User Modelling for Digital Libraries: A Data Mining Approach

A Thesis submitted for the degree of Doctor of Philosophy by

Enrique Frias-Martinez

School of Information Systems, Computing and Mathematics

Brunel University

November 2006

Abstract

Digital libraries provide information services for users who have diverse needs. Personalised digital libraries, constructed using a user-guided approach in which users need to state their preferences explicitly, have been proposed as a way to meet the needs of different users. The problems of using this approach are that users may not be aware of their preferences and that human factors are ignored. To address these problems, this thesis investigates an automatic approach that captures user preferences with data mining techniques and identifies relevant human factors for personalisation.

More specifically, this thesis aims to study to which extent data mining can identify user preferences and to clarify the role of human factors in determining user behaviour and user perception. In addition to studying the modelling capabilities of different individual data mining techniques, including K-means, Hierarchical Clustering and Fuzzy Clustering, Robust Clustering is applied for grouping user behaviour and user perception due to its capabilities of handling the inherent fuzziness of human data. The created clusters are used to find relationships with a number of human factors in order to identify their role in determining user behaviour and user perception.

The results show that there are relationships between cognitive styles and user behaviour. In addition, there are links between the levels of experience and user perception. However, novices do not show a homogenous perception so the levels of experience may not be suitable for personalisation. In other words, cognitive style has been considered as the main relevant human factor for personalisation, in which an adaptive interface is developed and compared with an interface that includes adaptability with the user-guided approach. The purpose of this comparison is to demonstrate to which extent data mining can capture user preferences. The results show that data mining is able to identify user preferences to such an extent that the inclusion of adaptability for personalisation does not increase user satisfaction.

In summary, this thesis makes contributions to three communities: digital libraries, data mining and personalisation. For the digital library community, this thesis has showed that cognitive style is the main relevant human factor that influences user behaviour in a digital library. Based on the results, an adaptive interface is developed to accommodate the needs of each cognitive style. For the data mining community, this thesis has indicated that data mining can be successfully used to identify user preferences. In particular, Robust Clustering is an effective technique to capture and model user behaviour and user perception. For the personalisation community, this thesis has demonstrated that personalisation has positive influences on the increase of user satisifaction.

Declaration

The following publications have resulted from work conducted as part of this investigation.

Journal Papers

1. E. Frias-Martinez, S. Chen, and X Liu, "Automatic Cognitive Style Identification of Digital Library Users for Personalisation". *J. of the American Society for Information Science and Technology* (forthcoming), 2006, Impact Factor (2005 JCR): 1.583
2. E. Frias-Martinez, S. Chen, and X. Liu "A Survey of Data Mining Approaches to User Modelling for Adaptive Hypermedia", in *IEEE Transactions in System, Man and Cybernetics – Part C* 36 (6), 2006, 734-749, Impact Factor (2005 JCR): 0.706.
3. E. Frias-Martinez, G. Magoulas, S. Chen, and R. Macredie, "Automated User Modelling for Personalised Digital Libraries". *International Journal of Information Management* 26, pp. 234-248, 2006, Impact Factor (2005 JCR): 0.479.
4. E. Frias-Martinez, and S. Chen, "An Empirical Study of Individual Differences in Digital Library Interfaces". *WSEAS Tran. on Computers, 10(4),* 1449-1462, 2005.
5. E. Frias-Martinez, G. Magoulas, S. Chen, and R. Macredie, "Modelling Human Behaviour in User-Adaptive Systems: Recent Advances Using Soft Computing Techniques", in *Expert Systems with Applications*, *vol. 29(2)*, 2005, 320-329, Impact Factor (2005 JCR): 1.236 – 8th in the Top 25 articles within the journal

Conference Papers

6. E. Frias-Martinez, and S. Chen, "Evaluation of User Satisfaction with Digital Library Interfaces", in *1st WSEAS International Symposium on Digital Libraries*, Corfu, Greece, August 2005
7. E. Frias-Martinez, G.D. Magoulas, S. Chen, and R. Macredie, "Recent Soft Computing Approaches to User Modelling in Adaptive Hypermedia", In Paul De Bra, Wolfgang Nejdl (eds), *Adaptive Hypermedia and adaptive web-based systems, Proceedings of 3rd Int. Conf. Adaptive Hypermedia, AH 2004*, Eindhoven, The Netherlands, Aug. 2004, LNCS, vol. 3137, Springer, 104-113. Impact Factor (2004 JCR): 0.513

Papers under Revision:

8. E. Frias-Martinez, S. Chen, and X. Liu, R. Macredie, "Behaviour and Perception of Digital Library Users: A Robust Clustering Approach", in *User Modelling and User Adapted Interaction* (submitted December 2005), Impact Factor (2005 JCR): 1.318

Acknowledgements

I am grateful to the Arts and Humanities Research Council (AHRC) for funding the project "Cognitive Personalised Interfaces for Web-based Library Catalogues" in which I have worked for the last three years. I am also extremely grateful to Dr. Sherry Y. Chen and Prof. Xiaohui Liu for their help and guidance throughout this project. Their ability to handle research projects, have new ideas and create an environment in which each individual is appreciated has helped me to increase my quality as a researcher. Thanks also to the DISC and the IDA group for providing an ideal environment for completing the goals of the research.

I am also grateful to my family that always support me even when I do not want their support. I hope some day I will be able to thank them for everything they have done. To Alicia Anne Colligan, what I can say, my life is an adventure because of you. I am sure that our next (and sunny) chapter will be even better than the previous ones.

My memories of London will always be tied to physical places that have a special meaning: Hammersmith Bridge, Frith Street, Soho Square, The Embankment, the French creperie in South Kensington, our ice creams in Leicester Square and our walks around Convent Garden. I will always remember these three years I have spent in London and hope to take with me some elements of Londoners that I deeply admire: entrepreneurship, courage and respect for one another. I can proudly say that, for some time, I was too a Londoner.

Table of Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

ANFIS	Adaptive-Network-based Fuzzy Inference Systems
BLC	Brunel Library Catalogue
CSA	Cognitive Style Analysis
DL	Digital Library
FC	Fuzzy Clustering
FCM	Fuzzy C-Means Algorithm
FD	Field Dependent
FD/FI	Field Dependent / Field Independent
FI	Field Independent
FL	Fuzzy Logic
HCI	Human-Computer Interaction
I	Imager
IF	Information Filtering
IR	Information Retrieval
MLP	Multi-Layer Perceptron
NFS	Neuro-Fuzzy System
NN	Neural Networks
PIE	Personalised Information Environment
RC	Robust Clustering
RMSE	Root Mean Square
RMSE	Root Mean Square Error
SC	Soft Computing
UM	User Modelling
V	Verbaliser
V/I	Verbaliser/Imager
WBLC	Web-based Library Catalogue

Chapter 1

Introduction

1.1 Thesis Context

This thesis presents a work, which relates to several disciplines, including digital libraries, personalisation, data mining and human-computer interaction. This section gives an introduction to these disciplines.

Digital Libraries (DLs), in general, can be defined as collections of information that have associated services delivered to user communities using a variety of technologies (Callan et al., 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. In a DL, the information is typically accessed using an interface presented by a computer that is connected to the Web via a network. From this perspective, DLs are just another example of a web-based application.

One of the major trends in web-based applications is personalisation. In general, personalisation is about building customer loyalty by developing a meaningful one-to-one relationship (Riecken, 2000). Personalisation can be defined as a technology that allows tailoring the content and presentation of a web-based application for each individual according to his/her preferences and characteristics (Perkowitz and Etzioni, 1999; 2000). The information needed for personalisation is stored in a user model. There are two main trends to create user models: (1) user guided, in which the user directly states his/her preferences and (2) automatic, in which the user models are created with data mining techniques.

Data Mining and Machine Learning techniques encompass techniques where a machine acquires/learns knowledge from its previous records (Witten and Frank, 1999; Hand et al., 2001). The rest of the thesis will use the term, data mining, to refer to both data mining and machine learning techniques. The output of a data mining technique is a structural description of what has been learned that can be used to explain original data and to make predictions. In this thesis, the output of data mining techniques will represent user models.

The field of Human-Computer Interaction (HCI) strives towards creating systems that are more user-friendly. In the context of HCI, human factors are defined as any individual differences that may make users have diverse experiences when they interact with web-based applications. Previous research has demonstrated that gender differences (Roy and Chi, 2003), levels of experience (Mitchell et al., 2005) and cognitive styles (Chen and Macredie, 2004) are significant human factors that influence users' interaction with web-based applications. The rest of the thesis will use the terms, human factors, individual factors and individual differences interchangeably.

The aim of this chapter is to introduce and define the areas under investigation. Firstly, section 1.2 gives a definition and justification of the problems. Subsequently, section 1.3 presents the thesis objectives, followed by section 1.4, which presents a description of the contributions of this thesis. Finally, Section 1.5 illustrates the structure of the thesis.

1.2 Problem Definition

Current Digital Libraries (DLs) are becoming more complex systems than traditional libraries because they provide mixed-mode, multimodal, and multimedia information. Moreover, they are used by users with diverse background, preferences and needs. In comparison with traditional libraries, DLs make information directly available to users via both intranets and the internet (Gonzalves and Fox, 2002). Without the mediation of librarians, it is necessary for DLs to bridge the terminological and cognitive gaps between the producers and the users of the information (Nordlie, 1999). In particular, previous studies indicate that unassisted online searching in DLs may make end-users meet more difficulties (Borgman, 1996). In addition, end-users have problems in choosing search terms to represent their needs and in judging the relevance of the documents (Large and Beheshti, 1997). Several factors contribute to these problems (Borgman, 1996; Large and Beheshti, 1997):

- Bibliographic description does not provide the relevance judgement based on users' personal preferences.

- There is a lack of appropriate navigation support for users with different needs.
- Search options and format presentations are not flexible enough to align with different users' tasks, behaviour, and experience.

To overcome these problems, there is a need to provide personalisation to meet each user's needs (Ramsden, 2002). The importance of personalisation has been demonstrated by previous research in several areas such as web-based learning (Magoulas et al., 2003) and electronic commerce (Ardissono and Goy, 2000). In the context of DLs, previous works (Hicks et al., 1999; Van House, 1995) indicate that personalisation can support user performance in complex tasks, for example collecting information from different types of resources. Nuernberg et al. (1995), claim that personalisation can facilitate effective information access.

Personalisation is recognised as an effective approach in DLs but existing applications mainly use a user-guided (or adaptable) approach in which users need to state their preferences explicitly (Hicks et al., 1999; Dushay, 2002). There are some problems associated with the user-guided approach. For example, users do not necessarily understand the concept of personalisation, and if they understand it, they are not aware of their preferences. The other problem is that human factors are ignored though empirical evidence indicates that human factors have significant effects on users' information seeking, including levels of experience (Chen and Ford, 1997), gender differences (Ford and Miller, 1996), and cognitive styles (Chen and Macredie, 2004). These problems can be solved by using an automatic (or adaptive) approach, which produces user models to describe users' preferences with data mining techniques. Nevertheless, it is unknown to which extent this automatic approach, implemented with data mining techniques, is able to capture user preferences. Thus, it is necessary to study to which extent data mining techniques are able to generate use models that reflect users' preferences.

1.3 Thesis Objectives

The aim of this thesis is to study to which extent data mining techniques are able to capture user preferences for personalisation. Along with this aim, the thesis will also aim to identify which human factors are relevant to determine user behaviour and user perception so that they can be considered for personalisation. These aims can be achieved by the following objectives:

- Identify the main patterns of user behaviour and user perception;

- Identify human factors that influence user behaviour and user perception;
- Design a personalised interface based on the identified user behaviour and user perception;
- Study to which extent the introduction of personalisation increases user satisfaction and facilitates information access;
- Study to which extent automatic user modelling approaches with data mining techniques are able to capture user preferences.

1.4 Contributions

This thesis presents an interdisciplinary study, which makes contributions to three communities, including the communities of digital libraries, data mining, and personalisation. These contributions are described below:

- ***Digital Library Community***

  This thesis compares the effects of different human factors on user behaviour in a DL. The results show that cognitive style is the main relevant human factor that influences user behaviour. Based on the results, a personalised DL interface is developed for each cognitive style. The goal of this personalisation is to accommodate the needs and preferences of different cognitive style groups.

- ***Data Mining Community***

  Although it is known that data mining can be used to identify users' preferences, it is unsure to which extent the information captured represents user preferences. The results presented in this thesis show that data mining techniques can effectively capture users' preferences. In particular, Robust Clustering is an useful technique to capture and model user behaviour and user perception.

- ***Personalisation Community***

  Although it is generally admitted that personalisation can increase user satisfaction and facilitate information access by tailoring the interface to each user's preferences, there is an absence of empirical evidence to identify whether personalisation can increase user satisfaction. This thesis presents an empirical study that shows personalisation has positive effects on user satisfaction.

1.5 Overview of Thesis

Following from this chapter, Chapter 2 presents the state of the art of data mining for user modelling. It describes the relationships between data mining and user modelling and presents the main steps that the process of user modelling involves. The state of the art focuses on supervised learning, unsupervised learning and soft computing techniques. In the end of the chapter, a set of guidelines is presented for the selection of data mining techniques for user modelling.

Chapter 3 presents the main concepts and the state of the art of DLs. It focuses on two main approaches for personalisation: user-guided and automatic, and highlights the benefits of using an automated approach for building personalised DLs. Subsequently, the chapter presents human factors that are relevant to build personalised DLs.

Chapter 4 presents the design of an experiment to identify users' interaction with Brunel Library Catalogue (BLC). Once the experiment is presented, a statistical analysis is used to identify the relationships between human factors and user behaviour and user perception within BLC. Although such results are useful to explain users' different behaviour and perception, they do not give enough justification as to decide which human factor is more relevant for personalisation.

Chapter 5 presents an analysis of user behaviour and user perception using various data mining techniques and identifies which human factors are responsible for user behaviour and user perception. The results show that cognitive style is responsible for determining user behaviour and the level of experience determines user perception. In addition, this issue is also identified by using Robust Clustering, a technique that combines different data mining techniques, to better deal with the fuzziness of human data.

Chapter 6, using the results of the previous two chapters, presents the design of a personalised interface for BLC based on the user behaviour of each cognitive style. The chapter also presents the other experiment designed to capture user satisfaction with the personalised BLC interface. The experiment is also designed to capture user satisfaction when users interact with an adaptable interface. The analysis of user behaviour and user perception of the personalised BLC interface shows an increase in user satisfaction and a decrease in the time and number of transactions needed to locate information. When the adaptive (or automatic) interface is compared with the adaptable (or user-guided) interface, user satisfaction does not really change. These results imply that data mining can contribute to capture users' preferences for personalisation.

Chapter 6 showed that personalisation based on cognitive styles is an effective approach to increase user satisfaction. In order to implement such a personalised interface, each user's cognitive style needs to be identified in advance. Therefore, it is necessary to develop a mechanism that can automatically identify users' cognitive styles. To this end, Chapter 7 presents a mechanism for automatically identifying users' cognitive styles with a variety of data mining techniques. The effectiveness of these data mining techniques for capturing users' cognitive styles is also compared in this chapter.

Chapter 8 presents the conclusions of this thesis: (1) that data mining can be used to capture user preferences for personalisation, (2) that cognitive style is relevant in determining user behaviour and (3) that level of experience is relevant in determining user perception. Directions for future research are also described in this chapter.

Chapter 2

Data Mining Approaches to User Modelling for Personalisation

2.1 Introduction

After giving an introduction of the thesis, this chapter is going to describe the basic concepts and the state of the art of data mining for personalisation. Personalisation can be defined as the technology that allows tailoring for each individual the content and presentation according to his/her preferences and characteristics (Perkowitz et al., 1999; Perkowitz et al., 2000). In general, personalisation is about building customer loyalty by establishing a meaningful one-to-one relationship; by understanding the needs of each individual and helping reaching a goal that efficiently and knowledgeably addresses each individual's need in a given context (Riecken, 2000). One of the first examples of a personalised environment is MyYahoo! (Manber et al., 2000).

The personalisation process of a hypermedia application is done by using a personalisation engine which adapts the contents of the hypermedia system according to the information contained in each user model. From this perspective, the key element of a personalised hypermedia application is the user model. The more information a user model has, the better the content and presentation will be personalised for each individual.

A user model can be created by using an automatic approach because users may exhibit specific patterns when accessing a hypermedia system. These patterns can then be used as the input of data mining techniques to automatically identify their preferences and produce user models as output. From this perspective, data mining makes it possible to create user models automatically. Data mining encompasses techniques where a computer acquires knowledge from previous experience, typically by discovering useful patterns in large data bases (Witten and Frank, 1999; Hand et al., 2001). The output of a data mining technique is a structural description of what has been learned that can be used to explain the original data and to make predictions.

More specifically, automatic user modelling is defined as the discovery of unobservable information about a user (such as preferences, behaviour, etc.) from observable information from that user, i.e. user interactions, (Zukerman et al., 1999), using data mining techniques. Such an automatic user modelling has been used in building adaptive hypermedia systems, in which users' behaviour can be unobtrusively observed using data mining techniques. The present chapter presents how data mining has been used to create adaptive hypermedia systems. The goals of this chapter are:

- To review and analyse the current literature on hypermedia systems, data mining and user modelling.
- To present the basic concepts covering hypermedia system, personalisation and automatic user modelling.
- To present a survey of the different data mining techniques available for modelling user behaviour for adaptive hypermedia systems.
- To give a set of guidelines for the selection of those data mining techniques.

The organisation of the chapter is as follows. The chapter starts by defining the concept of user model. Subsequently, the relationship between user modelling and adaptive hypermedia is highlighted. Then, the basic steps for the automatic creation of user models are explained. This section describes how to capture the information and represent the knowledge that a user model should contain. The next section emphasises on how data mining can help in the process of automatic creation of user models and which techniques have been used. For each technique, we present a theoretical background, its pros and cons and its applications in the field of user modelling. Furthermore, a set of guidelines is produced to provide guidance on how to create a user model according to the needs of the adaptive hypermedia systems. Finally, the conclusions section closes the chapter.

