Automated User Modeling for Personalized Digital Libraries

E. Frias-Martinez[1], G. Magoulas[2], S. Chen[1]* , R. Macredie[1]

[1]*Department of Information Systems & Computing, Brunel University, Uxbridge, Middlesex, UB8 3PH U.K.*

[2]*School of Computer Science & Information Systems, Birkbeck College, University of London, Malet Street, London WC1E 7HX U.K.*

Abstract.

Digital libraries (DL) have become one of the most typical ways of accessing any kind of digitalized information. Due to this key role, users welcome any improvements on the services they receive from digital libraries. One trend used to improve digital services is through personalization. Up to now, the most common approach for personalization in digital libraries has been user-driven. Nevertheless, the design of efficient personalized services has to be done, at least in part, in an automatic way. In this context, machine learning techniques automate the process of constructing user models. This paper proposes a new approach to construct digital libraries that satisfy user's necessity for information: Adaptive Digital Libraries, libraries that automatically learn user preferences and goals and personalize their interaction using this information.

Keywords: Digital Libraries, User Modeling, Personalization, Adaptive Library Services

1. Introduction

The term "digital libraries" became popular about fifteen years ago, although the concept behind digital libraries existed before the term was introduced. There is no clear consensus on the definition of Digital Libraries (DL), but, in general, they can be defined as collections of information that have associated services delivered to user communities using a variety of technologies (Callan, Smeaton, Beauliei, Borlund, & Brusilovsky 2003). The collections of information can be scientific, business or personal data and can be represented as a digital text, image, audio, video or other media. Due to the amount and great variety of information stored by DL, they are becoming more important in our everyday activities and their contents and services are every day more varied. This relevance of DLs has caused users to expect more intelligent services every time they access and search information. One of the key elements on which these intelligent services are based is personalization.

Personalization 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., 2003). Personalization is about building customer loyalty by creating 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 (Riecken, 2000). The key element of a personalized environment is the user model. A user model is a data structure that represents user interests, goals and behaviors. The more information a user model has, the better the content and presentation will be tailored for each individual user. A user model is created through a user modeling 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 hidden from the user. The hypermedia systems constructed using the user-guided approach

* Corresponding author. Tel: +44 (0) 1895 266023; Fax: +44 (0)1895 251686 (Sherry Chen)
E-mail address: sherry.chen@brunel.ac.uk

are called adaptable (Fink, Kobsa, & Nill, 1997), while the ones produced using an automatic approach are called adaptive (Fink, Kobsa, & Nill, 1997; Brusilovsky & Schwarz, 1997).

Within the context of DL, up to now, user modeling has been implemented using mainly user-guided approaches, which has produced adaptable DL. Nevertheless the problem of user modeling in DL can be easily implemented using an automatic approach because a typical user exhibits patterns when accessing DLs and the information containing these patterns is already usually stored in databases. For this purpose, machine learning techniques can be applied to recognize such regularities in order to integrate them as part of the user model. Machine learning encompasses techniques where a machine acquires '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, machine learning techniques make it possible to automatically create user models for the implementation of personalized digital library services.

We consider that the user's requirement for more efficient and tailored services when using a DL can be fulfilled using personalized DL based on user models that are automatically constructed using machine learning techniques: with adaptive DL. The paper's intentions are (1) to introduce the adaptive dimension of a DL, (2) to present the potential of applying machine learning to create Adaptive DLs and (3) to give basic guidelines about how to automatically create DL user models. The paper is organized as follows: first, we present the architecture, functionalities and state of the art of personalized DL. Once the main problems of the current approaches have been highlighted, Section 3 presents the adaptive dimension of a personalized DL, describing also some approaches already taken to implement adaptive DL services. Section 4 describes the elements that a DL user model should contain and which techniques can be used to model and capture those elements. The paper ends with the conclusions section.

2. Personalized digital libraries

DLs are more than simple web pages that give access to information. They also comprise, among others, a structure for the organization of the information, metadata regarding the semantic of the information and knowledge about who uses them and for what purposes. This implies that, if usually designing a good web page is problematic, the process of designing a good digital library is even more complex due to the syntactic and semantic organization that is needed. In general, DLs are made up of four components (Theng, Duncker, & Mohd-Nasir, 1999): (1) information; (2) structure, describing the syntactic and semantic characteristics of the information provided by the DL; (3) interaction elements, referring to the searching interface, screen design, etc.; and (4) properties, referring to security, copyright issues, etc., of the information available in the DL. The services provided by DL through their interaction elements (interface) can be classified into three groups:

- Mechanisms for the personalization of content. These mechanisms make it possible for each user to create a personal DL that contains only the information that is interesting and relevant to that user
- Mechanisms to help in the process of navigation. These services present each user with an environment that better suit the way in which that user interacts with the DL.
- Information filtering (IF) and information retrieval (IR) mechanisms. These services provide ways to find and filter the vast amount of information that a user accesses and receives.

Although these three basic types of services provide the basic functionality needed by a DL user, they can be improved by the introduction of personalization. Personalization will create more tailored

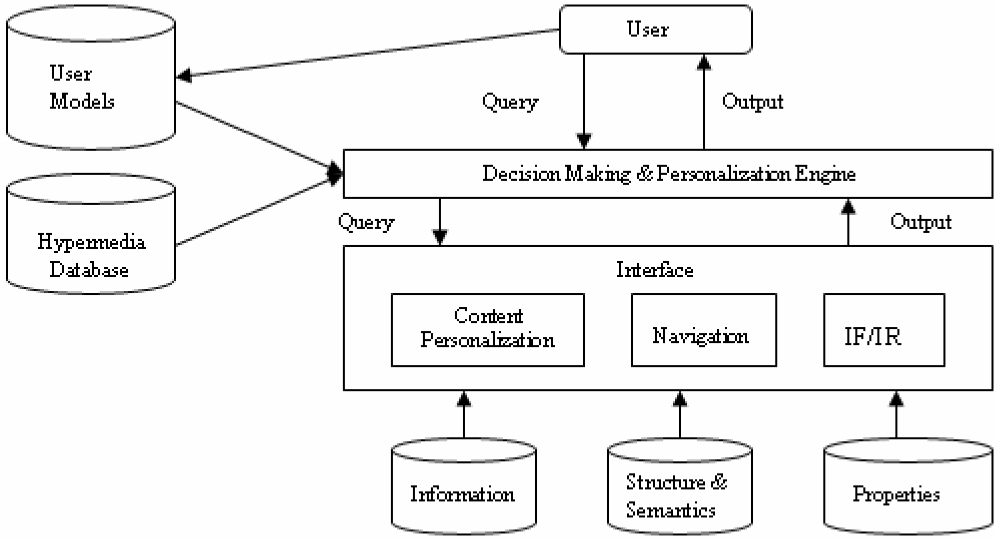

Fig. 2. Generic Architecture of a personalized adaptable DL.

services that help and simplify the process of finding relevant information by using the content stored in each user model. Formally, a user model is 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. In the context of user modeling a stereotype is defined as a cluster of users that share a common behavior.

