td>Student Modeling</td>
<td>A set of simulated students.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Magoulas et al. (2001)</td>
<td>Intelligent decision making for recommending educational content in a web-based</td>
<td>“Introduction to Computer Science” course of the Univ. of Athens.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>course depending on knowledge level.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>George &amp; Cardulillo (1999)</td>
<td>Modeling of human behavior.</td>
<td>10 subjects collected data for the one dimensional compensatory</td>
<td>P</td>
<td>G</td>
</tr>
<tr>
<td></td>
<td></td>
<td>task.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
the main problems that machine learning for UM faces according to Webb, Pazzani & Billsus (2001): Computational Complexity of the learning task; Dynamic Modeling, which indicates the suitability of the technique to change a user model on-the-fly; Labeled/Unlabeled, which reflects the need of labeled data; and size of training data, which reflects the amount of data needed to produce a reliable user model. The remaining dimensions present other relevant information: the ability of the techniques to handle uncertainty in modeling human behavior (Uncertainty), i.e., to produce a user model that takes into account the inherent fuzziness of UM; the ability to handle noisy data (Noise), i.e., how noisy training data will affect the user model produced. Lastly, the interpretability (Interpretability) of the results, i.e., how easy it is for a human to understand the knowledge captured, is considered a critical dimension, as interpretability is often cited as a critical problem that traditional machine learning and data mining methods encounter in UM applications (Li & Zhong, 2004; Kim & Nick Street, 2004).

For example, NNs have a high complexity in training, although they can provide a real-time response time during their operation. NFS have a Medium High interpretability, which depends on the architecture of the system. For example, ANFIS produce systems with high interpretability. Traditional GAs are not able to cope with dynamic modeling problems, nevertheless some recent approaches present dynamic modeling using evolutionary computation for specific problems (Stemberg & Reynolds, 1997).

Table 6.

| Characteristics of different Soft Computing techniques applied to User Modeling. |
|------------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Fuzzy Logic      | High           | No             | Both           | High           | Yes            | Yes            | High           |
| Neural Networks  | High           | No             | Both           | High           | No             | Yes            | Low            |
| Genetic/Evolutionary | High         | No             | Both           | Med/High       | Yes            | Yes            | Low            |
| Fuzzy Clustering | High/Low       | No             | Both           | Med/High       | Yes            | Yes            | Med/High       |
| Neuro-Fuzzy      | High/Low       | Yes            | Labeled        | Med/High       | Yes            | Yes            | Med/High       |

For user-adaptive system, Table 7 presents a classification of SC techniques with respect to possible interpretability requirements looking at the four main types of task introduced in Section 2, i.e., Prediction; Recommendation; Classification and Filtering. Table 7 considers two possible values for Interpretability: needed or not relevant. The first one expresses the necessity of having a human understandable output while the second one states that this factor is not important. It should be made clear that the classification of Table 7 does not necessarily mean that the techniques cannot be used to implement other types of task. For example, NNs are basically used for tasks that need no interpretability. However, methods to extract the knowledge embedded in NN are available in the NN literature (Ticlek et al., 1998). The combination of Tables 6 and 7 can be used to guide the choice of which technique to be used for UM in an adaptive system. First, Table 7 can be used to identify the set of techniques that satisfy the interpretability requirements of the adaptive application and, after that, Table 6 can be used to refine the search and take the final decision.

4 Conclusions

Hypermedia systems are becoming more important in our everyday activities and their contents and services are ever more varied. This is the main reason why users expect more intelligent and personalized services every time they use a hypermedia system. The key element necessary to provide such intelligent services is the concept of a user model. Due to the variety and amount of information available to create user models, data mining and machine learning techniques can be used to automatically identify user patterns and interests. Nevertheless, traditional machine learning and data mining techniques are not able to capture the inherent uncertainty of human behavior modeling. In this context, soft computing techniques arise as a powerful tool for automatically generate efficient user models for personalization.

This paper has presented a review of recent approaches to UM within the area of AH systems that employ soft computing techniques. The review demonstrates that one of the main problems that the development of AH faces is the lack of any kind of
standardization for the design of user models. In order to improve this situation this paper has tried to give a set of guidelines that formalize the design of user models using a SC approach.

It is our opinion that the future of UM is in hybrid approaches. The most successful part of examples of applications reviewed already combine some form of soft computing with other soft computing techniques, traditional machine learning techniques or symbolic knowledge representation techniques. For example, synergistic approaches that combine neural networks and fuzzy logic, or neural networks and genetic algorithms, neural networks and association rules, neural networks and case-based reasoning, clustering and fuzzy logic or genetic algorithms and rule extraction show great potential for UM. The combination of these SC techniques among themselves and with other machine learning techniques will provide a useful framework to efficiently model the natural complexity of human behavior.

Acknowledgments

The work presented in this paper is funded by the UK Arts and Humanities Research Board (AHRB grant reference: MRG/AN9183/APN16300).

References