2.2 User Modelling and Adaptive Hypermedia

A user model should capture the behaviour (patterns, goals, interesting topics, etc.) that a user shows when interacting with hypermedia systems. Ideally, a generic user model should store all the characteristics of a user. Nevertheless, typically user models are designed to store exclusively the information needed by particular personalised services being implemented.

A user model can be defined as a set of information structures designed to represent one or more of the following elements (Kobsa, 2001): (1) representation of 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; (3) the classification of a user in one or more of these subgroups; (4) the recording of user behaviour; (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 groups.

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 are called adaptable (Fink et al., 1997), while the ones produced using an automatic approach are called adaptive (Fink et al., 1997; Brusilovsky and Schwarz, 1997). Although the former has the main advantage of allowing the user to directly state his/her preferences, it also has some inconveniences:

- The concept of personalisation is not necessarily understood by all the users of the system.
- 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 can not provide information to the system.
- Even if the user is aware of his/her interests, the amount of information that today hypermedia systems have make it unrealistic for a user to specify his/her preferences completely.

The latter can solve some of the problems of the former mainly because user models can be constructed without the direct intervention of the user by using data mining techniques. It also faces some inconveniences:

- At the beginning, the hypermedia system does not have any information about the user, which means that a generic personalisation should be used.

- User interests and preferences will change over time and the system has to be designed to capture those changes.
- Data mining techniques need to be scalable in order to be able to cope with the millions of users that a system can have.
- 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). However, they are not necessarily true in all cases, so some noise may exist in the user models.

Another approach is a hybrid user model in which part of the information is given by the user and the other part is obtained using data mining techniques. Typically in these hybrids models, the user provides information regarding layout and colours while data mining obtains information about information filtering/retrieval and navigation patterns. The rest of the section presents the process of constructing automatic user models for adaptive hypermedia systems.

As described in Section 2.1, the personalisation is done by using a personalisation engine according to the information given by each user model. As seen in Figure 2.1, the input of the personalisation engine is the set of behaviour models and a hypermedia database that contains the basic elements to construct the adaptive hypermedia systems. User models are not only constructed with the patterns detected with data mining techniques but can also contain knowledge introduced by designers.

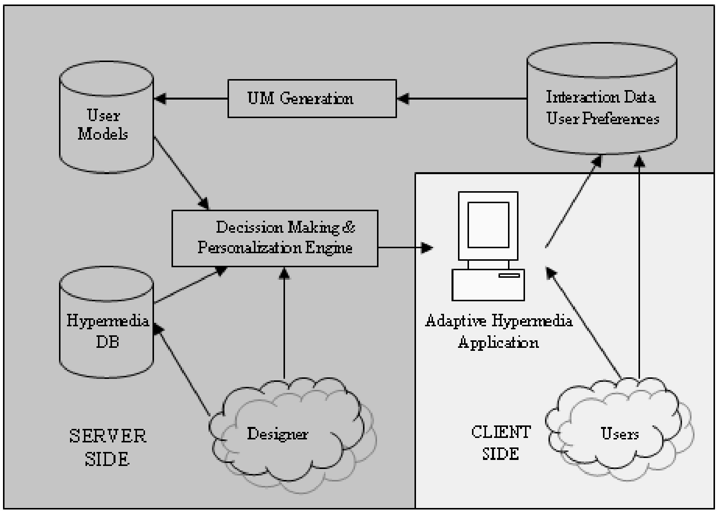


Figure 2.1: Generic Architecture of an Adaptive Hypermedia System

The personalisation engine retrieves the model of the present user and constructs a personalised interface using the elements of the user model and the multi-media data base. An adaptive hypermedia system, by its very nature, should respond in real time. To do so, the architecture of the system should provide a quick access to the multi-media databases.

Adaptive hypermedia systems use the knowledge given by user models to implement an adaptive task. Recommendation and classification are the two basic types of tasks:

- *Recommendation:* Recommendation is the capability of suggesting interesting elements to a user based on some information; for example from the items to be recommended or from the behaviour of other users. Recommendation is also known in the literature as collaborative filtering (Shardanand and Maes, 1995).
- *Classification:* Classification builds a model that maps or classifies data items into one of several predefined classes. Classification is done by using only data related to that particular item. This knowledge can be used to tailor the services of each user. In the literature, classification has also been presented as content-based filtering (Yan and Garcia-Molina, 1995).

The UM Generation module presented in Figure 2.1 generates user models from the interaction data between the users and the hypermedia system. The process of automatic generation of user models using data mining techniques is very similar to the standard process of extracting knowledge from data. Figure 2.2 presents the basic steps: (1) Data Collection, (2) Pre-processing, (3) Pattern Discovery and (4) Validation and Interpretation (Witten and Frank, 1999).

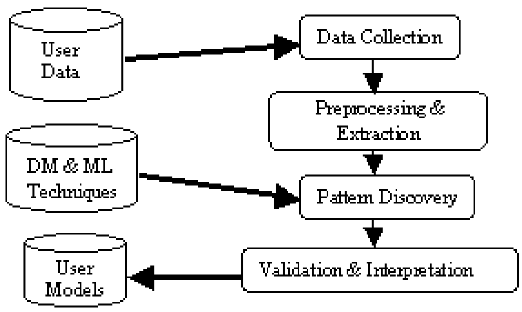


Figure 2.2: Steps for Automatic Generation of User Models

- *Data Collection.* In this stage, user data is gathered. For automatic user modelling, the data collected includes: data regarding the interaction between the user and the system, data regarding the environment of the user, direct feedback given by the user, etc.

- *Data Pre-processing/Information Extraction.* The information obtained in the previous stage cannot be directly processed. It needs to be cleaned from noise and inconsistencies in order to be used as the input of the next phase. For 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 system. Also, in this phase the data extracted should be adapted to the data structure used by standard pattern discovery algorithms used in the next step.
- *Pattern Discovery.* In this phase, data mining 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 have been learned about user behaviour and user interests. These descriptions constitute the base of a user model. Different techniques will capture different user properties and will express it in different ways. The knowledge needed to implement an adaptive service will determine which techniques to apply in this phase.
- *Validation and Interpretation.* In this phase, the structures obtained in the pattern discovery stage are analysed 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. In general, this process is done with the help of a user modelling designer.

2.3 Data Mining for User Modelling

As it has been presented in Figure 2.2, the phase of Pattern Discovery automatically finds out relevant information about the behaviour of a user. Data mining techniques are ideal for that process because they are designed to represent what has been learned from the input data with a structural representation. This representation stores the knowledge needed to implement the two types of tasks previously described.

Each data mining technique will capture different relationships among the data available and will express the results using different data structures. The key question is to find out which patterns need to be captured in order to implement an adaptive service. It is important, in order to choose a suitable method, to know what knowledge is captured by each technique and how that knowledge can be used to implement the two basic adaptive tasks. Furthermore, the choice of suitable methods largely depends on the type of training data available. Traditionally, the main distinction in 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 conditionally to the preclassification of the examples in the training set (Witten and Frank, 1999).

In contrast, unsupervised learning methods do not require preclassification of the training examples. These methods form clusters of items that share common characteristics. The main difference to supervised learning is that classes are not known in advance, but constructed by the data. When the cohesion of a cluster is high, a new class is defined (Witten and Frank, 1999).

Traditional data mining techniques have some limitations for modelling human behaviour, 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 User Modelling. SC is an approach to building computationally intelligent systems that differs from conventional (hard) computing in that it has tolerance for 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 cost solutions (Sinha et al., 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 et al., 1997).

The rest of the section presents how the data mining techniques that are going to be used in this thesis have been used for user modelling: which knowledge can be captured with each technique, examples of applications and its limits and strengths. The techniques presented are divided into three groups:

- Unsupervised Learning, which includes hierarchical clustering, non-hierarchical clustering, fuzzy clustering and robust clustering.
- Supervised Learning, which includes Decision trees and Neural Networks.
- Soft Computing, which includes Fuzzy Logic and Neuro-Fuzzy Systems.

An extensive study of how unsupervised learning, supervised learning and soft computing techniques have been used for user modelling, including other techniques such as association rules (Agrawal et al., 1993) self-organizing maps (Kohonen, 1997), K-nearest neighbour (Friedman, 1975), Support Vector Machines (Boser et al., 1992) and Genetic Algorithms

(Goldberg, 1989), can be found in Frias-Martinez et al. (2006) and Frias-Martinez et al. (2005).

Finally, there are other techniques not reviewed in this chapter, mainly predictive statistical techniques (Zukerman and Albrecht, 2001), that can be also used to create user models. For example, recommendation and classification tasks have also been implemented with Markov models (Anderson et al., 2001; Anderson et al., 2002; Deshpande and Karypis, 2001; Duchamp, 1999; Sarukkai, 2000), or with Bayesian networks (Witting, 2003; Conati et al., 1997).

2.3.1 Unsupervised Learning Approaches to User Modelling

Unsupervised learning techniques group two main families of algorithms: clustering and association rules. This section will focus on the clustering techniques used in this thesis. Clustering comprises a wide variety of different techniques based in the same concept. A collection of different clustering techniques and its variations can be found in Jain and Dubes (1999).

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) the similarity between examples of the same concepts is maximised, and (2) the similarity between examples of different concepts is minimised. In a cluster algorithm, the key element is how to obtain the similarity between two items of the training set.

Clustering techniques can be classified in hard (non-fuzzy) clustering and fuzzy clustering. In hard or non-fuzzy clustering, data is divided into crisp clusters, where each data point belongs to exactly one cluster. In fuzzy clustering, the data points can belong to more than one cluster, and associated with each of the instances are membership grades which indicate the degree to which they belong to the different clusters.

Hard clustering techniques may be grouped into two categories: non-hierarchical and hierarchical (Jain and Dubes, 1999). Non-hierarchical or partitional procedures end up with a particular number of clusters at a single step while hierarchical clustering procedures involve the construction of a hierarchy or tree-like structure, which is basically a nested sequence of partitions.

1) Basic Algorithms: Non-hierarchical Clustering Techniques

A typical example of a non-hierarchical clustering technique is k-means. The k-means clustering technique (MacQueen, 1967) is given as input the number of clusters *k*. The algorithm then picks k items, called seeds, from the training set in an arbitrary way. Then, in each iteration, each input item is assigned to the most similar seed, and the seed of each cluster is recalculated to be the centroid of all items assigned to that seed. This process is repeated until the seed coordinates stabilise. This algorithm aims at minimising an objective function, J, typically a squared error function:

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n} d_{ij} = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\| x_i^{(j)} - c_j \right\|^2 , \qquad (1)$$

where d_{ij} is the distance measure between a data point x_i and the cluster centre c_j. J is an indicator of the distance of the n data points from their respective cluster centres and it represents the compactness of the clusters created.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The K-means algorithm is popular because it is easy to understand and easy to implement. The main drawback is that its complexity depends linearly in the number of patterns involved and in the number of clusters selected. Another problem is that it is sensitive to the initial seeds, and may converge to a local minimum if the initial partition is not properly chosen. A possible remedy is to run the algorithm with a number of different initial seeds. If they all lead to the same final partition, this implies that the global minimum of the square error has been achieved. However, this can be time-consuming, and may not always work.

2) Basic Algorithms: Hierarchical Clustering Techniques

The main problem of non-hierarchical approaches is that when working with high dimensional problems, in general, there will not be enough items to populate the vector space, which will imply that most dimensions will be unreliable for similarity computations. In order to solve this problem, hierarchical clustering techniques were developed. There are two types of hierarchical clustering: agglomerative and divisive. Both share a common characteristic: they create a hierarchy of clusters. The agglomerative approach creates a bottom-up hierarchy while the divisive approach produces a top-down one. Generally speaking, divisive algorithms are computationally less efficient. A typical hierarchical agglomerative clustering algorithm is outlined below:

1) Place each pattern in a separate cluster;
2) Compute the proximity matrix of all the inter-pattern distances for all pairs of patterns;

3) Find the most similar pair of clusters using the matrix. Merge these two clusters into one, decrement number of clusters by one and update the proximity matrix to reflect this merge operation;
4) If all patterns are in one cluster, stop. Otherwise, go to the above step 2.

The output of such algorithm is a nested hierarchy of trees that can be cut at a desired dissimilarity level forming a partition. Hierarchical agglomerative clustering algorithms differ primarily in the way they measure the distance or similarity of two clusters where a cluster may consist of only a single object at a time. The most commonly used inter-cluster measures are:

$$d_{AB} = \min_{\substack{i \in A \\ j \in B}} (d_{ij}), \tag{2}$$

$$d_{AB} = \max_{\substack{i \in A \\ j \in B}} (d_{ij}), \tag{3}$$

$$d_{AB} = \frac{1}{n_A n_B} \sum_{i \in A} \sum_{j \in B} d_{ij}, \tag{4}$$

where d_{AB} is the dissimilarity between two clusters A and B, d_{ij} is the dissimilarity between two individual patterns i and j, n_A and n_B are the number of individuals in clusters A and B respectively. These three inter-cluster dissimilarity measures are the basis of the three of the most popular hierarchical clustering algorithms. The *single-linkage* algorithm uses equation (2), the *minimum* of the distances between all pairs of patterns drawn from the two clusters (one pattern from each cluster). The *complete-linkage* algorithm uses equation (3), the *maximum* of all pair wise distances between patterns in the two clusters. The *group-average* algorithm uses Equation (4), the average of the distances between all pairs of individuals that are made up of one individual from each cluster.

A challenging issue with hierarchical clustering is how to decide the *optimal* partition from the hierarchy. One approach is to select a partition that best fits the data in some sense, and there are many methods that have been suggested in the literature (Everitt, 1993). It has also been found that the single-linkage algorithm tends to exhibit the so-called *chaining* effect: it has a tendency to cluster together at a relatively low level objects linked by chains of intermediates. As such, the method is appropriate if one is looking for "optimally" connected clusters, rather than for homogeneous spherical clusters. The complete-linkage algorithm, on the other hand, tends to produce clusters that tightly bound or compact, and has been found to produce more useful hierarchies in many applications than the single-link algorithm (Jain and Dubes, 1999).

3) Basic Algorithms: Fuzzy Clustering

One of the most widely used fuzzy clustering algorithms is the Fuzzy C-Means (FCM) Algorithm (Bezdek, 1981). The FCM algorithm attempts to partition a finite collection of elements $X=\{x_1,\ldots,x_n\}$ into a collection of c fuzzy clusters with respect to some given criterion. Given a finite set of data, the algorithm returns a list of c cluster centres $C=\{c_1,\ldots,c_c\}$and a partition matrix $U=u_{i,j} \in [0,1], i=1,\ldots n, j=1,\ldots,c$, where each element tells the degree to which element x_i belongs to cluster c_j. Like the k-means algorithm, the fuzzy c-means aims to minimise an objective function. The standard function is:

$$J=\sum_{j=1}^{c}\sum_{i=1}^{n}(u_{i,j})^m \left\| x_i^{(j)}-c_j \right\|^2, \tag{5}$$

which differs from the k-means objective function by the addition of the membership values u_{ij} and the fuzzifier m. The fuzzifier m determines the level of cluster fuzziness. A large m results in smaller memberships u_{ij} and hence, fuzzier clusters. In the limit $m=1$, the memberships u_{ij} converge to 0 or 1, which implies a crisp partitioning. In the absence of experimentation or domain knowledge, m is commonly set to 2. The basic Fuzzy C-Means Algorithm, given n data points $(x_1,\ldots,x_n)$ to be clustered, a number of c clusters with $(c_1,\ldots,c_c)$ the centre of the clusters, and m the level of cluster fuzziness with,

$$m \in \mathbb{R} > 1, \tag{6}$$

first initialises the membership matrix U to random values, verifying that:

$$u_{ij} \in [0,1], \sum_{j=1}^{c} u_{ij} = 1 \tag{7}$$

After the initialisation, the algorithm obtains the centre of the clusters $c_j, j=1,\ldots,c$:

$$c_j = \frac{\sum_{i=1}^{n}(u_{ij})^m x_i}{\sum_{i=1}^{n}(u_{ij})^m} \tag{8}$$

And obtains the distance between all points $i=1,\ldots,n$ and all cluster centres $j=1,\ldots,c$

$$d_{ij} = \left\| x_i^{(j)} - c_j \right\| \cdot \tag{9}$$

Updating matrix U according to the new distances,

$$
\begin{aligned}
& d_{ij} = 0 \Rightarrow u_{ij} = 1 \\
& u_{ij} = \left[\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{ik}} \right)^{\frac{2}{m-1}} \right]^{-1}
\end{aligned}
\qquad (10)
$$

This process is repeated until the set of cluster centres is stabilised. There are other algorithms, which are optimizations of the original FCM, like Fuzzy c-Medoid Algorithm (FCMdd) or the Fuzzy c-Trimered Medoids Algorithm (FCTMdd) (Krishnapuram et al., 2001).

4) Applications for User Modelling

For user modelling (UM), there are two kinds of interesting clusters to be discovered: usage clusters and page clusters. Clustering of usage tends to establish groups of users exhibiting similar browsing patterns, which are usually called stereotypes. Such knowledge is especially useful for inferring user demographics in order to perform market segmentation in e-commerce-applications or provide personalised Web content 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 (Mobasher et al., 2001).

In the context of UM, clustering has a distinguishable characteristic: it is usually done with non-numerical data. This implies that, usually, the clustering techniques applied are relational, where numerical values represent the degrees to which two objects of the data set are related. Clustering applied to user modelling has to use techniques that can handle relational data because the information used to create clusters (pages visited, characteristics of the user, etc.) cannot usually be represented by numerical vectors. In case they are represented using vectors, part of the semantic of the original data is lost. In these systems, the definition of distance is done using vectorial representations of user interactions with the personalised hypermedia system (Mobasher and Cooley, 2000).