Typically, personalization in DL has been user-driven. In this approach the user specifies his/her preferences directly to the DL, from the color background of the page, to the layout of the components or to the content of the information presented. All this information is stored in the User Model of that particular user and is used by the services provided by the DL to tailor the output produced. This approach, in which a user has to directly specify his/her preferences, produces adaptable DL services and adaptable DL. Figure 2 presents the architecture of an adaptable DL in which the elements and services previously defined are presented. As can be seen in Figure 2, in a personalized DL the output to a user's query is not provided directly by the interface but through the combined action of a decision-making mechanism and a personalization engine that adapts the contents and the presentation according to a user model. Also, in this case, the user model is exclusively created using information directly provided by the user, which confers the adaptable nature of the personalized DL.

2.1 State of the art of personalized DL

The first developments for personalization in DL are basically different implementations of MyLibrary. MyLibrary provides basic personalization mechanisms regarding information retrieval and content personalization (Kohen et al., 2000; Winter, 1999), where all those processes are user-driven. There are a lot of different implementations of Mylibrary: MyLibrary@LANL Research library (Di Giacomo et al., 2001), My.UCLA (Winter, 1999) and MYLibrary@NCState, for example. The theoretical background for the concepts used by MyLibrary is given by the concept of Personalized Information Environment (PIE) (French & Viles, 1999; Jayawardana, Hewagamage, & Hirakawa, 2001). A PIE in a DL is a framework that provides a set of integrated tools based on an individual user's requirements with respect to his/her access to library materials. The following subsections describe different implementations of personalized DL services categorizing them into the three basic services provided by a DL: Personalization of Content, Interface Personalization and Personalization for Information Filtering and Information Retrieval.

2.1.1 Personalization of content

Different content personalization tools have been provided by the different MyLibrary implementations. In general these different tools have a set of elements in common: (1) they are always user-guided and (2) the information is always stored in folders where each folder contains a set of links. The main tools for content personalization are (Di Giacomo et al., 2001):

- Bookmarklets: Bookmarklets are like bookmarks, but instead of storing a static web link it stores a command. Bookmarklets can be added to the chosen folder of the personal catalogue (or personal library) during web navigation. Also bookmarklets are usually implemented with a link checking mechanism.
- Shared Libraries: In this case a library (catalogue) is owned by more that one user which can access and modify its content.
- Protection mechanisms: user name and encrypted passwords.

Different examples of the previous personalization tools can be found in Virginia Commonwealth University (www.library.vcu.edu/mylibrary), North Carolina State University, (my.lib.ncsu.edu), University of California Los Angeles (my.ucla.edu), Cornell University (www.mannlib.cornell.edu/mannorama/carrel). PADDLE (Hicks & Tochtermann, 2001) (Personal Adaptable Digital Library Environment) is another example of a personalization architecture for DL that provides some of the tools previously described.

2.1.2 Interface personalization

DLs have a basic set of mechanisms to personalize navigation. These mechanisms are common to any other interface personalized web pages. Typical services are customization of the interface by choosing among several colors, to order and rearrange libraries, folders, text color and size, link color, background colors, the information that is and is not presented, etc. The user creates a user profile that expresses his/her choices for interface personalization. A typical example of interface personalization is MyYahoo! (Manber, Patel, & Robinson, 2000), which was also one of the first personalized commercial sites. In MyYahoo! users can select from a set of modules, such as news, stock prices, weather and sports, place them in one or more web pages, arrange where within the page the information is presented, and specify the frequency with which the information is updated. Personalized interfaces have also extensively used in e-commerce sites and e-banking.

2.1.3 Personalization in information retrieval (IR) and information filtering (IF)

Information Filtering (IF) and Information Retrieval (IR) are two similar processes aimed at providing a user with relevant information (Belkin & Croft, 1992). The main difference of both processes is how information reaches the user. IR is an active process in which a user actively tries to find relevant information, typically by using search mechanisms, while, in IF, information tries to find the user. In these processes a set of filters define the concept of interesting information.

DLs have a basic mechanism of IR using keywords. This mechanism can be more or less complex depending on which other options are present: for example search only in a catalog or the web or combined, order the results by relevance, refine the search within the results obtained, etc. Typically those IR tools do not consider any user preference. Other IR mechanisms are population services (Di Giacomo et al., 2001) offered in order to find suitable journals and data bases when creating a personal library. These tools offer different mechanism to select the relevant information: (1) exploring the

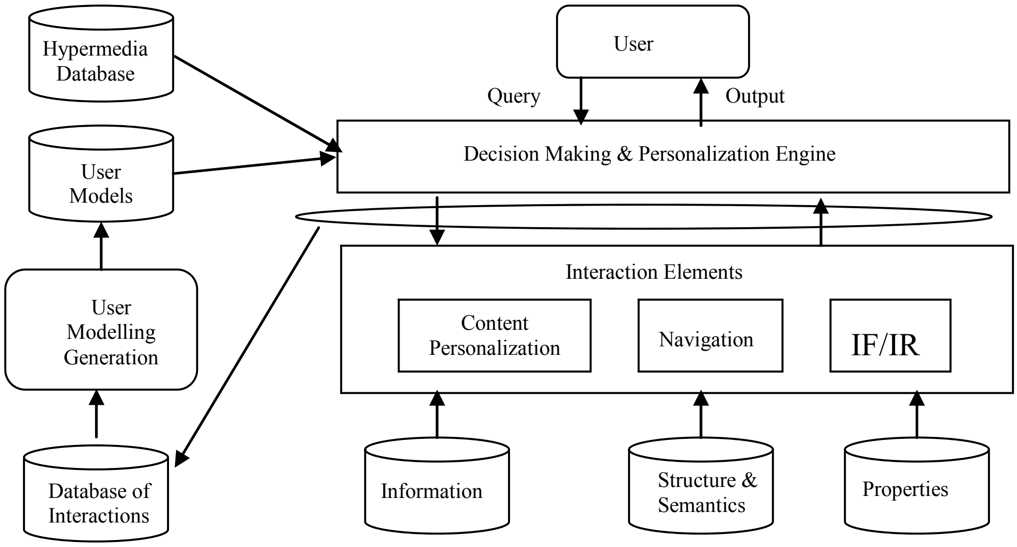

Fig. 3. Generic Architecture of an Adaptive DL.