Table 2.1 summarises some studies and applications of "hard" clustering for UM. Some examples of recommendation tasks implemented with clustering algorithms are presented in Mobasher and Cooley (2000), Fu et al. (1999), Mobasher et al. (2001). Examples of classification tasks implemented using clustering are presented in Hay et al. (2001).

Table 2.1: Examples of Clustering-based User Models

	Application	Input Data	Outcome
Mobasher and Cooley, 2000	Capture of web-users interests using k-means Clustering. Design of a cluster-based recommendation system	User logs from the Univ. of Minnesota Comp. Science web server collected during a month.	Example of the implementation of the recommendation system in a commercial site.
Paliouras et al., 2000	News- filtering system based on communities of users. Clustering allows to recommend interesting news to a user.	When registering a user specifies his/her interests.	Established machine learning techniques are very useful for the acquisition of communities of users
Doux et al., 1997	K-means clustering algorithm for user profiling in order to derive prototypical behavior from each user.	Data collected from a dedicated set of experiments where users are asked about their preferences.	The techniques proposed handle qualitative data for clustering users efficiently.
Fu et al., 1999	Grouping of users with a common behaviour in a web server taking into account access patterns.	Data collected from UMR web server log (www.umr.edu) containing 2.5 million records.	The clusters obtained can be used for personalisation purposes.
Hay et al., 2001	Clustering methods that capture the inherent sequentiality of web visits. A metric, Sequence Alignment Method, is introduced to be used instead of Euclidean distance for clustering purposes.	Log files of a Belgian telecom provider collected over a one-week period.	The results are as good as the ones obtained with Euclidean distance, while keeping the concept of order.
Mobasher et al., 2001	Clustering for collaborative filtering	12,000 sessions collected from the Association for Consumer Research web site.	With the proper data preprocessing the clustering approach outperforms more traditional approaches to this problem

When using Fuzzy Clustering (FC), a user can be at the same time in more than one cluster with different degrees of truth. This allows to better capture the inherent uncertainty that the problem of modelling user behaviour has. Examples of applications that implement a recommendation task using FC include Lampinen and Koivisto (2002) and Nasraoui et al. (1999). Examples of classification tasks are presented in Joshi et al. (2000) and Krishnapuram et al. (2001). Table 2.2 summarises some studies and applications of FC for UM.

Table 2.2: Examples of Fuzzy Clustering-based User Models

	Application	Input Data	Outcome
Lampinen and Koivisto , 2002	Obtain application profiles from network traffic data to manage network resources.	274000 samples of different applications from an edge router of a LAN network.	FCM produced better results than SOM. A method for the comparison of both solutions is also introduced.
Nasraoui et al. , 1999	A new algorithm (CARD) to mine user profiles from access logs is proposed.	12 day log data of the Dep. of Comp. Eng.. at Univ. of Missouri.	CARD is very effective for clustering many different profiles in user sessions.
Joshi et al. , 2000	Two algorithms to mine user profiles: FCMdd and FCTMdd.	CSEE logs of Univ. of Maryland	Both algorithms extract interesting user profiles. FCMdd is not able to handle noise as effectively as FCTMdd.
Krishnapuram et al. , 2001	Web access log analysis for user profiling using RFCMdd (Robust Fuzzy c-Medoids).	Five days of CSEE web server activity of Univ. of Maryland.	RFCMdd is very effective for clustering of relational data.

5) Clustering Limitations for User Modelling

The main problems that clustering techniques face are: (1) how to define the concept of distance that is going to be used and (2) for non-hierarchical clustering, that the algorithms are constructed using a number of clusters known a priori.

Regarding the definition of distance, in general, some knowledge of the problem is needed to define an optimum concept of distance. 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 are not expressed in a numerical way. Different techniques to characterise user behaviour using numerical vectors have been proposed (Joshi et al., 2000; Mobasher and Cooley, 2000), but in one way or another, the representations loose part of the semantics that the original data had.

Non-hierarchical clustering techniques assume that the number of clusters k is known a prioiri. For user modelling this is not usually the case. This implies that some heuristics need to be used to determine the number of clusters. The following two subsections present some techniques used to estimate the optimum number of clusters for k-means and fuzzy clustering.

(a) Determining the Optimum Number of Clusters for K-means

K-means algorithm has as inputs the number k of clusters (stereotypes in user modelling) used to partition the original data, the concept of distance used to measure the distance between two elements, and, if desired, k cluster centres used to initialise each cluster. If the cluster centres are not given, the algorithm assigns them randomly.

The method for determining the optimum number of clusters is based on the idea that the optimum partition is the one that maximises the compactness of the clusters. In order to measure the compactness of the partition for a given value of *k*, for each element *i* the method obtains an indication ϕ_i representing how similar that element is with the rest of the elements of the same cluster compared with all the items of all other clusters, formally:

$$\phi_i = \frac{\min(b_{i,m}, m = 1,\ldots,k) - d_i}{\max(d_i, \min(b_{i,m}, m = 1,\ldots,k))}, \tag{11}$$

where ϕ_i is a value ranging in $[-1, +1]$, d_i is the average distance of user *i* to all the users of its own cluster, $b_{i,k}$ is the average distance of user *i* to all the users of cluster *k*, and *m* is the number of user stereotypes. A value of $+1$ indicates that the element is very distant to the rest of the clusters, a value of 0 or near 0 indicates that the user is not distinctive of that cluster, and a negative value that indicates that the user has probably been assigned to the wrong cluster. The quality of a partition, q_k, with *k* the number of clusters can be obtained as the mean value of all the ϕ_i values of the system,

$$q_k = \frac{\sum_{i=1}^{N} \phi_i}{N}, \tag{12}$$

with *N* the number of items. Usually q_k is obtained for a set of values of *k*, for example for $k=2,\ldots,9$, and the optimum number of clusters *k* is defined as the value of *k* that maximises q_k, i.e. the compactness of the clusters. Also, usually, to avoid that the solution given for a given *k* is a local minima, because of the randomness of the original centres, for each value of *k* the algorithm is run T times and the solution considered is the one that minimises the objective function *J* presented in (1).

From a user modelling perspective, what this technique measures is how similar the behaviour of a user (item or element) is to the rest of the users of the cluster in which it is included compared with the rest of the users of the system. The solution obtained is the number of clusters *k* that maximises the compactness of the behaviour expressed by the clusters of the system.

(b) Determining the Optimum Number of Clusters for Fuzzy Clustering

Fuzzy Clustering needs to know in advance the number of clusters in which the data is going to be classified. A technique that is useful for estimating the number of clusters is subtractive clustering (Chiu, 1994). Subtractive clustering is a one-pass algorithm for estimating the number of clusters and the cluster centres in a set of data. Basically, the algorithm assumes that each data point is a potential cluster centre and calculates a measure of the likelihood of that point of being cluster centre based on the density of surrounding points

Subtractive clustering has as inputs the bounds in which each dimension of the input vector operates, in order to normalise them, and a *radii* of influence, which is used to determine the size of the possible clusters. Good values for the *radii* are around *[0.2, 0.5]*. Small *radii* values produce clusters with reduced influence areas and partitions with a high number of clusters, while higher values create partitions with a small number of clusters. The algorithm has the following steps:

1) For each point, the algorithm obtains the density of surrounding points using *radii*.
2) The set of data points with the highest potential to be a cluster (with the higher density) are selected.
3) The data points within the vicinity of the cluster determined by *radii* are removed.
4) The process is iterated until all data is included in one of the clusters.
5) The algorithm outputs the optimum number of clusters and candidates for the centres of those clusters.

2.3.2 Robust Clustering for User Modelling

Although the previous techniques have successfully been used for user modelling, they face some problems: (1) the bias of each techniques and (2) the lack of filtering capabilities of problematic items of the original dataset. In terms of the former, each technique presented has a bias that deeply affects its results. For example with k-means is the concept of distance selected and the randomness of the process of selecting the initial centres of the cluster; for fuzzy clustering is the concept of *radii* and again the randomness of initialising the cluster centres; and for hierarchical clustering the distance used to aggregate users. In respects of the latter, in the context of user modelling, the data is very noisy because of the inherent fuzziness of capturing human behaviour. That implies that, because the techniques do not filter any users, and because users can show behaviour that actually is not relevant for user modelling, the behaviour captured by each cluster is blurred by the addition of these ill-defined users.

Robust Clustering (Swift et al., 2004) is an algorithm originally developed for clustering highly similar gene-expression vectors. The algorithm basically creates clusters based on the information of other clustering techniques, creating clusters only if all the clustering techniques agree. As a collateral effect, elements for which the techniques do not agree are filtered from the final classification. This method solves the problems that using individual clustering methods have for user modelling: (1) it eliminates the bias of the techniques, due to the fact that clusters are created only if all techniques agree, and (2) it filters users that do not have a well-defined behaviour, because for these users one or more than one technique will not agree. Robust clustering (RC) is based on compiling the results of different clustering methods and on reporting only the elements that are co-clustered together by all the different algorithms. For two elements, all clustering methods must have allocated them to the same cluster in order for them to be assigned to a robust cluster. This gives a higher level of confidence to the correct assignment of elements appearing within the same cluster.

1) Basic Algorithm

RC (Swift et al., 2004) is based on an agreement matrix. The agreement matrix, of size $n \times n$, with n the number of elements to be clustered, is an upper triangular matrix that indicates for each combination of elements the number of agreements among the methods for clustering together the two variables, represented by the row and the column indices. RC uses the agreement matrix to generate an agreement list that contains all the pairs of elements of the matrix where the value is equal to the number of clustering methods used, C. Then, starting with an empty set of robust clusters, the first element created contains the elements of the first pair of the list of agreement. Then, the algorithm iterates for the rest of the elements of the agreement list, where, if one element of the current pair is found in a robust cluster and the other is not, that element is added to the robust cluster, otherwise a new robust cluster is created. After the algorithm has iterated for each pair of elements of the agreement list, it outputs the set of robust clusters found. Figure 2.3 presents a description of the algorithm that implements robust clustering.

Input : Agreement Matrix $A(n\ x\ n)$, C number of clustering techniques
Output: $RC = \{RC_1, ..., RC_m\}$ Set of robust clusters

$AgreementList$=Pairs (x, y) of A, with $A(x, y) = C$
$RC = \{\}$
$RC_1 = AgreementList_1$
For $i = 1$ **to** $Size(AgreementList)$
  $Found = False$
  For $j = 1$ **to** $Size(RC)$
    If $(AgreementList_{i,1}$ **or** $AgreementList_{i,2}) \in RC_j$
      $Found = True$
      If $AgreementList_{i,1} \notin RC_j$
        $RC_j = RC_j \cup AgreementList_{i,1}$
      end_if
      If $AgreementList_{i,2} \notin RC_j$
        $RC_j = RC_j \cup AgreementList_{i,2}$
      end_if
      $j = Size(RC)$
    end_if
  end_for
  if $NOT\ found$
    $RC = RC \cup \{AgreementList_i\}$
  end_if
end_for

Figure 2.3: Robust Clustering Algorithm

In general, the set of robust clusters obtained will not contain all the original elements, because those elements for which the clustering techniques used do not agree, will not be included into the final set of robust cluster. This filtering property of RC is very useful in user modelling because it eliminates users that can not be robustly grouped with other users. This helps to eliminates users that introduce fuzziness in the definition of the behaviour of a cluster. In summary, RC is a very valuable technique for a rapid drilling-down of datasets into clusters whose pattern is identified in a manner that is independent of the cluster method, thus eliminating the bias of each technique.

2) Applications for User Modelling and Limitations

RC has produced very good results for computational genetics for which was originally developed (Swift et al., 2004). RC is a very attractive option for user modelling, due to its filtering capabilities. The inherent fuzziness of dealing with human data makes it essential to be able to filter users that are not representative of any behaviour. One of the novelties presented in this thesis is the application of robust clustering to modelling user behaviour.

It obviously has some problems, mainly its complexity. The algorithm itself, once the agreement matrix has been obtained, is not very complex. Nevertheless, it requires that previously other clustering techniques have been run, thus inheriting and adding their complexity. An interesting topic is to identify which techniques should be included to construct the robust set of clusters. Ideally, each clustering family of algorithms (hierarchical, non-hierarchical and fuzzy) should be represented. Also, the number of techniques used can deeply affect the results obtained because a high number of techniques would probably filter a big part of the original data.

2.3.3 Supervised Learning Approaches for User Modelling

This section gives a review of how the supervised learning techniques used in this thesis have been used to model user behaviour, including Decision Trees and Neural Networks.

1) Decision Trees for User Modelling

Decision tree (Mitchell, 1997; Winston, 1992) is a method for approximating discrete-valued functions with disjunctive expressions. Decision tree is generally best suited to problems where instances are represented by attribute-value pairs and the target function has discrete output values.

(a) Basic Algorithms

The training process that creates a decision tree is called induction. A standard decision tree algorithm has two phases: (1) tree growing and (2) pruning. The growing phase can be done using two methods: (1) Top-Down induction and (2) Incremental induction (Mitchell, 1997).

Top-down induction is an iterative process which involves splitting the data into progressively smaller subsets. Each iteration considers the data in only one node. The first iteration considers the root node that contains all the data. Subsequent iterations work on derivative nodes that will contain subsets of the data. The algorithm begins by analysing the data to find the independent variable that, when used as a splitting rule will result in nodes that are most different from each other with respect to the dependent variable. The quality of a test is measured by the impurity/variance of example sets. The most common measure is the information gain. Typically, the set of possible tests is limited to splitting the examples according to the value of a certain attribute. Once a node is split, the same process is performed on the new nodes, each of which contains a subset of the data in the parent node. This process is repeated until only nodes where no splits should be made remain.

Incremental induction is a method for the task of concept learning. When a new training example is entered, it is classified by the decision tree. If it is incorrectly classified, then the tree is revised. Restructuring the tree can be done by storing all training examples or by maintaining statistics associated with nodes in the tree.

Tree-building algorithms usually have several stopping rules. These rules are usually based on several factors, including maximum tree depth, minimum number of elements in a node considered for splitting, or the minimum number of elements that must be in a new node.

The second phase of the algorithm optimises the resulting tree obtained in the first phase. Pruning is a technique used to make a tree more general. It removes splits and the subtrees are created by them. There is a great variety of different decision tree algorithms in the literature. Some of the more common algorithms are: Classification and Regression Trees (CART) (Breiman et al., 1984; Efron and Tibshirani, 1991), CHAID (Kass, 1980), C4.5 (Quinlan, 1993), C5.0 (Witten and Frank, 1999) and ID3 (Quinlan, 1993).

Classification rules (Hand, 1997) are an alternative representation of the knowledge obtained from decision trees. They construct a profile of items belonging to a particular group according to their common attributes. Rules are, at its simplest form, an equivalent form of expressing a decision tree. In order to obtain the set of rules of a decision tree, each path is traced from root node to leaf node, recording the test outcomes as antecedents and the leaf-node classification as the consequent. Algorithms, such as CART, C4.5 and C.5, include methods to generate rules.

(b) Applications for User Modelling

In the context of user modelling, decision trees can be used to classify users and/or documents in order to use this information for personalisation purposes. Table 2.3 summarises some studies and applications of Decision Trees for user modelling. Decision trees are typically used to implement classification tasks. In this case, the decision trees are used to construct user models based on a particular characteristic, for example regarding his/her level of experience, his/her cognitive style, etc (Tsukada and Washio, 2001; Beck et al., 2003). Due to its ability to group users with similar characteristics, decision tress can be also applied to implement recommendation tasks (Paliouras et al., 1999; Zhu et al., 2003).

Table 2.3: Examples of Decision Tree-based User Models

	Application	Input Data	Outcome
Paliouras et al., 1999	Construction of user stereotypes using C4.5. Stereotypes are used in a news retrieval system.	Questions answered by 31 users regarding the ECRAN information system.	It is very important to have good data in order to obtain good models.
Tsukada and Washio, 2001	Automatic classification of web pages in a pre-specified set of categories using C4.5 and association rules.	14 top categories of Yahoo! JAPAN. From each category 200 pages.	This method provides acceptable accuracy with the classification of web-page into top categories of Yahoo! JAPAN.
Webb et al., 1997	Use of C4.5 to build the Feature Based Modelling instruction module. The results are applied to the Subtraction Modeller.	Test administered to 73 nine to ten year old primary school students.	C4.5 increases the number of predictions made.
Zhu et al., 2003	Construction of a recommender system to help users find relevant information on the web using C4.5 and Naïve Bayesian Classifier.	Collected data from 129 participants, asking each participant to perform two search tasks.	C4.5 outperforms Naïve Bayesian Classifier.
Beck et al., 2003	Construction of a User Model for an adaptive tutor with C5.0 and Naïve Bayesian classifier.	Data collected from the interaction of 88 students with the Reading Tutor.	Naïve Bayesian Classifier outperforms C5.0 for individual modelling and C5.0 outperforms Naïve Bayesian Classifier for Group modelling.

(c) Limitations

Decision trees/Classification rules produce results that are highly dependent on the quality of the data available. The reason for that is the fact that subtrees are created using the maximum information possibly gained. In some cases, if the information available is not appropriate, which typically happens when the information used to create user models has been obtained using user feedback or in a noisy environment, the models created will not correctly capture user preferences. Also, decision trees have the problem that for high dimensional problems, the response time can be very high. This is an inconvenient when working with personalised systems, because real-time response is needed. This problem can be solved in some cases using classification rules.

Currently, special interest for user modelling has the combination of classification rules with soft computing techniques (fuzzy logic and neural networks especially) in order to create more flexible user models (Pal et al., 2002). Fuzzy classification rules are able to overlap user models and to improve the interpretability of the results.

2) Neural Networks for User Modelling

A Neural Network (NN) is an information processing paradigm that is inspired by the way biological nervous systems process information (Fausett, 1994; Haykin, 1999). Although typical ANNs are designed to solve supervised learning problems, there are also architectures to solve unsupervised learning problems.

(a) Basic Concepts

The key element of this paradigm is the structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in parallel. They consist of the following elements: (1) Neurones, (2) Weighted interconnections, (3) An activation rule, to propagate signals through the network and (4) learning algorithm, specifying how weights are adjusted.

The basic element of any NN is a neuron (Figure 2.4). A neuron has *N* weighted input lines and a single output. The neuron will combine these weighted inputs by forming their sum and, with reference to an activation function and a threshold value; it will determine its output.

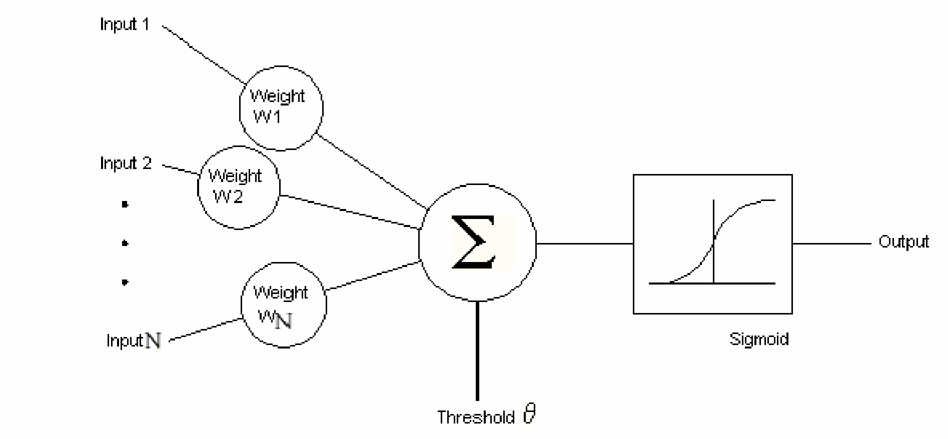


Figure 2.4: Architecture of an Artificial Neuron

Being $x_1, x_2, \ldots, x_N$ the input signals, $w_1, \ldots, w_N$ the synaptic weights, u the activation potential, θ the threshold and y the output signal and f the activation function:

$$u = \sum_{i=1}^{N} w_i x_i \tag{13}$$

$$y = f(u - \theta) \tag{14}$$

Defining $w_0=\theta$ and $x_0=-1$, the output of the system can be reformulated as:

$$y = f\left(\sum_{i=0}^{N} w_i x_i\right). \tag{15}$$

The activation function *f* defines the output of the neuron in terms of the activity level at its input. The most common form of activation function used is the sigmoid function.

There are very different ways in which a set of neurons can be connected among them. The traditional cluster of artificial neurons is called neural network. Neural networks are basically divided in three layers: The Input Layer, The Hidden Layer, which may contain one ore more layers, and the output layer.

The layer of input neurons receives the data either from input files or directly from electronic sensors in real-time applications. The output layer sends information directly to the outside world, to a secondary computer process, or to other devices such as a mechanical control system. Between these two layers can be many hidden layers. These internal layers contain many of the neurons in various interconnected structures. The inputs and outputs of each of these hidden neurons simply go to other neurons. In most networks, each neuron in a hidden layer receives the signals from all of the neurons in a layer above it, typically an input layer. After a neuron performs its function, the output is passed to all of the neurons in the layer below it, providing a feed forward path to the output. Another type of connection is feedback. This is where the output of one layer routes back to a previous layer.

Multi-Layer Perceptrons are the typical architecture of NNs. MLP are full-connected feed-forward nets with one or more layers of nodes between the input and the output nodes.

Classification and recognition capabilities of NNs stem from the non-linearities used within the nodes. A single-layered perceptron implements a single hyperplane. A two-layer perceptron implements arbitrary convex regions consisting of intersection of hyperplanes. A three-layer NN implements decision surfaces of arbitrary complexity (Lippmann, 1987; Looney, 1997). That is the reason why a three layer NN is the most typical architecture.

NNs learn through an iterative process of adjustments. There are two training approaches: supervised and unsupervised. In supervised training, both the inputs and the outputs are provided. The net is trained by initially selecting small random weights and internal thresholds, and presenting all training data repeatedly. Weights are adjusted after every trial using information specifying the correct class until weights converge and the cost function is reduced to an acceptable value. The vast bulk of networks utilises supervised training. The most common supervised technique is the back-propagation learning algorithm. It uses a gradient search technique to minimise a cost function defined by the mean square error (MSE) between the desired and the actual net outputs, with l the number of training points:

$$MSE = \frac{1}{l}\sum_{i=1}^{l}(y_i - \hat{y}_i)^2 \qquad (16)$$

The generally good performance found for the back-propagation algorithm is somewhat surprising, considering that it is a gradient descent technique that may find a local minimum in the cost function, instead of the desired global minimum.

(b) Applications for User Modelling

NNs are able to derive meaning from complicated and/or imprecise data. Also, NN do not require the definition of any metric (unlike k-NN or clustering) which make them completely application independent. No initial knowledge about the problem that is going to be solved is needed. These characteristics make NNs as a powerful method to model human behaviour and an ideal technique to create user models for adaptive hypermedia applications.

NNs have been used for classification and recommendation in order to group together users with the same characteristics and create profiles and stereotypes. Bidel et al. (2003) is an example of NNs used for classification, Sas et al. (2003) and Sheperd et al. (2002) are some examples of NNs used for recommendation tasks. Table 2.4 presents more details of these applications.

Table 2.4: Examples of NNs-based User Models

	Application	Input Data	Outcome
Bidel et al., 2003	Classification and tracking of user navigation.	Data generated from an on-line encyclopedia.	A labeled approach to the problem produces better accuracy.
Sas et al., 2003	Prediction of user's next step in a virtual environment	30 users performed exploration and searching within the environment.	Very accurate predictions of the next step
Sheperd et al., 2002	Adaptive filtering system for electronic news using stereotypes.	The Halifax Herald Ltd.	Very useful for readers with specific information needs.
Roh et al. , 2003	Three step recommendation model based on collaborative filtering that combines NN with case-cased reasoning.	MoviLens data sets (GroupLens Research Project, Univ. of Minnesota) containing ratings of movies.	The new algorithm gives useful recommendations to each user.
Changchien and Lu, 2001	On-line recommendation system for e-commerce sites based on customer and products fragmentation.	Sample of sales records from a Database.	Recommendation knowledge can promote internet sales.
Hsieh, 2004	Modeling of bank users for marketing purposes.	Bank databases provided by a major Taiwanese credit card issuer.	Identifying model by a behavioral scoring model and facilitates customer marketing/

(c) Limitations

NNs have been successfully used for UM mainly because they do not need any heuristic to produce a model. Nevertheless, it still faces important limitations: (1) the training time needed to produce a model (which in cases can be measured in the order of many hours and even days) and (2) the amount of information needed. The training time is an inconvenience for creating dynamic models. Although there are techniques able to retrain NNs dynamically, the techniques used so far for UM retrain the system from scratch in case more information, e.g. a new user or a new document, is added. Another important limitation of NNs is their black box behaviour. While the previous techniques, to a different extent, can be interpreted and manually changed, NNs cannot be interpreted, which limits its applications.

2.3.4 Soft Computing Approaches to User Modelling

Soft Computing (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. User interaction is critical for any hypermedia applications, which implies that the data available will usually be imprecise, incomplete and heterogeneous. In this context, SC seems to be an appropriate paradigm to handle the uncertainty and fuzziness of the data available to create user models (Pal et al., 2002). The elements that a user model captures (including goals, plans, preferences, common characteristics of users) can exploit the ability of SC to mix different behaviour and to capture human interaction 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 Network offers a flexible mechanism for the representation of common characteristics of a user and the definition of complex stereotypes; Fuzzy Clustering supplies a mechanism in which a user can be part of more than one stereotype at the same time; and Neuro-Fuzzy systems presents a mechanism to capture and tune expert knowledge which can be used to obtain assumptions about the user. This section presents how the SC techniques used in this thesis, fuzzy logic and neuro-fuzzy systems have been used for user modelling. Although neural networks and fuzzy clustering can also be considered soft computing techniques, they have been presented as part of supervised learning techniques and unsupervised learning techniques respectively.

1) Fuzzy Logic for User Modelling

Fuzzy Logic (FL) defines a framework in which the inherent ambiguity of real information can be captured, modelled and used to reason under uncertainty (Klir and Yuan, 1995; Yan et al. 1994). 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 with 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 that is inherent to real data.

FL is not strictly a data mining technique but a technique for representing information. Nevertheless, due to its ability to handle uncertainty, it is used in combination with other data mining techniques in order to produce behaviour models that are able to capture and manage the uncertainty of human behaviour. Some examples of these combinations are Fuzzy Clustering, or Fuzzy Association Rules. 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 user modelling does not necessarily realise all of the three steps, but maybe only a subset of them.

(a) Basic Algorithms

The key concept that introduces FL is the concept of fuzzy set. A Fuzzy Set describes the degree of membership of a variable in that set. A Fuzzy Set *A* in *X* is defined as:

$$A = \{(x, \mu_A(x)) / x \in X\}, \mu_A(X) \to [0,1] , \qquad (17)$$

where μa is the membership function that characterises the fuzzy set A. A set of operations that work with fuzzy sets are also defined by fuzzy logic. The three basic operations are complement, intersection and union.

Complement is a function *N* defined as *N : [0,1] → [0,1]* than verifies:

- N(0) =1 , N(1)=0
- N(a) ≥ N(b), si a ≤ b
- N(N(a))=a

Some examples of complements are:

- *N(a) = 1 -a*

- N(a) = $\dfrac{1-a}{1+sa}$ Sugeno's complement

The intersection of two fuzzy sets A and B is defined as a function T : [0,1] → [0,1].The intersection operator is also called T-norm. Some examples of typical T-norms are:

- Minimum: $T_{min}(a,b) = \min(a,b)$
- Product: $T_{al} = ab$

The union operator (also called S-norm) is defined as a function S : [0,1] → [0,1]. Some examples of traditional S-norm operators are:

- Maximum: $S_{max}(a,b) = \max(a,b)$
- Sum: $S_{al}(a,b) = a+b-ab$

Fuzzy Inference Systems are constructed using a set membership functions for each input (also called linguistic labels) and fuzzy inference rules. Fuzzy inference rules take the form ``*IF x is a, THEN y is b''*, where *x* and *y* are inputs of the system and *a* and *b* are membership functions defined in *x* and *y* respectively. Under classical logic, the *THEN* implication is true if the antecedent is evaluated as true. For fuzzy rules, the implication is set to be true to the same degree as the antecedent. The process of fuzzy inference is divided into four steps:

- Fuzzification: Fuzzification is the process of determining the degree of membership the data has to all appropriate fuzzy sets. In this step the degree of truth of each input in the set of membership functions defined for that input is obtained.
- Rule Evaluation: The degree of truth of the antecedent of a rule is obtained combining the different degrees of truth of each input using the T-norm (AND) and T-conorm (OR) operators. Once the degree of truth of the antecedent is obtained, this degree is passed onto the consequent (or consequents) using a T-norm operator. This is done for each rule of the fuzzy knowledge base.
- Combination of Rule Consequents: As a result of the previous step, the system will have as many consequents as rules. The set of all consequents is aggregated using a T-conorm operator.
- Defuzzification: Deffuzification is the process of transforming the fuzzy membership function obtained from aggregating the consequents of all rules into

a real number. There are a variety of differing definitions for defuzzification, the most intuitive and common being the centre of mass.

Depending on the definition of the parameters of the system, different types of fuzzy inference systems are obtained. For example if T-norm and T-conorm are defined as maximum and minimum, and the deffuzification as centre of gravity, the system obtained is a Mamdani inference system. If the T-norm and T-conorm are defined as product and sum, and the membership functions of the outputs of the system are defined as singletons (Kronecker Delta), we obtain a Takagi-Sugeno system. Mamdani inference typically is used when a system aims to emulate the intuitive human expert thought process. Takagi-Sugeno inference is used in optimisation and adaptive algorithms, particularly for control systems.

(b) Applications for User Modelling

Typically FL has been employed in recommendation 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 and Petenes (2003), which uses user profiles obtained with hierarchical unsupervised clustering. In Ardissono and Goy (2000), FL is used to model user behaviour and provide recommendations using this fuzzy behaviour model. Although, strictly speaking, there is no actual fuzzy inference involved, the stereotypes that characterise users are modelled using membership functions, and the recommendation process is done using a fuzzy *AND* operator. Table 2.5 presents some applications of user modelling using FL as part of their modelling or reasoning architecture.

Table 2.5: Examples of Fuzzy Logic-based User Models

	Application	Input Data	Outcome
Nasraoui and Petenes (2003)	Web recommendation system based on a fuzzy inference engine that uses a rule-based representation of the user profile.	12 days access log data of the Web site of the Dep. Comp. Eng. at the University of Missouri.	Fuzzy recommendation achieves high coverage compared to other data mining solutions.
Vrettos and Stafylopatis (2001)	Agent for information retrieval and filtering in the context of e-learning.	Cranfield data set (www.cs.utk.edu/lsi) which includes 1398 documents, 225 queries and an average of 8.2 relevant documents per query.	Re-ranking the search according to user's profile.
Ardissono and Goy (2000)	Introduction of personalisation techniques in a shell supporting the construction of adaptive web stores.	Not Presented.	Fuzzy logic can be applied in electronic sales to produce personalised environments.
Schmitt et al. (2003)	Recommendation of items of an e-commerce site to its users using a structure-based system.	Preferences specified by the user.	On-line demo: www2.dfki.de:8080 /mautmachine.html
Kuo and Chen (2004)	Decision support system that integrates both qualitative and quantitative factors	Simulation	Considering both qualitative and quantitative factors produces more accurate results that considering only quantitative factors.

(c) Limitations

Although FL is an ideal technique for modelling 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 later, have emerged as an approach to alleviate these challenging situations.

2) Neuro-Fuzzy Systems for User Modelling

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 (Jang and Sun, 1995). With this approach, the drawbacks of NNs and FL, the black box behaviour 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. One of the most typical NFS is ANFIS (Adaptive-Network-based Fuzzy Inference Systems) (Jang, 1993), which has been used in a wide range of applications (Bonisone et al., 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.

(a) Basic Algorithms

Prior to training a neuro-fuzzy system, a number of membership functions and their type must be assigned to each input. For training, the input search space is partitioned into a grid using the membership functions chosen along each of the input dimensions. Knowledge is then captured during training in the form of fuzzy *If-Then* rules. Each rule describes the output of the system in a particular cell given the input conditions. One possible architecture of a NFS is shown in Figure 2.5, which contains three different layers: (1) fuzzification layer, (2) fuzzy rule layer and (3) defuzzification layer. In the fuzzification layer, each neuron represents an input membership function of the antecedent of a fuzzy rule. In the fuzzy inference layer, fuzzy rules are fired and the value at the end of each rule represents the initial weight of the rule. In the defuzzification layer, each neuron represents a consequent proposition. After getting the corresponding output, the adjustment is made in the connection weights and the membership functions in order to compensate the error.

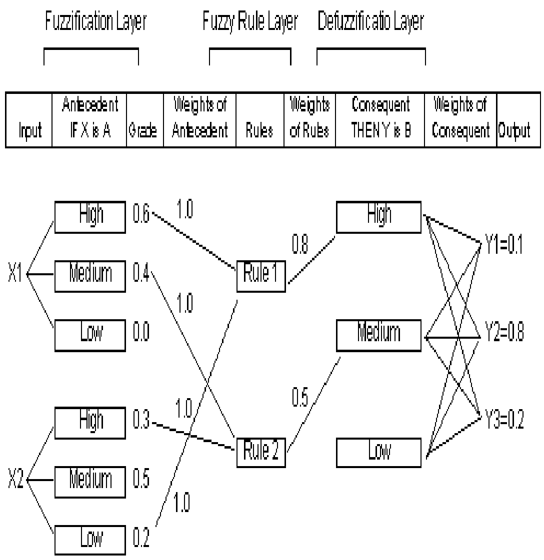


Figure 2.5: Typical NFS Architecture

(b) Applications for User Modelling

The combination of NN and fuzzy sets offers a powerful method to model human behaviour which allows NFS to be used for a variety of tasks. Lee (2001) and Stathacopoulou et al. (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 based on user and enterprises profile data. Magoulas et al. (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. Table 2.6 summarises studies and applications of NFS for user modelling.

Table 2.6: Examples of NFS-based User Models

	Application	Training Data	Outcome
Lee (2001)	Mobile web shopping agent that finds products that suit user needs using a NFS and FL.	A test is implemented using a product data-base with 200 items and 8 categories.	Provides a more efficient result when compared with other solutions; processing time is shorter.
Stathacopoulou et al. (2003)	Student Modelling	A set of simulated students.	High accuracy in the diagnosis of student problems during learning.
Magoulas et al. (2001)	Intelligent decision making for recommending educational content in a web-based course.	"Introduction to Computer Science" course of the Univ. of Athens.	Successful handling of fuzziness associated with the evaluation of learner's knowledge.
George and Cardullo (1999)	Modelling of human behaviour.	10 subjects collected data for the one dimensional compensatory task.	Generate a model of human behaviour.
Drigas et al. (2004)	Assignation of jobs to unemployed people using enterprises profile data.	General Secretariat of Social Training database (Greece).	Age and Previous Experience of the applicants seem to be the most determinant fields.

(c) Limitations

The basic idea of combining fuzzy systems and neural networks is to design an architecture that uses a fuzzy system to represent knowledge in an interpretable manner and the learning ability of a neural network to optimise its parameters. The drawbacks of both of the individual approaches - the black box behaviour of neural networks, and the problems of finding suitable membership values for fuzzy systems - could thus be avoided. Nevertheless, NFS still maintains some of the limitations of both approaches, mainly the training time needed for dynamic modelling. NFS can be used as an interpretable model that is capable of learning and can use problem-specific prior knowledge. Therefore, neuro-fuzzy methods are especially suitable for applications, where user interaction in model design or interpretation is desired.