classification of journals and selecting those that are interesting and (2) find journals using keywords. Typically DL have messaging services for users that provide messages related to the library like new journals, book due dates, holds and recalls, special events, etc. In some cases this messaging service can be personalized by the user where the user selects in which kind of information she or he is interested, which can be seen as an example of personalized and adaptable IF service. CYCLADES (Candela & Straccia, 2003) is a tool aimed at providing an integrated environment for users and groups of users (communities) that want to use, in a highly personalized and flexible way, electronic archives of documents. CYCLADES provides functionality for advanced search in large and heterogeneous archives, for collaboration, for filtering, for recommendation and for management of collections. The tool allows some degree of personalization in IR/IF processes by defining groups of users that share a common interest. Scirus (Scirus, 2004) (www.scirus.com) is a science-specific search engine that is able to filter all non-scientific sites and find peer-reviewed articles. The engine also has and advanced mechanism for IF that offers the possibility of refining the results obtained using as filtering words the more relevant key-words found in the recommended papers .

2.1.4 Limitations of the current approach

The previous sections have presented examples of personalized DL where the information of each user model is basically provided by each user. In all cases each user constructs his/her own user model and the DL uses this information in a static way. The main inconveniences that this approach has are:

- The concept of personalization cannot be necessarily understood by all the users of a DL.
- Users are not usually willing to give feed back to the system, even if it is for receiving a better service.
- Users do not necessarily know what their interests are and how they change over time, and can not provide information to the system.
- Even if the user is aware of his/her interests, the amount of information that today DL have make it unrealistic for a user to specify completely his/her preferences.

Although the personalized tools provided are useful there is still a gap between what a user expects from a personalized DL and what DLs are providing to each user. We think that the next level of DL services will be given by personalized services implemented using user models that are automatically constructed using machine learning techniques, because the application of these techniques will solve the limitations that a user-driven approach has. Although machine learning techniques have been extensively used in e-commerce sites (mainly for recommendation purposes), up to now, their implementation in DL has been very limited.

3. Adaptive dimension of personalized DLs

The adaptive dimension of a personalized DL refers to the ability of a DL to construct a user model without the direct intervention of the user. This automatic approach allows to implement adaptive DL services and creates adaptive DL, in which the system identifies user preferences, in contrast with the adaptable approach in which the user was supposed to specify his/her preferences. In this context, user models are obtained using machine learning techniques that are able to detect user patterns using as input data the interaction between the user and the DL. Figure 3 presents the architecture of an adaptive DL. As can be seen, when compared with the architecture of an adaptable DL, the main difference is that in this case the database of user models is created by a User Model Generation module that has as input a database containing the interactions between the set of users and the library. This automatic approach allows to observe users in an unobtrusively way and solves the problem that the adaptive approach has: (1) the user does not need to understand what personalization is because the system creates the user model, (2) this approach makes possible to create user models in an environment such as DL in which users are not willing to give feedback of their actions, (3) the system is responsible for discovering user preferences and how these change over time and (5) the automatic approach makes possible to deal with the amount of information that DLs have.

Although this automatic approach solves the main problems that the user-driven (or adaptable) approach has, it still faces some problems: (1) at the beginning the system does not have any information about the user, which means that some standard personalization should be used, (2) the machine learning techniques used should be scalable in order to be able to cope with the millions of users that a system can have, (3) ideally those techniques should be incremental in order to avoid the construction of user models from scratch every time user interests change and (4) the knowledge captured by those techniques will be based on some assumptions (for example, if a user spends more than 3 minutes in a page, the page is interesting to the user), than are not necessarily true in all cases, yielding to some noise in the user models obtained. An intermediate approach for user modeling is a hybrid user model in which part of the information is given by the user and part is obtained using machine learning. Typically in these hybrids models the user provides information regarding layout and colors while machine learning obtains information about information filtering/retrieval and personalized navigation.

The concept of Adaptive DL has been already sketched in some applications and implementations. Sections 3.1 through 3.4 give some examples of adaptive DL services for content personalization, interface personalization, IR/IF personalization and other related services.

3.1 Adaptive personalization of content

Adaptive personalization of content aims at developing systems that are able to automatically construct personal libraries according to user preferences. This process is intimately related with adaptive IF, by which a user incorporates information to his/her personal library. The main approaches for automatically constructing and refining a personal library are: (1) by defining a user as part of a stereotype and (2) by querying the DL using the interest of the user. The first approach can be used to create a personal library for a first-time user and/or to recommend new documents using personal data or domain expertise. An example of the second approach is Semeraro et al., (2000), which presents an agent designed to suggest improved ways to query the DL on the grounds of the documents stored in a personal catalogue.

3.2 Adaptive interface personalization

Adaptive interface personalization systems tailor the interface used by each user according to a set of user characteristics. These characteristics are basically: (1) the physical device used for accessing the DL and (2) the stereotype in which that particular user is included. Examples of stereotypes are the cognitive style or the level of tool expertise. An example of adaptive interface personalization using the first approach is Fernandez, Diaz, & Aedo (1999), which provides adaptation of the interface at a very basic level depending on the operative system and the hardware and software platform. Semeraro et al., (1999) and Semeraro et al., (2001) present an example of adaptive interface personalization using a stereotype, in this case the level of tool expertise. The system, once a user has started a session, obtains the level of expertise of the user and provides him/her with the most relevant interface (Costabile et al., 1999). The ideal adaptive interface service should combine all these information to personalize the interface.