2.4 Criteria for the Selection of the Techniques

The previous sections have shown the variety of possibilities that data mining techniques offer to model user behaviour. Nevertheless, each technique has its own strengths and weaknesses, represents the information in different ways, has different complexities and needs different type of input data. It is then essential to give some criteria for the selection of suitable techniques.

We consider that, in the context of UM, there are three main criteria that determine which data mining technique is suitable for a specific personalised application: (1) the labelled/unlabelled nature of the data available; (2) the type of task that is going to be implemented (Recommendation or Classification) and (3) the "readability" needed for the results. "Readability" is defined as the ability of a technique to produce a human-readable output of the knowledge captured for a non-technical user {References}. There are two possible values for Readability: (1) needed and (2) not needed. The first one expresses the necessity of having a human readable output while the second one states that this factor is not relevant. Table 2.7 presents a set of guidelines of what data mining techniques are useful based on the criteria previously introduced. The techniques are classified according to the set of references used in this study. The set of techniques can be applied when the systems needs readability but it can also be applied when this factor is not relevant.

Table 2.7: Selection of Suitable Data Mining Techniques

	Labelled Data		**Unlabelled Data**	
Task	**Readability Needed**	**Readability Not Needed**	**Readability Needed**	**Readability Not Needed**
Recommendation	Decision Trees NFS Fuzzy Logic	NNs		K-means Clustering Fuzzy Clustering
Classification	Decision Trees Fuzzy Logic	NNs		K-means Clustering Fuzzy Clustering

When selecting a data mining technique, two of the more important factors are (1) the ability to handle high dimensional data and (2) scalability. Although in a generic context the ability of a technique to handle high dimensional data is a very important characteristic, for user modelling, it is not. The main reason is that typically the dimension of the data available for each user is not high because of the difficulties of capturing and representing human-interaction data. Nevertheless, in the context of user modelling, the scalability of the techniques is a very important factor due to the high number of users that, in general, will interact with a personalised hypermedia system. The scalability of each technique regarding the number of users will depend on how the information of each user is presented. An indication of the scalability of each technique is presented in the first column of Table 2.8. Table 2.8 summarises the characteristics of the techniques presented along four dimensions. The first three dimensions capture some of the main problems that data mining for user modelling faces (Webb et al., 2001): Computational Complexity for off-line processing; Dynamic Modelling, which indicates the suitability of the technique to change a user model on-the-fly; and Labelled/Unlabeled. The "Readability" dimension has also been added to the table.

Table 2.8: General Characteristics of the Revised Techniques

	Off-Line Complexity (Indication of Scalability)	**Dynamic Modelling**	**Labelled / 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 and Satyanarayana, 2003).	No	Unlabeled	No
Fuzzy Clustering	$O(n^2)$ with n the number of objects For some optimised algorithms O(nlogn) (Krishnapuram et al., 2001)	No	Unlabeled	No
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 and Hirschberg, 1995).	Yes	Labelled	Yes
Neural Networks	NP-Complete for a generic 3 layer NN Polynomial for some simple two layer networks (Blum and Rivest, 1992)	Yes	Both	No
Fuzzy Logic	N/A	No	N/A	Yes
Neuro-Fuzzy Systems	The same as a Neural Network	Yes	Labelled	Yes

The combination of Table 2.7 and Table 2.8 can be used to guide a choice of which technique to use when modelling user behaviour for personalised hypermedia systems. First, Table 2.7 identifies the set of techniques suitable for the adaptive application and, after that, Table 2.8 can be used to refine the choice considering the scalability and dynamic modelling capabilities of each technique. Frias-Martinez et al. (2006) presents the same tables for a more comprehensive set of data mining techniques.

2.5 Conclusions

This chapter has presented a review of the state of the art of data mining techniques within the area of user modelling for personalised hypermedia systems. The review demonstrates that one of the main problems in developing user models is the lack of any kind of standardization for the design of such models. In order to improve this situation, the chapter has tried to give a set of guidelines that formalises the design of user models using a data mining approach. It seems that the future of user modelling will take a hybrid approach. As has been shown, each technique captures different elements of user behaviour. The combination of these techniques among themselves and with other data mining techniques, especially with soft computing techniques, will provide a useful framework to efficiently capture the natural complexity of human behaviour.

The following chapter presents how personalisation has been implemented in the focus of this thesis, Digital Libraries. The goal of the chapter is to introduce the basic concepts and architectures behind digital libraries and how personalisation, adaptive and adaptable, has been so far implemented.

Chapter 3

Adaptive and Adaptable Digital Libraries

3.1 Introduction

The previous chapter has presented a review of the state of the art of data mining for user modelling. This chapter is going to focus in the other main research areas of this thesis: digital libraries. 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 et al., 2003). The collections of information can be scientific or academic (Stelmaszewska and Blandford, 2004), medical (Adams and Blandford, 2002), 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, DLs have become, with search engines in general, one of the major web services (Liaw and Huang, 2003). Typically, DLs have a global approach in which all users are presented with the same interface, regardless of the diversity of users in terms of preferences or skills. Nevertheless, different studies in information seeking have shown that matching the interface with users' preferences can help them to achieve their tasks in a satisfactory way (Marchionini et al., 1998; Blandford et al., 2001).

As DLs become more important in our everyday activities, their contents and services become more varied, and their users expect more intelligent services. DL must move from being passive, with little adaptation to their users, to being more proactive in offering and tailoring information for individuals and communities (Callan et al., 2003). This can be done by connecting people with computers in a personal way. From this perspective, personalisation is a key tool to develop the next level of DLs.

Within the context of DL, up to now, user modelling has been implemented using mainly user-guided approaches, which has produced adaptable DLs. Nevertheless, the problem of user modelling in DL can be easily implemented using an automatic approach. This thesis is based on the idea that personalised DL established based on automatic user modelling using data mining techniques can match users' requirements so that more efficient and tailored services can be provided. Such personalised DLs are also named as "adaptive DLs".

The chapter is organised as follows: it starts by presenting the architecture, functionalities and state of the art of personalised DL. Once the main problems of the current approaches have been highlighted, the next section presents the adaptive dimension of a DL, describing also some approaches already taken to implement adaptive DL services. Subsequently, the elements that a DL user model should contain and which techniques can be used to model and capture those elements are presented.

3.2. Basic Architecture of Digital Libraries

DLs are more than web pages that give access to information. They also consist of, 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. In general, DLs are made up of four components (Theng et al., 1999):

1) Information.
2) Structure, describing the syntactic and semantic of the information.
3) Properties, referring to security, copyright issues, etc.
4) Interaction elements, referring to the searching interface, screen design of the information available in the DL, etc.

The services provided by DL through their interaction elements can be classified into three groups:

1) Mechanisms for the personalisation 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.
2) Mechanisms to help in the process of navigation. These services present each user with an environment that better suits the way in which the user interacts with the DL.

3) 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.

Figure 3.1 shows the basic architecture of a Digital Library which presents the interaction between the previous elements. In this architecture, no personalisation has been introduced.

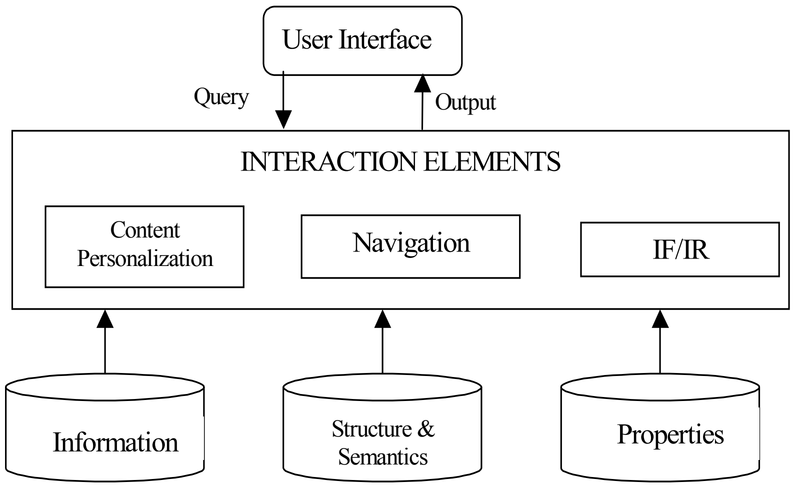


Figure 3.1: Generic Architecture of a DL

DLs comprise multiple distributed and autonomous information sources. The main architectures for organizing these sources of information are centralised and multi-searcher:

- Centralised Architecture. In the centralised architecture, the DL collects information about the documents of the different resources and constructs a local index. Searches are done in that database and the output is presented to the user.
- Multi-Searcher Architecture. A Multi-Search DL has more than one index, each storing information about documents of different resources. When a user starts a search, the interface produces a query to each one of those indexes, collects the information produced, and presents the results to the user as a single set of recommendations.

Figure 3.2 presents both approaches. The architecture presented in Figure 3.1 applies to both cases, but each case will have different organization of the information.

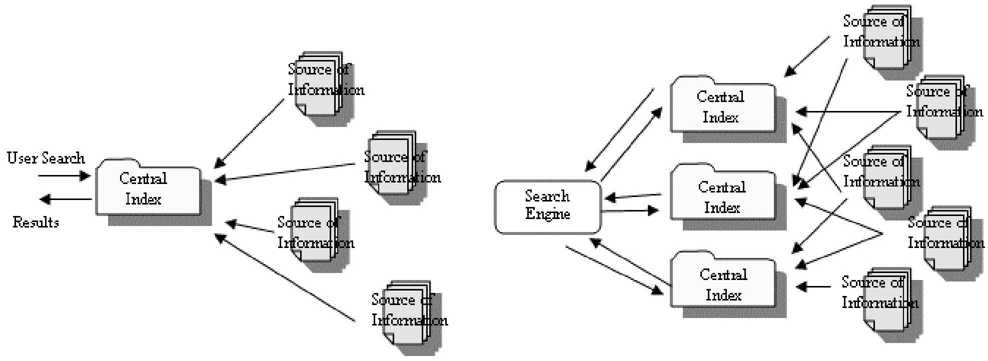


Figure 3.2: Example of Centralised (Left) and Multi-search (Right) Architectures

3.3. Adaptable Digital Libraries

Typically, personalisation in DL has taken an adaptable (user-driven) approach. Figure 3.3 presents the architecture of an adaptable DL, where 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 personalisation engine that adapts the contents and the presentation according to a user model.

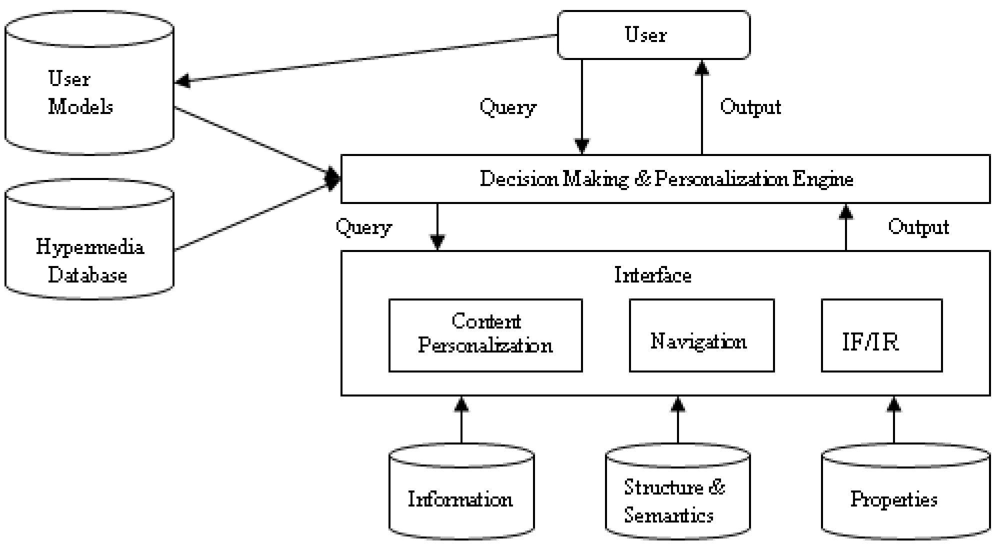


Figure 3.3: Generic Architecture of a Personalised Adaptable DL

The first developments for adaptable DLs are different implementations of MyLibrary. MyLibrary provides basic personalisation mechanisms regarding information retrieval and content personalisation (Cohen 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 Personalised Information Environment (PIE) (French and Viles, 1999; Jayawardana, et al., 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 adaptable DL services, which divide them into the three basic services provided by a DL: Adaptable Content, Adaptable Interface and Adaptable Online Searching.

1) Adaptable Content

Different content 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 stored in folders where each folder contains a set of links. The main tools for content personalisation are as follows (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.
- Shared Libraries: In this case, a library (catalogue) is owned by more than one user which can access and modify its content.
- Protection mechanisms: user name and encrypted passwords.

Different examples of the previous tools can be found in Virginia Commonwealth University (www.library.vcu.edu/mylibrary) and North Carolina State University, (my.lib.ncsu.edu). PADDLE (Hicks and Tochtermann, 1999) (Personal Adaptable Digital Library Environment) is another example of an architecture for adaptable DL that provides some of the tools previously described.

2) Adaptable Interface

DLs have a basic set of mechanisms to customise navigation. These mechanisms are common to any other individualised web pages. Typical services are customization of the interface by choosing among several colours, to order and rearrange libraries, folders, text colour and size, link colour, background colours, etc. The user creates a user profile that expresses his/her choices for an adaptable interface. A typical example of adaptable interface is MyYahoo! (Manber et al., 2000), which was also one of the first individualised 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. Adaptable interfaces have also extensively been used in e-commerce sites and e-banking.

3) Adaptable Online Searching

In terms of online searching, Information Filtering (IF) and Information Retrieval (IR) are two similar processes, which aim at providing a user with relevant information (Belkin and 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, IF is a passive process in which a user defines his/her information requirements and the information reaches the user once it has been filtered.

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 searching only in a catalogue or the web or combined, ordering the results by the levels of relevance, refining the search within the results obtained, etc. Typically those IR tools do not consider any user preferences. In the context of DLs, IF is used to implement population services (Di Giacomo et al., 2001) offered in order to find suitable journals and databases when creating a personal library. The literature already presents some adaptable IF and IR tools. CYCLADES (Candela and Straccia, 2003) is one of the tools aimed at providing an integrated environment for users who want to use electronic archives of documents, allowing some degree of personalisation in IR/IF processes by defining groups of users that share a common interest. The other example is Scirus (Scirus, 2004), a science-specific search engine, has an advanced mechanism for IF, offering the possibility of refining the results by filtering keywords.

3.4. Adaptive Dimensions of Personalised DLs

Although the adaptable tools described in the previous section are useful, they face the same limitations as any other adaptable service, as described in section 2.2. In order to solve these limitations, an automatic or adaptive approach should be used. The adaptive dimension of a personalised DL refers to the ability of a DL to automatically construct a user model without the direct intervention of the user.

Figure 3.4 presents the architecture of an adaptive DL. As showed in this Figure, 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 solves the problems that the adaptive approach has: (1) the user does not need to understand what personalisation is, (2) this approach makes it possible to create user models in an environment such as DL in which users are not willing to give feedback of their actions,

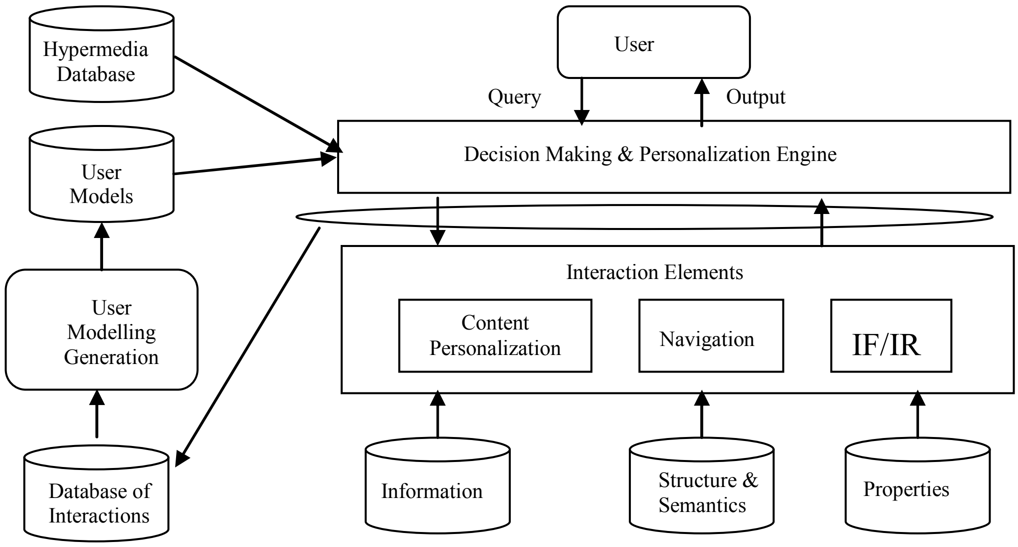


Figure 3.4: Generic Architecture of an Adaptive DL

(3) the DL is responsible for discovering user preferences and how these change over time, and (4) the adaptive approach makes it possible to deal with the amount of information thatDLs have. This adaptive approach still faces the same problems as any other adaptive approaches, as described in section 2.2.

The concept of Adaptive DL has been already sketched in some applications and implementations. Sections 3.4.1 through 3.4.3 give some examples of adaptive DL services for content personalisation, interface personalisation, and IR/IF personalisation.

3.4.1 Adaptive Content

Adaptive personalisation 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 make queries with the DL on the grounds of the documents stored in a personal catalogue.

3.4.2 Adaptive Interface

Adaptive interface tailors 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. An example of adaptive interface using the first approach is Fernandez et al. (1999), which provides adaptation of the interface at a very basic level depending on the operating system and the hardware. Costabile et al. (1999) and Semeraro et al. (2001) present an example of adaptive interface using the level of experience as stereotype.