3.3 Adaptive information filtering (IF) & information retrieval (IR)

Adaptive IF and IR systems personalize information according mainly to user’s interests and goals. In order to obtain user’s interests, adaptive systems use the information provided by the personal library of each user. An example of IR using this approach is McKeown, Elhadad, & Hatzivassiglou (2003), which presents a personalized IR system for medical literature that re-ranks the results of a search taking into account the patient record in order to help the doctor in the process of finding relevant literature to that particular patient. An example of IF using that same approach is Bollacker & Lawrence (1999) which presents a personalized IF system of scientific literature that constructs the user model by combining two methods: (1) constraint matching (keyword matching) and (2) feature relatedness. In the second approach the user indicates to the system papers that finds interesting and the system can use this information to suggest new papers based on some concept of distance. To some extent some tools for creating repositories of DL include some kind of adaptive IF/IR system. Cornelis (2003) presents a study to personalize IR for Greenstone (Greenstone, 2005). Fernandez, Diaz, & Aedo (1999), which presents an adaptive access to DL catalogues through Z39.50 servers, provides personalization for IF and IR by learning user interests from previous queries. In general user modeling for IF and IR is a very active research field that has focused mainly in news systems. Widyantoro (1999) and Montaner, Lopez, & de la Rosa (2003) present an extensive review of user modeling for news filtering systems. Ideally an adaptive IF/IR system will also use information regarding the context, the goal, the history and the domain expertise level to re-rank the documents obtained.

3.4 Other adaptive DL services

Automatic document classification using machine learning does not provide any direct adaptive personalized services but allows to implement systems that perform better adaptive searches and that automatically cluster documents. This approach indirectly allows to implement efficient and adaptive access to information. Some examples of automatic document classification are Rauber & Merkl (1999), Tsukada & Washio (2001) and Aihara & Tasaku (2001).

4. User modeling for adaptive DL services

The previous review has presented examples of adaptive DL services. In order to automatically create user models for adaptive DL services, three questions need to be answered: (1) what information should a DL user model contain, (2) which techniques can be used to automatically capture that information and (3) how the information captured can be used to create DL user models. These questions are answered in sections 4.1, 4.2 and 4.3 respectively.

4.1 Dimensions of a DL user model

One of the main problems that user modeling faces is the lack of any kind of standard of what a user model should contain. In general, the answer to the previous question is that the content of a user model is application dependent. Within the context of a personalized DL, we consider that there are nine potential dimensions that a user model should have:

- Personal Data. Personal data includes gender, age, language, culture, etc. Some of these factors affect the perception of the interface layout, and, in general, can be used to personalize any DL service.
- Cognitive Style (CS). Cognitive Style indicates the way in which a given user processes information. There are already studies that indicate how individuals from different cognitive styles interact differently with web-based services (Ford, & Chen, 2000) and in learning environments (Magoulas, Papanikolau, & Grigoriadou, 2003). It can be used to adapt the service to the way the user processes information.
- Device. Device captures the hardware used by the user to access the DL (PDA, laptop, Smartphone, etc.). The device affects the personalization in two ways: (1) size of the screen and (2) download speed. The system should consider the size of the screen when presenting the results to the user, while at the same time dealing with the bandwidth limitations of that device.
- Context. Context captures the physical environment from where the user is accessing the DL (from work, at home, from the university, from the Computer Science Department, etc.). This information can be used to infer the goals of that user.
- History. History captures user past interaction with the system and can be used to personalized any kind of service using the assumption that a user is going to behave in an immediate future in the same way it has behaved in the immediate past.
- Interests. Interests indicate, usually in the form of keywords, the more relevant topics for that user.
- Goal. Goal indicates, for that particular session, the reason for which that user is searching information. For example it is not the same to search information about China as a tourist searching for information about his/her destination or as a student writing a school report.
- System Experience. System experience indicates the knowledge of that particular user when interacting with the DL. This information can be used to adapt the interface to the user.
- Domain Expertise. Domain expertise indicates the knowledge of that particular user in the topics that interest that user. Note that a user can have different Domain Expertise levels for different topics. This information can be used to re-rank and recommend new documents.

To implement a given DL service, not all the presented dimensions are needed; again, the dimensions needed are service dependent. Table 1 presents which dimensions are relevant for each type of service: content personalization, interface personalization and IR/IF personalization. Table 1 does not imply that all the relevant dimensions of a given type of service should be captured for a specific service of that type, but that the final user model will contain a subset of those dimensions.

Table 1
Dimensions of a User Model and their relation with each DL Service.

	Content Personalization	**Interface Personalization**	**IR/IF Personalization**
Cognitive Style			
System Experience			
Domain Expertise			
History			
Device			
Context			
Personal Data			
Interests			
Goals			

4.2 Tools for automatic creation of UM

One of the modules presented in Figure 3 is the "User Modelling Generation" module, which, using machine learning techniques, automatically generates user models from the interaction data. The process of generating DL user models using machine learning is very similar to the process of extracting knowledge from data and can be seen as a standard process of extracting knowledge where DL user modelling is used as a wrapper for the entire process. It comprises the basic steps: (1) Data Collection, (2) Preprocessing, (3) Pattern Discovery and (4) Validation:

- Data Collection. In this stage user data is gathered. In the digital library context the data collected includes: data regarding the interaction between the user and the library, data regarding the environment of the user when interacting with the library, direct feedback given by the user, etc. This is the data stored in the Database of Interactions module of Fig. 3.
- Data Pre-processing. The information obtained in the previous stage cannot be directly processed. For DL user modelling, this involves mainly user identification and session reconstruction. This stage is aimed at obtaining, from the data available, the semantic content about the user interaction with the digital library. Also in this phase the data extracted should be adapted to the data structure used by machine learning techniques.
- Pattern Discovery. In this phase, machine learning techniques are applied to the data obtained in the previous stage in order to capture user behaviour. The output of this stage is a set of structural descriptions of what has been learned about user behaviour and user interests when interacting with the DL. These descriptions constitute the base of a user model. Different techniques will capture different user properties and will express it in different ways.
- Validation and Interpretation. In this phase the structures obtained in the pattern discovery stage are analyzed and interpreted. The patterns discovered can be interpreted and validated, using domain knowledge and visualization tools, in order to test the importance and usability of the knowledge obtained.

These steps take part in the modules presented in Fig. 3. The first step, Data Collection, takes place in the Database of Interactions module, and the other three steps, data pre-processing, pattern recognition and validation, take place in the User Modelling Generation module, which produces the data base of user models used by the personalization engine. Fig. 4 presents these relations.

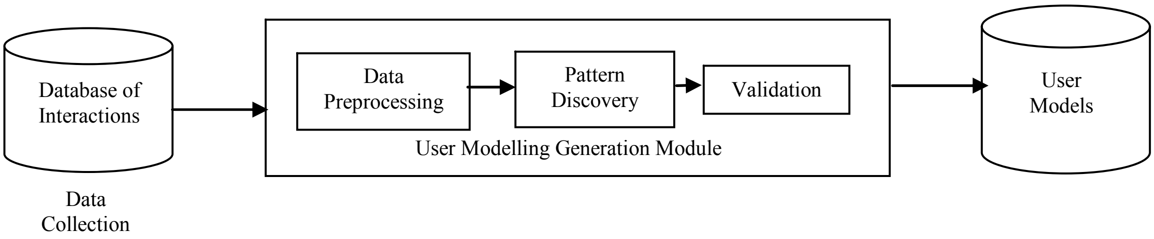

Fig. 4. Steps for Automatic Generation of User Models.

4.2.1 Machine learning for user modelling

The key phase for the automatic creation of user models is Pattern Discovery. Pattern Discovery is based on the idea that, from the interaction between a user and the DL, the set of preferences and interests of that user can be discovered. Machine learning techniques are ideal for that process because they are designed to capture patterns and to represent what have been learned from the input data with a structural representation.

Machine learning comprises a wide variety of techniques and is a very active research field. The main distinction in machine learning research is between supervised and unsupervised learning. Supervised learning requires the training data to be preclassified. This means that each training item is assigned a unique label, signifying the class to which the item belongs. Given these data, the learning algorithm builds a characteristic description for each class, covering the examples of this class. The important feature of this approach is that the class descriptions are built conditional to the preclassification of the examples in the training set. In contrast, unsupervised learning methods do not require preclassification of the training examples. These methods form clusters of examples, which share common characteristics. The main difference to supervised learning is that categories are not known in advance, but constructed by the learner itself. When the cohesion of a cluster is high, i.e., the examples in it are similar, it defines a new class.

The main supervised learning techniques used for modelling user behaviour are: k-Nearest Neighbour, Decision Tress/Classification Rules, Neural Networks and Support Vector Machines. Decision tree learning (Mitchell, 1997; Winston, 1992) is a method for approximating discrete-valued functions with disjunctive expressions. The most common decision tree algorithm is C4.5 (Witten & Frank, 1999). Classification rules are an alternative representation of the knowledge obtained from classification trees based on constructing a profile of items belonging to a particular group according to their common attributes. k-Nearest Neighbor (k-NN) is a predictive technique suitable for classification models (Friedman, Baskett, & Shustek, 1975). Unlike other learning techniques in which the training data is processed to create the model, in k-NN the training data represents the model. An Artificial Neural Network (ANN) (Haykin, 1999; Fausett, 1994) is an information processing paradigm that is inspired by the way biological nervous systems process information. SVM (Cristianini, & Shawe-Taylor, 2000) is a classifier derived from Statistical Learning Theory.

The main unsupervised learning techniques used for user modeling are clustering (which includes k-means clustering (Kaski, 1997), Self-Organizing Maps (SOM) (Kohonen, 1997), Hierarchical Clustering and fuzzy clustering (Bezdek, 1981)) and association rules. The task of clustering (Jain & Dubes, 1998) 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) the similarity between examples of the same concepts is maximized, and (2) the similarity between examples of different concepts is minimized. In a cluster

Table 2
General Characteristics of the Revised Techniques

	Off - Complexity	Dynamic Modeling	Labeled / Unlabeled	Readability
K-means Clustering	$O(k\ m\ n\ i)$ (Hartigan, 1975) n number of instances to cluster m number of attributes k number of clusters i number of iterations, with $i=O(n)$ (Davidson, & Satyanarayana, 2003)	No	Unlabeled	No
SOM	$O(n)$, with n the number of feature vectors (Ramsey, Zhen, & Zhu,1999) $O(M^2)$ where M is the number of map units: each learning step requires $O(M)$ and to achieve a sufficient statistical accuracy the number of iterations should be at least some multiple of M. (Kaski, 1997)	No	Unlabeled	Yes
Fuzzy Clustering	$O(n^2)$ with n the number of objects For some optimized algorithms $O(nlogn)$ (Krishnapuram et al., 2001)	No	Unlabeled	No
Association Rules	NP-Complete Exponential with the number of items (Angiulli, Ianni, & Palopoli, 1998)	No	N/A	Yes
Decision Trees	For single attribute, multi-way splits on A discrete variables and data size of N: $O(A^2N)$ For continuous attributes: $O(A^2N^3)$ (Martin & Hirschberg, 1995) Prunning: $O(N\ \ h^*)$ (Martin & Hirschberg, 1995)	Yes	Labeled	Yes
Classification Rules	Same as Decision Trees + Rule generation	Yes	Labeled	Yes
k-NN	Linear with the number of samples Empirical sample complexity is exponential in the number of irrelevant features (Langley & Iba, 1993)	Yes	Labeled	No
Neural Networks	NP-Complete for a generic 3 layer NN Polynomial for some simple two layer networks (Blum & Rivest, 1992)	Yes	Both	No
SVM	Complexity of Solving a Quadratic Optimization problem (QP) at each iteration: $O(N^3)$ with N total number of training points. In general it is highly dependent of the SVM implementation used.	No	Labeled	No

algorithm the key element is how to obtain the similarity between two items of the training set. Association rule discovery (Agrawal, Imielinski, & Swami, 1993) aims at discovering all frequent patterns among transactions and was based on detecting frequent items in a market basket.

Table 2 summarizes the characteristics of the techniques presented along four dimensions. The first three dimensions capture some of the main problems that machine learning for user modeling faces according to Webb, Pazzani, & Billsus (2001): Computational Complexity for off-line processing (training time); Dynamic Modeling, which indicates the suitability of the technique to change a user model on-the-fly; and Labeled/Unlabeled, which reflects the need of labeled data. An extra dimension has been added to characterize each technique: the "Readability" of the results, i.e. if the technique produces a human-readable output of the knowledge captured for a non-technical user.

4.3 Construction of user models for adaptive DL services using machine learning

The straight forward solution for user modeling is a user-driven or adaptable approach, in which the user directly gives all the information. In our context, the user could directly state his/her cognitive style, tool expertise, domain expertise, device, context, personal data, interests, and goals. This approach has a lot of problems, as previously stated, and in general, an adaptive approach is much better. Table 3 presents for each DL user model dimension how it can be obtained using an adaptive approach. The first column (Modeling) indicates how to model or to learn to classify a given user in the different groups or stereotypes of that dimension, and the second column (Operation) indicates how the DL runs the model obtained. Note that Personal Data dimension can be only obtained asking directly this information to the user. The following subsections detail for each dimension of the DL user model, the data needed, the machine learning techniques that can be useful and some implementation examples.