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

Adaptive IF and IR systems personalise information access mainly according 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 et al. (2003), which presents a personalised 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 and Lawrence (1999) which presents a personalised IF system of scientific literature that constructs the user model by combining two methods: (1) constraint matching (keyword matching) and (2) related papers. In the second approach, the user indicates to the system papers that finds interesting and the system uses this information to suggest new papers. To some extent, some tools for creating repositories of DL include some kind of adaptive IF/IR system, for example Cornelis (2003) presents a study to personalise IR for Greenstone (Greenstone, 2006), or Fernandez et al. (1999), which presents an adaptive access to DL catalogues through Z39.50 servers, provides personalisation for IF and IR by learning user interests from previous queries.

In general, user modelling for IF and IR is a very active research field that has focused mainly in news systems. Widyantoro (1999) and Montaner et al. (2003) present an extensive review of user modelling for news filtering systems.

3.5. User Modelling for Adaptive DL Services

In order to automatically create user models for adaptive DL services, two questions need to be answered: (1) what information should a DL user model contain and (2) which techniques can be used to automatically capture that information. These questions are answered the following sections.

3.5.1 Dimensions of a DL User Model

One of the main problems that user modelling faces is the lack of any kind of standard of what a user model should contain. In general, the answer to this question is that the content of a user model is application dependent. Within the context of a personalised DL, there are eight potential dimensions that a user model should have:

- Device. Device captures the hardware used by the user to access the DL (such as PDA, laptop, Smartphone, etc.). The device affects the personalisation 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 Computer Science Department, etc.). This information can be used to infer the goals of that user.
- History. History captures users' past interaction with the system and can be used to personalise any kind of services using the assumption that a user is going to behave in the near 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 a destination or as a student writing a school report.
- Domain Expertise. Domain expertise indicates the knowledge of that particular user in the topics that are interesting to that user. Note that a user can have different levels of experience for different domains. This information can be used to re-rank and recommend new documents.
- Human Factors. Human factors are defined as any human characteristics. Common human factors that influence users' interaction with hypermedia systems include gender, system experience and cognitive styles. A more detailed description of human factors is given in the next section.

To implement a given DL service, not all the presented dimensions are needed. Table 3.1 presents which dimensions are relevant for each type of service: content personalisation, interface personalisation and IR/IF personalisation. Table 3.1 does not imply that all the

Table 3.1: Dimensions of a DL User Model and Their Relation with Each DL Service

	Content Personalisation	Interface Personalisation	IR/IF Personalisation
Human Factors	√	√	√
Domain Expertise	√		√
History	√		√
Device		√	
Context	√	√	√
Interests	√		√
Goal	√		√

relevant dimensions of a given type of service should be captured for a specific service of that type, but that the final user model may need to contain a subset of those dimensions.

3.5.2 Human Factors

Typically, for hypermedia applications, the relevant human factors considered have been gender (Ford and Miller, 1996), levels of experience (Mitchell et al. 2005), and cognitive styles (Chen and Macredie, 2004), because previous research indicates that these three factors have significant effects on users' interaction with web-based applications.

Gender is a typical human factor used to study individual characteristics in human-computer interaction (HCI). Different studies have concluded that female users have more problems when interacting with the web (Ford and Miller, 1996; Brosnan, 1998; Morahan-Martin, 1998). Large et al. (2002) investigated gender differences in collaborative web searching and their results revealed that males spent less time viewing pages than females. In addition, they found that the male group was more actively engaged in browsing than the female group, and that the male group explored more hypertext links per minute. Roy and Chi (2003) examined gender differences in searching the web for information by analyzing students' navigation styles. Their findings are in agreement with the results of Large et al. (2002), indicating that males and females possess different navigation styles while searching information on the Web. Males tended to navigate in a broader way than females. They also found that males tended to perform more page jumps per minute, which indicates that they navigate the information space in a nonlinear way. In general, females get lost more easily and find more difficult to locate information than males.

Level of experience is also a typical human factor used to study individual characteristics in human-computer interaction (HCI). It is a very interesting variable because it can highlight how the level of satisfaction of a user evolves over time (Mitchell at al., 2005). Some studies have already focussed on implementing specialised services according to different degrees of experience (Semeraro et al., 1999; Semeraro et al., 2001). Previous research has also highlighted the relevance of level of experience for web interaction and information seeking (Lazander et al., 2000; Palmquist and Kim, 2000). Torkzadeh and Van Dyke (2002) examined the change of users' Internet self-efficacy, in terms of belief in their own ability to succeed, before and after computer training, with the results of their study indicating that computer training significantly improved Internet self-efficacy. In other words, when the users develop from novice to experienced, their efficacy increases. In general individuals with higher levels of experience require less time to search information, needing fewer interactions and producing more correct responses.

Cognitive styles can be defined as an individual's preferred and habitual approach to organizing and representing information (Riding and Rayner, 1998). Cognitive style is a personality dimension, which influences the way individuals collect, analyse, evaluate, and interpret information (Harrison and Rainer, 1992). Previous studies indicated how individuals from different cognitive styles interact differently with web-based services (Ford and Chen, 2000). It can be used to adapt the DL to the way the user processes information. There are a variety of dimensions of cognitive styles, but among these dimensions, Field Dependence versus Field Independence and Imager versus Verbaliser have significant impacts on users' information processing.

Field Dependence/Field Independence reflects how well an individual is able to restructure information based on the use of salient cues and field arrangement (Weller et al., 1994). Their different characteristics are:

- *Field Dependence (FD)*: Field Dependence describes the degree to which a user's perception or comprehension of information is affected by the surrounding perceptual or contextual field (Witkin et al., 1981). Field Dependent individuals typically see the global picture, ignore the details, and approach a task more holistically. Field Dependent individuals are considered to have a more social orientation than Field Independent persons since they are more likely to make use of externally developed social frameworks. They tend to seek out external referents for processing and structuring their information. They are more readily influenced by the opinions of others, and are affected by the approval or disapproval of authority figures.
- *Field Independence (FI)*: Field Independent individuals tend to discern figures as being discrete from their background, to focus on details, and to be more serialistic in their approach to learning. These individuals tend to exhibit more individualistic behaviour since they are not in need of external referents to aide in the processing of information. They are better at processing impersonal abstract material, are not easily influenced by

others, and are not overly affected by the approval or disapproval of superiors (Witkin et al., 1981).

This dimension also defines intermediate individuals as the ones that have intermediate characteristics between FD and FI.

Recent studies have found that users' FD/FI parameter significantly influence their reaction to the user interface in terms of user control, multiple tools, and non-linear interaction (Chen and Macredie, 2002). With respect to user control, several studies have suggested (Chuang 1999; Chanlin 1998) that FI individuals could particularly get benefit from the control of media choice. Other studies (Marrison and Frick 1994) have suggested that FD users prefer to have auditory cues in the systems. Regarding multiple tools, Ford and Chen (2000) showed that FD individuals tend to build a global picture with the hierarchical map when interacting with web services, while Palmquist and Kim (2000) found that FD novices tend to follow links prescribed by a web page. Regarding non-linear interaction, Dufresne and Turcotte (1997) investigated the effect of cognitive style within a searching information environment. They found that FD students who used the system with non-linear structure spent more time completing the test than those who used the system with linear structure. FI individuals consulted the user guide for a longer period than FD individuals in the linear version, while FD individuals consulted it for longer in the non-linear version. In general, FD users tend to feel lost in hyperspace easily (Liu and Reed, 1995) and prefer a guided approach to the system (Wang et al., 2000).

All these results suggest that different cognitive style groups prefer different interface features and presentation formats provided by web-based applications and highlight the relevance of cognitive styles for personalisation. Therefore, there is a need to consider FD/FI parameter, which so far it has only played a minor role in personalisation in general and in DL in particular.

Another dimension of cognitive styles, Verbaliser vs. Imager, has been defined as the tendency for individuals to represent information being processed in the form of text or in the form of images (Riding and Cheema, 1991). Their different characteristics are:

- *Imagers (I):* Imagers tend to be internal and passive. In addition, imagers use diagrams more often than verbalisers to illustrate their ideas. Imagers perform better if the environment presents text and also pictorial material such as pictures, diagrams, charts, and graphs (Liu and Ginther, 1999).
- *Verbalisers (V):* Verbalisers tend to be external and stimulating. Verbaliser individuals perform better if the environment presents only information in the form of text.

This dimension also defines bimodal individuals as the ones that can represent and process information equally well both in the form of text and images. There are a variety of studies that highlight the relevance of the V/I dimension with how the users interact with a web-based application (Ford and Miller, 1996; Ford et al., 2001). These studies usually link imager individuals with poor retrieval success in information seeking environments.

Riding and Rayner (1998) combined both dimensions to create nine families. Each combination of FD/FI and V/I dimension is called a cognitive style (CS). The nine CS are: (1) Field Independent-Verbaliser, (2) Field Independent-Bimodal, (3) Field Independent-Imager, (4) Intermediate-Verbaliser, (5) Intermediate-Bimodal, (6) Intermediate-Imager, (7) Field Dependent-Verbaliser, (8) Field Dependent-Bimodal, and (9) Field Dependent-Imager. Each one of these nine types of cognitive styles combines the characteristics of each one of its dimensions. This approach has the advantage of clustering users into highly defined types, which allows identifying clear behaviour.

In general, results from the aforementioned studies suggest that gender differences, levels of experience, and cognitive styles have significant effects on users' behaviour on the web and their perception towards the use of the web. Thus, there is a need to consider these human factors in the process of user modelling so that personalised web-based applications can accommodate the needs of different types of user.

3.5.3 Construction of User Models for Adaptive DL Services

The automatic construction of a DL user model will be done by the automatic identification of each one of the dimensions of the DL user model presented in section 3.6.2. The following subsections give more detail about how to construct the user model in an adaptive way.

1) Modelling Human Factors: Cognitive Style and System Experience

The problem of identifying the system experience and the cognitive style 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 is contained in the interaction logs stored in the server. The problem can be solved using supervised learning techniques like decision trees, classification rules, 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 decision trees to classify into different stereotypes of interaction the set of users of a news information retrieval system.

2) Modelling Domain Expertise and History

Modelling the Domain Expertise dimension is intimately related with how each document of the DL is represented. Typically 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 an indication of the levels of difficulty of that document. Some standards for semantic web, like ARIADNE (Ariadne, 2004) and Dublin Core (Dublin, 2005), already contain fields that indicate the levels of difficulty. Using this information, the domain expertise of a user in a given topic would be given by a combination of the difficulty level of the documents of that topic stored in the user's personal library.

The History dimension of the model can be solved using association rules (Agrawa let al., 1993). Nanopoulos et al. (2001) models web user history using association rules and applies it to predict the requests of the next user. Sarukkai (2000) used Markov chains (Rabiner, 1986) to capture user historic behaviour 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.

3) Modelling User Interest

To model User Interest, it is necessary that the metadata that represents the document has a field describing the document's content, which is typically expressed in the form of keywords. ARIADNE and Dublin Core already contain such fields. In case the representation of the document does not include any description, the keywords can be found using a variety of document modelling techniques like TF-IDF (Term Frequency-Inverse Document Frequency). The combination of the keywords obtained from the documents of a user's personal library will indicate the set of user interests.

In order to implement personalised 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. To support this task, there is variety of algorithms to indicate similarity like k-nearest neighbour, clustering or neural networks. Paliouras et al. (1999) uses clustering to recommend interesting news to a given user in a personalised news system. Sheperd et al. (2002) use neural networks to construct and adaptive news filtering system according to user interests.

4) Modelling User Goals

Regarding the construction of a model to identify the goal of a user when interacting with a DL, the mechanism consists basically of a classification system that has a set of predefined categories (goals). In order to define these set of goals, some elements needed to be considered are: (1) the content and organisation 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 Departments). 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 characterise 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 (Cristianini and Shaw-Taylor, 2000). Other possible solution for modelling goals that has obtained very good results is Bayesian networks. Horvitz et al. (1998) present the construction of a goal prediction system using Bayesian networks that infers the objectives of a user within a software environment.

3.6 Conclusions

This chapter has presented a review of adaptive and adaptable approaches in DLs from which it can be concluded that the technology, especially the adaptive approach, is still in a premature phase. Although the best part of implementations has been done using adaptable approaches, the next level of DL services should be oriented towards the implementation of adaptive DLs based on data mining techniques that automatically construct DL user models. Up to now, the solutions of this approach are very limited.

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

The study has revealed two main areas for further research in adaptive DLs: (1) personalisation in DLs has been mainly focused on personalisation of content using user-guided techniques and personalisation of information retrieval and information filtering using both adaptive and adaptable techniques, but little work has been done in the field of personalised navigation; and (2) although human factors in general, and cognitive styles in particular, have been proved very relevant for determining user behaviour and user perception when interacting with hypermedia systems; few studies show how different human factors affect user interaction with DLs.

This thesis is going to combine these two lines and will target to examine which human factors are responsible for user behaviour and user perception in DLs in order to use such information to design a personalised interface for navigation. The following chapter presents the experiment and the study needed to accomplish these targets.

Chapter 4

Capturing User Behaviour and User Perception

4.1 Introduction

The previous chapter has highlighted the importance of designing personalised services for digital libraries. This chapter focuses on capturing and analysing user behaviour and perception of digital library users by conducting an empirical study. More specifically, the empirical study examines how different human factors can affect the perception and behaviour of DL users. The empirical study was conducted with Brunel Library Catalogue (BLC) because of the following reasons:

1) It uses a standard interface for digital libraries which will allow to some extent to generalise the results obtained.
2) The population that uses the library is very heterogeneous regarding human factors such as gender, levels of experience and cognitive styles.
3) If needed, there is a direct contact with the team that maintains BLC.

The chapter starts by describing experimental design, including participants, research instruments, task activities, and experiment procedure. Subsequently, both captured behaviour and perception data will be analysed in order to identify which human factors play a more relevant role for personalisation.

4.2 Experiment Design

This section describes the different characteristics of the experiment that were designed to capture users' navigation behaviour and perception data. The following subsections present the characteristics of the participants, the research instruments used, including BLC, the tasks designed and the data collection techniques used.

4.2.1 Participants

A total of 50 individuals participated in this study. Participants were students at Brunel University in the United Kingdom and they volunteered to take part in the study. A request was issued to students in lectures, and further by email, making clear the nature of the studies and their participation. All participants had the basic computing and Internet skills necessary to operate the Brunel digital library catalogue.

The classification of users according to the human factors presented in the previous chapter is: (1) considering Field Dependent/Field Independent (FD/FI) dimension of Cognitive Style (CS): 18 FI, 21 Intermediate and 11 FD, (2) considering the Verbaliser/Imager (V/I) dimension of CS: 18 I, 18 Bimodal and 14 V, (3) considering gender: 26 male and 24 female and (4) considering level of experience: 3 users have never used the BLC, 12 are novice, 17 are medium and 18 are expert.

4.2.2 Research Instruments

The research instruments used include: (1) Cognitive Style Analysis (Riding, 1991) to measure participants' cognitive styles, both the FD/FI and the V/I dimension (2) Brunel Digital Library catalogue, (3) Webquilt, a tool for capturing user interaction and storing a user questionnaire, and (4) a set of questionnaires for capturing the perception of the users.

1) Cognitive Style Analysis

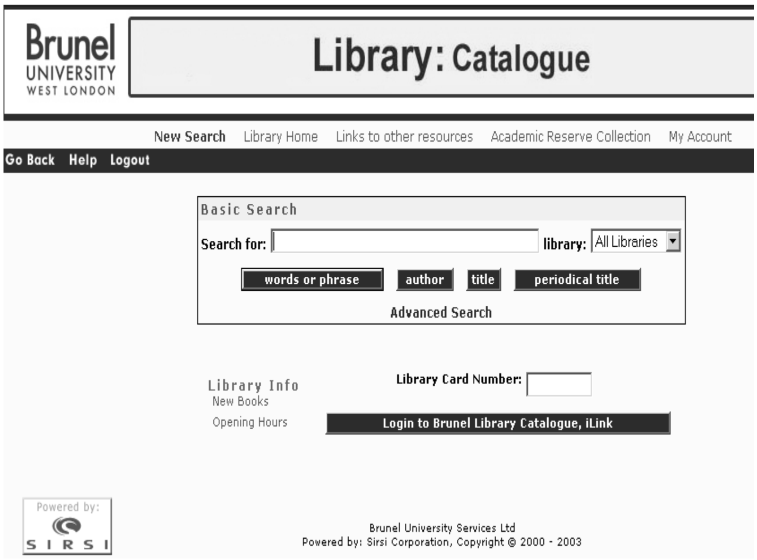
A number of instruments have been developed to measure Field Dependence/Field Independence (FD/FI) and Verbaliser/Imager (V/I) dimensions. Cognitive Styles Analysis (CSA) by Riding (1991) was chosen because it offers computerised test. The CSA test includes three sub-tests: (1) the individual is asked to classify items within classes using just textual representation, (2) the individual is required to judge if the pairs of complex geometrical figures presented are equal or different and (3) the individual is asked to indicate whether or not a simple geometrical shape, such as a square or a triangle, is contained in a complex geometrical figure (Riding and Grimley, 1999). There are 48 statements in total covering the three subtests.

These three sub-tests have different purposes. The second sub-test is a task requiring FD capacity, while the third sub-test requires the disembedding capacity associated with FI. This provides a big advantage over other methods that only measure one of the factors. Regarding V/I dimensions, it is assumed that Imagers respond more quickly to the appearance statements (second and third subtests), because the objects can be readily represented as mental pictures and the information for the comparison can be obtained directly and rapidly from these images. In the case of the conceptual category items (first subtest), it is assumed that Verbalisers have a shorter response time because the semantic conceptual category membership is verbally abstract in nature and cannot be represented in visual form.

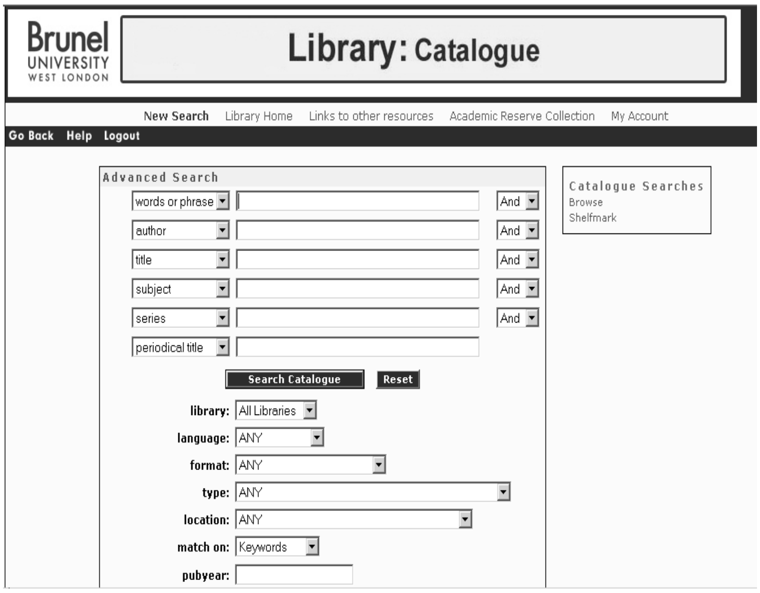
The CSA measures what the authors refer to as a FD/FI dimension (WA ratio) and the V/I dimension (VI ratio). Both ratios are real numbers that are used to identify each dimension. For the FD/DI dimension, Riding's (1991) recommendation is that WA scores below 1.03 denote Field Dependent individuals; scores of 1.36 and above denote Field Independent individuals; and scores between 1.03 and 1.35 are classified as Intermediate. For the V/I dimensions the recommendation is that VI ratios below 0.98 denote verbalisers; scores of 1.09 and above imagers; and scores between 0.98 and 1.09 bimodals.

2) Brunel Library Catalogue

Brunel Library Catalogue (BLC) is a typical digital library used to access the bibliographical resources of Brunel University. BLC has two main mechanisms that provide different strategies for finding information: (1) Basic Search (Figure 1a), which is the one presented by default by the system, and (2) Advanced Search (Figure 1b), which is accessed through the corresponding link presented in Figure 4.1(a). Basic Search allows the user to run a quick search of the library catalogue using a set of keywords and one of the following commands: "word or phrase", "author" "title" or "periodical title". The help link briefly describes what each link is supposed to do. Advanced Search, as presented in Figure 4.1(b), presents the user with a much broader way of searching for information. The user can give a value to each field (a generic work, author, title, subject, etc.), and combine these words using and/or Boolean operators.



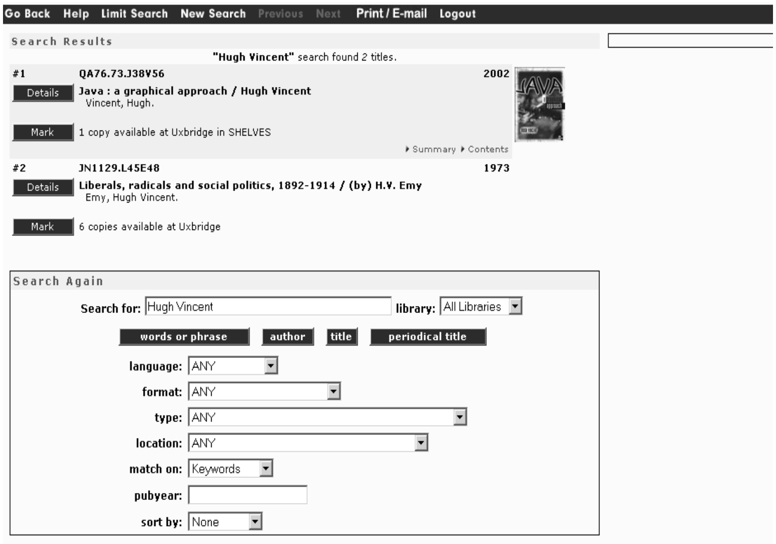
4.1(a)



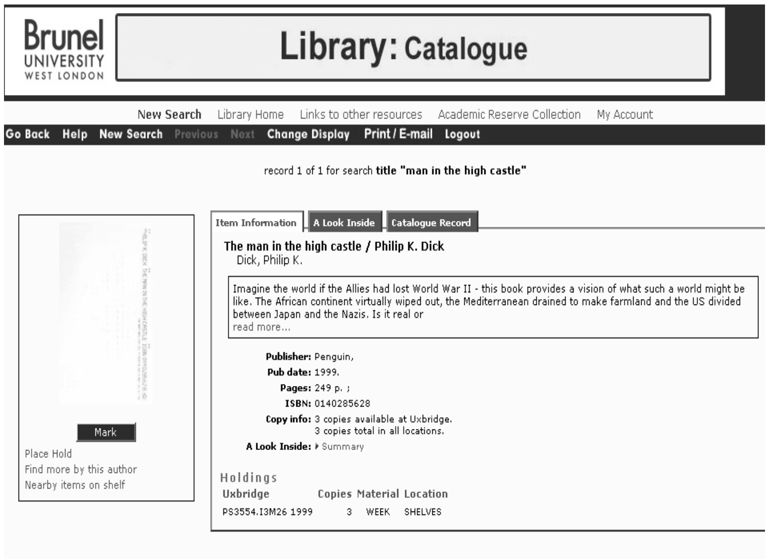
4.1(b)

Figure 4.1(a): Basic Search Interface of BLC and 4.1(b): Advanced Search Interface of BLC

Once a user submits a query to the system using the Basic Search or the Advanced Search, the system responds with the items found in the database. The results are presented using the alphabetical order of the titles found. An example of the interface presented is given in Figure 4.2(a). The system presents a set of buttons in the top part: “Go Back”, “Limit Search”, “New Search”, “Backward”, “Forward”, “Prefs” and “Exit”. The “Limit Search” option is a link to the bottom of the page where the search mechanism used (Basic Search or



4.2(a)



4.2(b)

Figure 4.2(a): Multiple Results Interface of BLC, and 4.2(b): Single Result Interface of BLC

Advanced Search) is presented with the terms used and a set of options for Search Limits (language, publication year, etc.).

The limit search is obtained by adding more words to the set of terms already introduced. The “New Search” option presents again the interface of Figure 1a. The “Backward/Forward” button allows the user to move up and down the items found. Once a user selects one item, the information and interface given is presented as in Figure 4.2(b).

3) WebQuilt

The WebQuilt Proxy Server (Hong et al., 2001) is a proxy system implemented using Java Servlet technology that unobtrusively gathers click stream data as users complete specified tasks. It is designed to conduct remote usability testing on a variety of Internet-enabled devices and provides a way to identify potential usability problems. Figure 4.3 presents the basic communication between each user and BLC through the proxy server. All the information captured is stored in the proxy server using an identification number for each user. This allows centralising all the information in the same place and at the same time being able to access the information of each user independently. Webquilt offers the possibility of adding a task box that can be used to indicate when a task has been finished. Once a user finishes each task, Web Quilt provides to each user a set of questions regarding the task. All these processes are done in a transparent way to the user. The use of a proxy server architecture makes it possible to capture all the interactions between users and BLC, which would otherwise be far more difficult as significant software changes would need to be implemented in BLC.

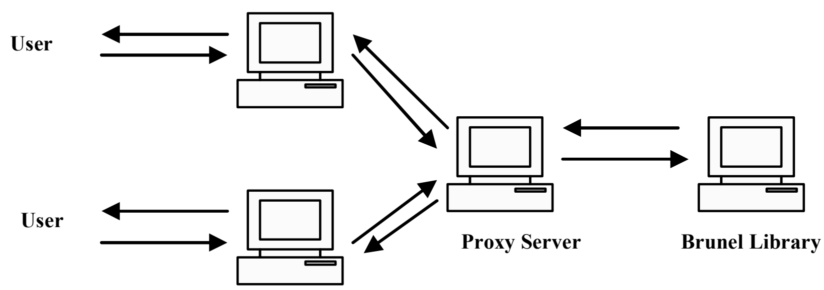


Figure 4.3: Typical Architecture of WebQuilt Working As a Proxy Server

WebQuilt organises its log files based on (a) the task being performed by the user, and (b) a user's ID. These two values can be passed in as query string variables when beginning a user session. For each page requested, WebQuilt stores all the information needed to trace the visit of that user. Table 4.1 details all the information stored for each request. Webquilt also stores the information sent from the user to BLC using the field URL, which includes for example the key words used for the search and the type of search used.

Table 4.1: Information Stored by Webquilt for Each Request

Field	Description
Time	The amount of time, in milliseconds, since the start of the user's session.
From	The transaction ID of the previous page the user came from.
To	The current transaction ID.
Parent	The transaction ID of the current page's frame parent, or -1 if none.
Code	The HTTP response code. 200 means OK, 404 means page not found.
Frame	The frame number of the current page (ie the Nth frame in the parent frameset). -1 if the page is not a frame.
Link	The link the user clicked to get to this page (ie the Nth link on the page). This counts both <A> and <AREA> tags. This value is -1 if the page was not reached through a link.
Method	The HTTP method used to retrieve the page (e.g. GET or POST).
URL	The current URL.
Query	The query data sent along with the page request, if any.

4) Perception Questionnaires

In order to capture users' perception when using BLC, three standardised questionnaires were used in this study: the Questionnaire for User Interface Satisfaction (QUIS) (Chin et al., 1988), Computer Usability Questionnaire (CSUQ) (Lewis, 1995), and After-Scenario Questionnaire (ASQ) (Lewis, 1995).

QUIS is a tool designed to assess users' subjective satisfaction with specific aspects of the human-computer interaction. Although QUIS is a very complete questionnaire, for the purpose of this study, a summarised QUIS test, which is available on-line (http://www.acm.org/perlman/question.cgi?form=QUIS), has been selected. In this version, the questionnaire is divided into five sections (Overall reaction to the software, Screen, System Information, Learning and System Capabilities) with a total of 27 questions. Each area measures the users' overall satisfaction with that facet of the interface, as well as the factors that make up that facet, using a [0-9] scale. An example of some of the questions that QUIS has is presented in Table 4.2.

Table 4.2: Examples of QUIS Questions

Question	Question	Area
1	The interface is: terrible (0) – wonderful (9)	Overall reaction to the software
2	The interface is: Difficult (0) – Easy (9)	Overall reaction to the software
4	The interface has: Inadequate Power (0) – Adequate Power (9)	Overall reaction to the software
6	The system is: Rigid (0) – Flexible (9)	Overall reaction to the software
17	Learning to operate the system is: Difficult (0) – Easy (9)	Learning
18	Exploring new features by trial an error is: Difficult (0) – Easy (9)	Learning
27	The system is designed for all level of users: Never (0) – Always (9)	System Capabilities

CSUQ (http://www.acm.org/perlman/question.cgi?form=CSUQ) was developed by IBM to evaluate the usability of a computer program, not necessarily a web service. It contains 19 questions, each being a statement that the user has to rate on a [1-7] scale ranging from "strongly disagree" to "strongly agree". An example of some of the questions that CSUQ contains is presented in Table 4.3.

Table 4.3: Examples of CSUQ Questions

Question #	Question
1	Overall, I am satisfied with how easy it is to use the system: 1 (Strongly disagree) – 7 (Strongly agree)
3	I can effectively complete my work using this system: 1 (Strongly disagree) – 7 (Strongly agree)
6	I feel comfortable using this system: 1 (Strongly disagree) – 7 (Strongly agree)
7	It was easy to learn to use this system: 1 (Strongly disagree) – 7 (Strongly agree)
16	The interface of this system is pleasant: 1 (Strongly disagree) – 7 (Strongly agree)
17	I like using the interface of this system: 1 (Strongly disagree) – 7 (Strongly agree)
18	This system has all the functions I expect it to have: 1 (Strongly disagree) – 7 (Strongly agree)
19	Overall, I am satisfied with this system: : 1 (Strongly disagree) – 7 (Strongly agree)

ASQ (http://www.acm.org/perlman/question.cgi?form=ASQ) is a CSUQ complementary test and is designed to be done once the user has finished all the tasks. As in CSUQ, the answers are in the range [1-7]. Appendix A presents the complete set of questions of the three questionnaires.

4.2.3 Task Design

Two main types of behaviour can be identified when users interact with digital libraries: browsing and searching (Bryan-Kinns et al., 2000). In this context, browsing is defined as the search of ill-defined information while searching is defined as the localisation of specific and well-defined information.

In order to capture these two types of behaviour, participants were asked to perform a set of seven practical tasks. The set of tasks was designed to involve all the functionalities that BLC provides to each user and the different behaviours (i.e., searching and browsing) that a user can show. Table 4.4 presents the tasks designed. The first question captures a searching behaviour, as it has a clear well-defined answer contained in the library catalogue. It is also designed to capture if the user uses the "Word or Phrase", "Author" or "Title" options (which are different ways of approaching the problem) or if an Advanced Search is used. When the Advanced Search is used, the proxy server will capture which elements are used (title, author, year), and if any search limit is introduced. The second task is a browsing question designed to test whether the user uses the "Subject" option of the Advance Search or prefers an

Table 4.4: Set of Tasks Designed and Their Type

No.	Task	Type
1	Find the Call Number of the book "The Man in the High Castle" by Philip Kendred Dick.	Search
2	Find the title of any book related to applications of fuzzy logic.	Browse
3	Find the number of books written by Aldous Huxley that are part of the TWICKENHAM Library	Search
4	Find a book about how to implement data mining with Java.	Browse
5	Find a Java book written by Hugh Vincent.	Search
6	Find a book about 20th century American Drama.	Browse
7	Find an IEEE journal on consumer electronics.	Search

approach using "*Title*" or "*Word or Phrase*". The rest of the tasks are designed to replicate some of the functionalities and/or behaviours in order to have more relevant data to work with.

4.2.4 Experimental Procedure

The experiment was conducted using the Brunel Library Catalogue (BLC) and comprised five different steps:

1) The CSA was used to classify participants' cognitive styles into FI, Intermediate or FD and Verbaliser, Bimodal, or Imager.
2) Participants were given a task sheet, which described the task activities that they needed to complete with the BLC (presented in Table 4.4). One participant carried out the experiment at a time.
3) Participants were observed while they were carrying out the seven tasks, and clarifications were given when requested. All interactions between the participants and the BLC were stored by Webquilt. The participants wrote the solution to each task after completing it. In case a participant did not find the solution to a question and wanted to skip it, he/she was allowed to do it.
4) Each participant answered the QUIS, CSUQ and ASQ questionnaires on-line.
5) Participants ended by answering the following questions: (a) gender, (b) level of experience in BLC (Never used the system, Novice, Medium or Expert), (c) their positions in the university: researcher/professor, graduate student, undergraduate student or others and (d) if they prefer the results of the search to be presented by alphabetical order or by relevance.

4.2.5 Data Collection and Summarisation

The interaction data collected from each user was centrally stored on the proxy server. This information was combined with the perception data obtained from the questionnaires and the human factors obtained from each participant to construct a 61-dimensional vectors that contained all the information from each user. The data captured for each participant is presented in Table 4.5.

Table 4.5: Dimensions of a BLC User Vector

No.	Variable	Information
1	BS	Number of times that Basic Search was used to solve a generic task.
2	AS	Number of times that Advance Search was used to solve a generic task.
3	SE	Number of times Word or Phrase was used to solve a generic task.
4	ATS	Number of times that Author, Title and Periodical were used to solve a generic task.
5	BF	Number of times that Backward/Forward was used to solve a generic task.
6	NS	Number of times that New Search was used to solve a generic task.
7	GB	Number of times that Go Back was used to solve a generic task.
8-14	T(i)	Time in microseconds needed to solve task i, i=1…7.
15-21	Trans(i)	Number of transactions needed to solve task i, i=1…7.
22-71		Answers to QUIS, CSUQ and ASQ.
72	CS	User cognitive style obtained using CSA test.
73	WA	*WA* ratio of the user provided by the CSA test.
74	VI	*VI* ratio of the user provided by the CSA test.
75	LE	Level of experience indicated by the User (Expert, Medium, Novice or Do not Usually use the system).
76	P	Position within the university.
77	G	Gender.
78	Pref	States if the user prefer the results ordered by alphabetical order or by relevance

For each user, the behavioural data captured for solving the seven tasks was summarised into seven dependant variables, variables 1 to 7 in Table 4.5. In order to obtain the value of each dimension, a compiler processed the information stored in the proxy server for each user solving the seven tasks, obtaining the total number of times that the user used a given functionality to solve the set of seven questions. After that, each variable of the dimensions, 1 to 7, was then normalised "to one task" by dividing each value by seven. The final value expresses the average number of times that the user would use each functionality to solve a generic task. Dimensions 8 to 14 indicate the amount of time that a user spent in solving each one of the tasks. Dimensions 15-21 indicate the number of transactions that the user needed to solve each task. A transaction is defined in this context as the number of pages visited by the user until the solution to the question was found. From a behavioural perspective, dimensions

1 to 7 captured the way in which each user interacts with the BLC to solve one generic task and 8-21 give indications of the amount of time and transactions needed.

As for the perception data, each vector contained the 49 answers given to the questionnaires in dimensions 22 to 71. Human factors were stored in dimensions 72 to 77 using six independent variables: users' cognitive style (CS), WA ratio, VI ratio, level of experience (LE), position within the university (P) and gender (G). Dimension 78 states if the user prefers the results presented by alphabetical order or by relevance.

4.3 Human Factors and User Behaviour

The goal of this section is to analyse the behaviour of digital library users when interacting with the library in order to identify which human factors are more relevant to personalise BLC interface. In this context, user behaviour is understood as how users have interacted with the functionalities offered by BLC to solve the questions presented in Table 4.4. The behaviour has been identified as a vector containing the dimensions 1 to 21 of Table 4.5.