Table 3.
Adaptive implementation of each DL user model dimension.

	Modeling	Operation
Cognitive Style (CS)	(1) Group the interactions done by the users of each CS, (2) Construct of a classification system to identify each style.	When a new user enters the system: (1) Track the discriminative characteristics and (2) use them to classify the user.
System Experience (SE)	(1) Group the interactions done by the users of each SE, (2) Construct a classification system to identify each group of each individual.	When a new user enters the system: (1) Track the discriminative characteristics and (2) use them to classify the user.
Domain Expertise (DE)	The DL models each document indicating not only the content but also the level of difficulty of the document (metadata).	The DL tracks the documents accessed by the user and assigns a level of DE combining the level of the documents accessed.
History	Apply data mining and/or statistical techniques to capture relevant associations to obtain a model of behavior.	The model of behavior is applied to obtain a prediction (of a link requested, of a button pressed, etc.)
Device	Set of categories of Devices defined by the DL.	The DL identifies the device when the user starts a session (or the device identifies itself to the DL).
Context	Set of categories of Context defined by the DL.	(1) Identification of the position using GPS or the localization of the computer within the network. (2) Assign a context to that particular position.
Interests	Obtain from the personal library of each user the set of key words that represents his/her interests (use the metadata of the documents or document modeling techniques).	The resulting key words describe user's interest. Use that description to implement adaptive services. Run the algorithm regularly to track user changes.
Goals	Construct a goal-decision model The set of goals will be defined by the DL according to: (1) content of the DL, and (2) context from which the DL is being used.	(1) Obtain the Context of the DL user, (2) Retrieve user Personal Data and Interests, (3) run the goal-decision model, (4) Use the information of the goal to implement adaptive services.

4.3.1 Modeling cognitive style and system experience

The problem of identifying both the cognitive style and the system experience of a DL user is basically a classification problem in which a user, taking into account his/her interaction with the system, is assigned to a specific group. The data needed to construct the classification models to identify the cognitive style and the system experience is contained in the interaction logs stored in the server. The problem can be solved using supervised techniques like classification trees, classification rules, Support Vector Machines or neural networks. The labels needed for these classification techniques can be obtained using expert domain that classifies the set of interactions/user characteristics in each cognitive style or system experience level. Semeraro et al., (1999) is an example of this approach that implements an adaptive DL interface for each level of system experience using decision trees. Zhang (2003) uses classification rules to classify into different stereotypes the set of users of an news information retrieval system.

4.3.2 Modeling domain expertise and history

The modeling of the Domain Expertise dimension is intimately related with how each document of the DL is modeled. In this context, the model of a document will contain the document itself and metadata indicating the author, date, category, etc. In order to capture the Domain Expertise of a particular user, the metadata model should also contain and indication of the level of difficulty of that document. Some standards for semantic web based on ARIADNE (Ariadne, 2005) and Dublin Core (Dublin, 2005) already contain fields that indicate the level of difficulty. Using this information, the level of expertise of a user in a given topic, would be given by a combination of the difficulty level of the documents of that topic accessed by that user and/or stored in his/her personal library.

The History dimension of the model can be solved using association rules. Nanopoulos, Katsaros, & Manopolulos (2001) models web user history using association rules and uses it to predict the next user request. Other possible approach would be to use Markov models (Rabiner, 1986), for example (Sarukkai, 2000) uses Markov chains to capture user historic behavior in a web site and implement a link prediction service. The data needed to construct this dimension is contained in the interaction logs stored in the server.

4.3.3 Modeling user interest

To model the Interest dimension, the key element is also how a document is modeled. In this case, it is needed some kind of indication about the topic of a particular document, which is usually expressed in the form of keywords. In order to obtain these keywords, there are two possibilities: (1) the metadata already has a field that contains them or (2) the metadata does not contain keywords, they are obtained using a variety of document modeling techniques like TF-IDF (Term Frequency-Inverse Document Frequency). The combination of the keywords obtained from the documents of the personal library constructed by a user will indicate his/her interests.

In order to implement personalized IF/IR systems using the interests of a particular user it will be necessary to define a similarity measure between a user interest profile and the content of a document. For that, there are a variety of algorithms to indicate similarity like k-nearest neighbor, clustering or neural networks. Paliouras et al.,(1999) use clustering to recommend interesting news to a given user in a personalized news system. Billsus & Pazzani (1999) use k-NN to model the short-term interest of a user for a personalized news system. Sheperd, Watters, & Marath (2002) use neural networks to construct and adaptive news filtering system according to user interests.

4.3.4 Modeling user goals

Regarding the construction of a model to identify the goal of a user when interacting with a DL, the mechanism consists basically on a classification system that has a set of predefined categories (goals). In order to define these set of goals, some elements to consider are: (1) the content and organization of the DL (obviously a DL that contain only scientific documents will not be useful when searching information for holiday destinations), and (2) the context (it is not the same to search the term Java from the Computer Science Department or from the History and Geography department). In order to train the classification system, the data needed is given by the interaction logs of users searching information in the DL and their history and interests. Expert knowledge can be used to classify each set of interactions into the predefined goal categories. The next step is the use of that knowledge as training data to construct a classification system which will identify the elements that characterize each goal. Ruvini (2003) presents an example of this approach that constructs a system that infers the goal of a search using Support Vector Machines. Other valid approaches would be classification trees, decision rules or neural networks. Other possible solution for modeling goals that has obtained very good results is Bayesian networks. Horvitz, Breese, & Heckerman (1998) present the construction of a goal prediction system using Bayesian networks that infers the objectives of a user within a software environment.

5. Conclusions

This paper has presented a review of personalization in DL from which it can be concluded that the technology is still in a premature phase. The largest part of implementations are done using a user-guided solution and at a very basic level. We think that the next step of DL services should be oriented towards the implementation of adaptive DLs. This next level of DL services will be based on machine learning techniques that automate the process of constructing each one of the dimensions of a DL user model. Up to now the solutions that have used this approach are very limited.

The review also demonstrates that one of the main problems that personalized DL faces is the lack of any kind of standardization for the design of DL user models. In order to improve this situation this paper proposed a set of dimensions to create DL user models and has presented how to automatically capture them.

Acknowledgments

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

References

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining Association Rules between Sets of Items in Large Databases. *Proceedings of the 1993 ACM SIGMOD Conference*, 207-216.

Aihara, K., & Takasu, A. (2001). Category Based Customization Approach for Information Retrieval. *Proceedings of the 8th International Conf. on User Modeling, LNAI 2109*, 207-209.

Angiulli, F., Ianni, G., & Palopoli, L. (1998). On the complexity of mining association rules. *Data Mining and Knowledge Discovery*, 2(3), 263-281.

ARIADNE. (2004). ARIADNE Strategy White Paper. URL http://www.ariadne-eu.org.

Belkin, N.J., & Croft, W.B. (1992). Information Filtering and Information Retrieval: Two Sides of the Same Coin?. *Communications of the ACM*, 35(12), 29-38.

Bezdek, J.C. (1981*). Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York.

Billsus, D., & Pazzani, N. (1999). A hybrid User Model for News Story Classification. *Proceedings of the 7th International Conference on User Modeling*, 99-108.

Blum, A.L., & Rivest, R.L. (1992). Training a 3-Node Neural Network is NP-Complete. *Neural Networks*, 5(1), 117-127.

Bollacker, K.D., & Lawrence, S. (1999). A System for Automatic Personalized Tracking of Scientific Literature. *Proceedings of the 4th ACM Conference on Digital Libraries*, 105-113.

Brusilovsky, P., & Schwarz, E. (1997). User as Student: Towards an Adaptive Interface for Advanced Web-Based Applications. *A. Jamesson, C. Paris and C. Tasso (Eds.), User Modeling: Proceedings of the Sixth International Conference, UM97*, 177-188.

Callan, J., Smeaton, A., Beaulieu, M., Borlund, P., Brusilovsky, P., Chalmers, M., Lynch, C., Riedl, J., Smyth, B., Straccia, U., & Toms E. (2003). Personalization and Recommender Systems in Digital Libraries, Joint NSF-EU DELOS Working Group Report , URL http://www.ercim.org/publication/ws-proceedings/Delos-NSF/Personalisation.pdf.

Candela, L., & Straccia, U.(2003). The Personalized, Collaborative Digital Library Environment CYCLADES and its Collections Management. *Distributed Multimedia Information Retrieval - SIGIR 2003 Worksho*
Information Retrieval, LNCS 2924, 156-172.

Cohen, S., Fereira, J., Horne, A., Kibbee, B., Mistlebauer, H., & Smith, A. (2000). MyLibrary Personalized Electronic Services in the Cornell University Library. *D-Lib Magazine*, 6(4), URL http://www.dlib.org/dlib/april00/mistlebauer/04mistlebauer.html.

Cornelis, B. (2003). *Personalizing Search in Digital Libraries*. M.SC. Thesis, University of Maastricht.

Costabile, M.F., Esposito, F., Semeraro, G., & Fanizzi, N. (1999). An Adaptive Visual Environment for Digital Libraries. *International Journal of Digital Libraries*, 2(2-3), 124-143.

Cristianini, N., & Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines*. Cambridge University Press.

Davidson, I., & Satyanarayana, A. (2003). Speeding up k-means Clustering by Bootstrap Averaging. *IEEE Data Mining Workshop on Clustering Large Data Sets, Third IEEE International Conference on Data Mining*, URL http://www.cs.albany.edu/~ashwin/speeding-up-k-means.pdf.

Di Giacomo, M., Mahoney, D., Bollen, J., Monroy-Hernandez, A., & Ruiz-Meraz, C.M. (2001). MyLibrary, A Personalized Service for Digital Library Environments. *Proceedings of the 2nd DELOS Workshop on Personalization and Recommender Systems in Digital Libraries*, URL http://www.ercim.org/publication/ws-proceedings/DelNoe02/Giacomo.pdf.

Dubli. (2005). Dublin Core Metadata Initiative. URL http://dublincore.org.

Esposito, F., Fanizzi, N., Ferilli, S., & Semeraro, G. (1999). Supporting Document Acquisition and Organization in a Digital Library Service trough MLTechniques. *Proceedings of the ACAI-99 Works*
Intelligent Information Access, 15-21.

Fausett L. (1994). *Fundamentals of Neural Networks*. Prentice-Hall.

Fernandez, C., Diaz, P., & Aedo, I. (1999). WAY: A User Adapted Access to Information. *Proceedings of the Fifth International Conference on Information Systems, Analysis and Synthesis, ISAS99*, 37-42.

Fink, J., Kobsa, A., & Nill, A. (1997). Adaptable and Adaptive Information Access for All Users, Including the Disabled and the Elderly. A. *Jamesson, C. Paris and C. Tasso (Eds.), User Modeling: Proceedings of the Sixth International Conference, UM97*, 171-173.

Ford, N., & Chen, S. (2000). Individual Differences, Hypermedia Navigation and Learning: An Empirical Study. *Journal of Education Multimedia and Hypermedia*, 9 (4), 281-311.

French, C.J., & Viles, L.C. (1999). Personalized Information Environments. *D-Lib Magazine*, 5(6), URL http://www.dlib.org/dlib/june99/french/06french.html.

Friedman, J.H., Baskett, F., & Shustek, L.J. (1975). An algorithm for finding nearest neighbors. *IEEE Transactions on Computers*, 24, 1000-1006.

Greenston. (2005). Greenstone Project Home Page. URL http://www.greenstone.org.

Hartigan, J. (1975). *Clustering Algorithms*. Wiley Publishing.

Hartigan, J. A. (1983). *Bayes Theory*. New York Springer-Verlag.

Haykin S. (1999). *Neural Networks , 2nd Edition*. Prentice Hall.

Hicks, D.L., & Tochtermann, K. (2001). Towards Support for Personalization in Distributed Digital Library Settings. *Proceedings of the 2nd DELOS Network of Excellence Workshop on Personalisation and Recommender Systems in Digital Libraries*, 56-60.

Horvitz, E., Breese, J., & Heckerman, D. (1998). The Lumière project: bayesian user modeling for inferring the goals and needs of software users. *Proceedings of the 14th Conference on Uncertainty in AI*, 256-265.

Jain, A., & Dubes, R.C. (1988). *Algorithms for Clustering Data*. Prentice Hall.

Jayawardana, C., Hewagamage, K.P., & Hirakawa, M. (2001). Personalization tools for Active Learning in Digital Libraries. *Journal of Academic Media Librarianship*, 8 (1), http://wings.buffalo.edu/publications/mcjrnl/v8n1/active.pdf.

Jensen, F.V. (2001). *Bayesian Networks and Decision Graphs*. Springer.

Kaski, S. (1997). *Data Exploration Using Self Organizing Maps*. Acta Polytechnica Scandinavica, Mathematics, Computing and Management in Engineering Series No. 82.

Kobsa, A. (2001). Generic User Modeling Systems. *User Modeling and User-Adapted Int*, 11, 49-63.

Kohonen, T. (1997). *Self-Organizing Maps*. New York Springer-Verlag.

Krishnapuram, R., Joshi, A., Nasraoui, O., & Yi, L. (2001). Low-Complexity Fuzzy Relational Clustering Algorithms for Web Mining. *IEEE Transactions on Fuzzy Systems*, 9(4), 595-608.

Langley, P., & Iba, W. (1993). Average-case analysis of a nearest neighbor algorithm. *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, 889-894.

MacQueen, J.B. (1967). Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, 281-297.

Magoulas, G., Papanikolau, K., & Grigoriadou, M. (2003). Adaptive web-based learning: accomodating individual differences through system's adaptation. *British Journal of Educational Technology*, 34 (4), 1-19.

Manber, U., Patel, A., & Robinson, J. (2000). Experience with Personalization on Yahoo!. *Communications of the ACM*, 43 (8), 35-39.

Martin, J.K., & Hirschberg, D.S. (1995). *The time complexity of Decision Tree Induction*. Department of Information and Computer Science, University of California at Irvine, Technical Report 95-27 (ICS/TR-95-27).

McKeown, K.R., Elhadad, N., Hatzivassiglou, V. (2003). Leveraging a Common Representation for Personalized Search and Summarization in a Medical Digital Library. *Proceedings of the 2003 Joint Conference on Digital Libraries, JCDL03*, 159-173.

Mitchell, T. (1997). *Decision Tree Learning*. In Machine Learning, McGraw-Hill, Inc., 52-78.

Montaner, M., Lopez, B., & de la Rosa, J.L. (2003). A Taxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review*, 19(4), 285-330.

Nanopoulos, A., Katsaros, D., & Manolopoulos, Y. (2001). Effective Prediction of Web-user Accesses: A Data Mining Approach. *Proceeding of the WEBKDD 2001 Workshop, LNAI 2356*, 48-68.

Paliouras, G., Karkaletsis, V., Papathedorou, C., & Spyropoulos, C. (1999). Exploiting Learning Techniques for the Acquisition of User Stereotypes and Communities. *Proceedings of the Seventh International Conference on User Modelling (UM '99)*, 169-178.

Rabiner, J. (1986). An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, 3(1), 4-16.

Ramsey, M., Chen, H. and Zhu, B. (1999). A Collection of Visual Thesauri for Browsing Large Collections of Geographic Images. *Journal of the American Society for Information Science & Technology*, 50(9), 826-834.

Rauber, A., & Merkl, D. (1999). SOMLib: A Digital Library System Based on Neural Networks. *Proceedings of the 4th ACM Conference on Digital Libraries (DL'99)*, 240-241.

Riecken, D. (2000). Personalized Views of Personalization. *Communications of the ACM*, 43 (8),27-28.

Ruvini, J. (2003). Adapting to the User's Internet Search Strategy. *Proceedings of the 8th international conference on intelligent user interfaces*, 284-286.

Sarukkai, R.R. (2000). Link Prediction and Path Analysis using Markov Chains. *Computer Networks*, 33(1-6),377-386.

Scirus (2004). Scirus White Paper. *How Scirus Work*. URL http://www.scirus.com/press/pdf/WhitePaper_Scirus.pdf.

Semeraro, G., Abbatista, F., Fanizzi, N., & Ferilli, S. (2000). Intelligent Information Retrieval in Digital Library Service. *Proceedings of the First DELOS Network of Excellence Workshop on Information Seeking, Searching and Querying in Digital Libraries*, 135-140.

Semeraro, G., Costabile, M.F., Esposito, F., Fanzini, N., & Ferilli, S. (1999). Machine Learning Techniques for Adaptive User Interfaces in a Corporate Digital Library Service. *Proceedings of the ACAI 99 Workshop on Machine Learning in User Modeling*, 21-29.

Semeraro, G., Ferilli, S., Fanizzi, N., & Abbattista, F. (2001).Learning Interaction Models in a Digital Library Service. *Proceedings of the 8th International Conferece on User Modelling, LNAI 2109*, 44-53.

Sheperd, A., Watters, C., & Marath, A.T. (2002). Adaptive User Modeling for Filtering Electronic News. *Proceedings of the 35th Annual Hawaii International Conference on Sys*, Vol. 4. 102-111

Theng, Y.L., Duncker, E., & Mohd-Nasir, N. (1999). Design Guidelines and User-Centred Digital Libraries. *Proceedings of the Third European Conference on Research and Advanced Technology for Digital Libraries, LNAI 1696*, 167-183.

Tsukada, M., Washio, T., (2001). Automatic Web-Page Classification by Using Machine Learning Methods. *Web Intelligence: Research and Development, LNAI 2198*, 303-313.

Webb, G.I., Pazzani, M.J., & Billsus, D. (2001). Machine Learning for User Modeling. *User Modeling an U Interaction*, 11, 19-29.

Widyantoro D.H. (1999). *Dynamic Modeling and Learning User Profile in Personalized News Agent*. Ms.C. Thesis, Texas A&M University.

Winston, P. (1992). *Learning by Building Identification Trees. Artificial Intelligence*. Addison-Wesley Publishing Company, 423-442

Winter, K. (1999). MyLibrary can help your library. *American Libraries*, 30(7), 65-57.

Witten, I.H., & Frank E. (1999). *Data Mining. Practical Machine Learning Tools and Techniques with JAVA Implementations*. Morgan Kaufman Publishers.

Zhang, X. (2003). Discriminant Analysis as a Machine Learning Method for Revision of User Stereotypes of Information Retrieval Systems. *MLIRUM'03: Second Workshop on Machine Learning, Information Retrieval and User Modeling, 9th International Conference on User Modeling*, 1-11.

Zukerman, I., Albrecht, D.W., & Nicholson, A.E. (1999). Predicting Users Request on the WWW. *Proceedings of the 7th International Conference on User Modeling, UM99*, 275-284.