Table 4.6 presents, from a global perspective, the number of times that each one of the functionalities offered by BLC has been used to solve a generic task and the standard deviation. As can be seen, a generic user interacts with the Basic Search (BS) option almost eight times more than with the Advanced Search (AS) option, while the Search Everything (SE) and the Author/Title/Periodical (ATS) buttons are used in the same proportion. Nevertheless, the high values of the standard deviation show that if individual human factors are considered, differences will arise. This information can be also read literally as: A generic user that solves a generic question with BLC, in average, uses the Basic Search (BS) interface 1.72 times and the Advance Search (AS) interface 0.31 times. Regarding the buttons, when using the Basic Search interface, Word or Phrase (SE) is used 0.63 times and the combination of Author, Title or Series (ATS) 0.77. The Backward/Forward (BF) button, the New Search (NS) button and the Go Back (GB) button are used 0.12, 0.15 and 0.17 times respectively. This interpretation of the information will be constant through the rest of this section, when analysing the same values for different human factors.

Table 4.6: Global Mean and Standard Deviation of BLC User Behaviour

	BS	*AS*	*SE*	*ATS*	*BF*	*NS*	*GB*
Mean	1.72	0.31	0.63	0.77	0.12	0.15	0.17
Std. Deviation	0.84	0.55	0.62	0.55	0.13	0.26	0.33

Table 4.7 shows the behaviour characteristics relating to the time and number of transactions needed to solve the seven tasks. The firs two columns, Time and Trans, show the average time needed to solve the seven questions by all users and the average number of transactions respectively. The next two columns, TimeSearch and TransSearch, show the same information but focussing only on the search questions (questions 1, 3, 5 and 7 of table 4.4), while the last two questions TimeBrowse and TransBrowse show the information for the browsing questions (questions 2, 4, and 6 of table 4.4). A generic user takes 63 seconds to solve a generic question, but there is a big difference depending on the type of questions, while search questions are solved in only 55 seconds, browse questions need 87 seconds, a 50% more time. This considerable time difference between browsing and searching tasks is probably led by their different definitions. Searching tasks contain the keywords needed to find information in the text that defines the task. For example, a searching task, "Find the Call Number of the book "The Man in the High Castle" by Philip Kendred Dick", already defines keywords in the title and the author. Nevertheless, browsing tasks are ill defined, so the user has to make more decisions about which keywords should be used. For example, a browsing task, "Find a book about 20th century American Drama", just gives some indications about how to select the keywords.

Table 4.7: Global Mean and Standard Deviation for the Time and Number of Transactions Needed

	Time	*Trans*	*TimeSearch*	*TransSeacrh*	*TimeBrowse*	*TransBrowse*
Mean	63656	5.12	55170	4.57	87452	5.22
Std. Deviation	14236	1.14	12682	0.8	22345	1.64

As showed in Table 4.6, the standard deviation is considerable which opens a door to find different behaviours for different human factors. The rest of the subsections analyse the interaction with BLC and the amount of time needed to solve the questions based on each human factor.

4.3.1 Field Dependence/Field Independence (FD/FI) Dimension

Tables 4.8 and 4.9 present user behaviour considering the FD/FI dimension as the individual human factor. Table 4.8 presents some important behavioural differences depending of the FD/FI dimension:

Table 4.8: Behaviour Characteristics Considering Each FD/FI Dimension (I)

	Field Dependent	*Intermediate*	*Field Independent*
BS	2.06	1.41	1.5
AS	0.0158	0.291	0.14
SE	0.34	0.64	0.52
ATS	0.84	0.65	0.87
BF	0.25	0.011	0.013
NS	0.15	0.095	0.22
GB	0.2	0.2	0.081

Table 4.9: Behaviour Characteristics Considering Each FD/FI Dimension (II)

	Field Dependent	*Intermediate*	*Field Independent*
AVTime	69732	49424	71813
AvTrans	6.14	4.5	5.03
AvTimeSearch	69932	42248.68	56392
AvTransSearch	5.37	4.54	4.68
AvTimeBrowse	69582	54805.55	83380
AvTransBrowse	6.7	4.6	5.29

- Field Dependent users use only the *Basic Search (BS)* option combined with *Author/Title/Periodical (ATS)* and to some extent *Word or Phrase (SE)*. In addition, the use of the *Backward/Forward (BF)* buttons is notable.
- Intermediate users use mainly *Basic Search (BS)*, although *Advance Search (AS)* also plays an important role in searching for information. *Word or Phrase (SE)* and *Author/Title/Periodical (ATS)* are used in the same proportion.
- Field Independent Users mainly use *Basic Search* (BS) and these users also rely more on *Author/Title/Periodical (ATS)* than on *Word or Phrase (SE)*. There is a relevant use of the *New Search (NS)* button. *Advance Search (AS)* is also used although the proportion is smaller when compared with *Basic Search (BS)*.

There are obvious differences between the FD and FI users, while FD do not use Basic Search, FI make a relevant use of this option. Also, while FD users use Backward/Forward and the Go Back button, these buttons are not used by FI users. This reinforces the literature regarding behaviour differences between FD and FI individuals when interacting with hypermedia systems. In general, FD users prefer a linear approach to exploring the system, which justifies the use of the Backward/Forward button, and the New Search button. Also, FD users are more passive, which may explain the lack of use of the Advance Search option. On the other hand, FI users would like actively to explore the systems by themselves, which may also explain their use of the Advance Search option.

There are also considerable differences between FI and FD users when considering the time and transactions needed to solve a task. As showed in Table 4.9, Intermediate users solve the question faster than FD and FI users (20 and 22 seconds less respectively) and need fewer transactions. Considering search and browse tasks separately, it can be observed that while there is not a big change for FD users in the amount of time needed, FI users solve much faster search questions than browse questions and need less transactions. Considering the time needed to solve the questions as an indication of the matching between BLC interface and user preferences, it can be concluded that the interface is best suited for Intermediate users, while FD and FI users meet more problems when interacting with BLC.

There is also a wide difference in the preferences of users, when considering the FD/FI dimension, regarding the way in which results have to be presented. While 72% of FI users prefer results presented by relevance, 78% of FD users prefer results presented by alphabetical order. Intermediate users have a tendency for alphabetical order (64% of Intermediate users prefer alphabetical order). When other human factors were considered, no clear tendency of the preference was identified.

The wide behaviour differences between FD/FI users make this dimension be as a very good candidate for personalisation.

4.3.2 Verbaliser/Imager (V/I) Dimension

Table 4.10 and Table 4.11 analyse user behaviour considering the Verbalise/Imager dimension of the cognitive style. Table 4.10 shows that there are no important differences between verbalisers, bimodals and imagers. They use mainly use Basic Search (BS), although Advance Search (AS) plays an important role in all of them, especially for Verbalisers that use it twice as much as Bimodals and Imagers. The rest of the buttons are used in the same way by these three types of users.

Table 4.10: Behaviour Characteristics Considering Each V/I Dimension (I)

	Verbaliser	*Bimodal*	*Imager*
BS	1.37	1.75	1.52
AS	0.49	0.27	0.2
SE	0.44	0.77	0.56
ATS	0.64	0.85	0.74
BF	0	0.1	0.02
NS	0.2	0.19	0.08
GB	0.15	0.18	0.16

Table 4.11: Behaviour Characteristics Considering Each V/I Dimension (II)

	Verbaliser	*Bimodal*	*Imager*
AvTime	56315	68154	66742
AvTrans	5	5.4	4.52
AvTimeSearch	45160	60137	57384
AvTransSearch	4.84	5.08	4.2
AvTimeBrowse	64682	74167	73760
AvTransBrowse	5.11	5.73	4.71

Although the interaction with BLC is the same for these three types of users, there is an important difference in the time needed. Table 4.11 shows that Verbalisers take on average 10 and 12 seconds less to solve a generic question. This trend is also true when considering searching and browsing tasks separately. For the three dimensions, browsing tasks take 20 seconds more to solve than searching tasks.

These results highlight the fact that the interface is better suited for Verbalisers probably, because the design of BLC lacks multimedia elements. The inclusion of more multimedia elements in the interface may help to reduce the difference in time needed to solve questions among them. There is a lack of differences showed in the behaviour of these three types of users so V/I values may not be a good dimension for personalisation.

4.3.3 Levels of Experience

Table 4.12 and Table 4.13 present the interaction with the interface and the time taken to solve the questions based on the levels of experience of the users. As showed in Table 4.12, in general, there is a reduction in the number of times that each function is used for users with

Table 4.12: Behaviour Characteristics Considering Each Level of Experience (I)

	Never use the System	*Novice*	*Medium*	*Expert*
BS	1.75	1.7	1.5	1.2
AS	0.28	0.28	0.34	0.12
SE	0.48	0.61	0.7	0.52
ATS	0.89	0.61	0.8	0.68
BF	0	0	0.09	0
NS	0.23	0.1	0.15	0.14
GB	0.017	0.04	0.25	0.11

Table 4.13: Behaviour of Each User According to Each Level of Experience (II)

	Never use the System	*Novice*	*Medium*	*Expert*
AvTime	72684	69353	67639	38303
AvTrans	5.12	4.26	4.44	4.07
AvTimeSearch	69992	59873	57398	241178
AvTransSearch	4.79	4.3	5.1	3.7
AvTimeBrowse	74703	79463	75320	48898
AvTransBrowse	5.37	4.2	5.7	4.3

higher levels of experience. It may be due to the fact that users with higher levels of experience need less interaction with the digital library to solve the questions.

This result can also be observed in Table 4.13, which shows that, on average, the higher the level of experience of the user the shorter the time needed to find the solution of a question. This also applies to the number of interactions needed to solve a task and to the analysis of the questions if searching and browsing questions are considered separately. Again there is a considerable difference between the time needed to solve searching tasks and browsing tasks. This is especially noticeable in expert users, which take half the amount of time to solve searching tasks than browsing tasks.

Apart from the aforementioned limited differences, users with different levels of experience behave similarly. Therefore, the level of experience is also not considered as a good candidate for personalisation.

4.3.4 Gender Differences

Table 4.14 and Table 4.15 present the behaviour of users based on gender differences. As can be seen in both tables, there are no relevant differences between females and males. The only noticeable differences are in the use of the Backward/Forward (BF) and Go Back (GB) buttons and on the average number of transactions needed to complete a task. In all cases, the value of each parameter is slightly higher for females.

Table 4.14: Behaviour Characteristics Considering Gender (I)

	Male	*Female*
BS	1.6	1.58
AS	0.29	0.29
SE	0.66	0.58
ATS	0.78	0.75
BF	0.00409	0.108
NS	0.13	0.18
GB	0.02	0.26

Table 4.15: Behaviour Characteristics Considering Gender (II)

	Male	*Female*
AvTime	61300	61765
AvTrans	4.8	5.24
AvTimeSearch	52577	52114
AvTransSearch	4.6	4.78
AvTimeBrowse	67843	69004
AvTransBrowse	4.9	5.58

Traditionally, the literature has reported that females take longer to find information in hyperspace and that they tend to get lost more easily than males (Large et al., 2002). This is not the case for BLC, and the reason for that will probably lie in the functionalities that offer a linear and guided access to the system. While males do not use the Backward/Forward (BF), or the Go Back (GB) button, females use them far more often, 50 times more for the Backward/Forward (BF) button and 13 times more for the Go Back (GB) button. The use of these buttons probably helps females to have a more structured access to information. Thus, the problem of getting lost can be avoided and they can find information at the same speed as males.

4.4 Human Factors and User Perception

This section analyses the perception of BLC users based on different human factors. The results are presented for a selected subgroup of QUIS and CSUQ questions, presented in Tables 4.2 and 4.3 respectively. The selection of these questions has been done simply considering the semantic relevance of the questions to this study.

Table 4.16 and Table 4.17 present the global mean and standard deviation for the selected questions. It has to be noted that while QUIS results are measured in a 0-9 scale, CSUQ questions are measured in a 1-7 scale. In general, when analysing QUIS results, it seems that users have a neutral opinion about the interface (5.23 in QUIS 1), that they think that BLC is an easy interface to deal with (6.63 in QUIS 2) and that it is easy to learn to operate (6.43 in QUIS 17). Users also find BLC interface a little bit rigid (4.87 in QUIS 6). CSUQ answers show that users have the perception that learning the system is simple (5.33 in CSUQ 7) and that they feel comfortable using BLC (4.93 in CSUQ 6). The standard deviation in both cases is quite high, 1.5 for CSUQ and 2 for QUIS, which highlights the variety of perception of BLC users. The rest of this section analyses user perception considering different human factors.

Table 4.16: Global Mean and Standard Deviation of Selected QUIS Questions

	QUIS 1	*QUIS 2*	*QUIS 4*	*QUIS 6*	*QUIS17*	*QUIS 18*	*QUIS 27*
Mean	5.23	6.63	5.13	4.87	6.43	5.67	5.27
Std. Deviation	2.300	1.903	2.417	2.300	2.161	2.591	2.518

Table 4.17: Global Mean and Standard Deviation for the Selected CSUQ Questions

	CSUQ 1	*CSUQ 3*	*CSUQ 6*	*CSUQ 7*	*CSUQ 16*	*CSUQ 17*	*CSUQ 18*	*CSUQ 19*
Mean	4.74	4.78	4.93	5.33	3.96	3.67	3.98	4.33
Std. Deviation	1.443	1.525	1.439	1.274	1.636	1.554	1.548	1.467

Table 4.18 and Table 4.19 present the perception considering FD/FI. Intermediate and FD users are more satisfied with the interface than FI users (QUIS 1), and also, by comparing the standard deviation, it can be seen that there is a more consistent opinion among Intermediate users (with std of 0.882), than among FD and FI users (with the std of 2). Intermediate users find that the system is more flexible than FD and FI (QUIS 6) and that it has an adequate power (QUIS 4). Regarding how simple is to use the system (QUIS 17), how difficult is to learn to use it (QUIS 18) and how comfortable a user feels using the interface (CSUQ 6), FD users find BLC easier to operate and to learn than FI and Intermediate users.

Table 4.18: Mean and Standard Deviation for Selected QUIS Questions and FD/FI

FD/FI		*QUIS 1*	*QUIS 2*	*QUIS 4*	*QUIS 6*	*QUIS 17*	*QUIS 18*	*QUIS 27*
Field Independent	Mean	5.00	6.63	4.94	4.75	6.31	5.50	5.25
	Std. Deviation	2.852	2.125	2.620	2.745	2.549	2.582	3.044
Intermediate	Mean	5.56	7.00	5.89	5.22	6.22	5.22	5.44
	Std. Deviation	.882	1.323	1.764	1.641	1.394	2.728	1.667
Field Dependent	Mean	5.40	6.00	4.40	4.60	7.20	7.00	5.00
	Std. Deviation	2.408	2.236	2.881	2.074	2.168	2.449	2.345

Table 4.19: Mean and Standard Deviation for Selected CSUQ Questions and FD/FI

FD/FI		*CSUQ 1*	*CSUQ 3*	*CSUQ 6*	*CSUQ 7*	*CSUQ 16*	*CSUQ 17*	*CSUQ 18*	*CSUQ 19*
Field Independent	Mean	4.76	4.67	4.76	4.95	3.95	3.57	3.52	4.00
	Std	1.480	1.683	1.411	1.203	1.596	1.399	1.504	1.612
Intermediate	Mean	4.71	4.75	4.96	5.62	3.96	3.54	4.38	4.54
	Std	1.488	1.567	1.488	1.209	1.546	1.641	1.583	1.318
Field Dependent	Mean	4.78	5.11	5.22	5.44	4.00	4.22	4.00	4.56
	Std	1.394	1.054	1.481	1.509	2.121	1.716	1.414	1.509

Globally, it can be concluded that while no dimension is really satisfied with the interface as it stands, Intermediate and FD users are more satisfied with the power and flexibility (CSUQ 19), while FI users desire more functionalities (CSUQ 18). Among those, extra functionalities needed to improve FI user's perception are mechanisms to learn to operate BLC and functionalities that add flexibility. The fact that FD users are more satisfied with the interface than FI users is probably motivated because the simplicity of BLC interface helps FD users to avoid the problem of feeling lost in hyperspace (Liu and Reed, 1995). These conclusions are in line with the results obtained from analysing behaviour of FD/FI users. Intermediate users are happy with the system as it stands because they can find solutions to the questions quite fast and without using a high number of transactions, as compared with FD and FI users that need more time and transactions.

Regarding the V/I dimension (Table 4.20 and Table 4.21), in general it can be said that Verbalisers are far more satisfied with the interface than Imagers (QUIS 1, 2, 4 and CSUQ 3, 6). Probably, the main reason is that the interface does not have any relevant presentation of the information in the form of images. One of the main differences between both dimensions is that Imagers see the system as far more rigid than Verbalisers (QUIS 6, 6.50 compared with 3.71). Again, this difference is probably produced because the interface of BLC is mainly text-based. These results and observations highlight the results already found in Table 4.10 and Table 4.11 in which Verbalisers solved the questions faster and with less number of interactions.

Table 4.20: Mean and Standard Deviation for Selected QUIS Questions and Verbaliser/Imager

Imager / Verbaliser		*QUIS 1*	*QUIS 2*	*QUIS 4*	*QUIS 6*	*QUIS 17*	*QUIS 18*	*QUIS 27*
Imager	Mean	4.57	5.79	4.21	3.71	6.14	4.79	4.64
	Std. Deviation	2.174	2.259	2.723	2.431	1.875	2.636	2.649
Bimodal	Mean	5.80	7.30	5.80	5.50	7.30	7.00	5.60
	Std. Deviation	2.860	1.337	2.201	1.958	1.337	1.333	2.836
Verbaliser	Mean	5.83	7.50	6.17	6.50	5.67	5.50	6.17
	Std. Deviation	1.169	.837	1.169	.837	3.502	3.450	1.329

Table 4.21: Mean and Standard Deviation for Selected CSUQ Questions and Verbaliser/Imager

Imager / Verbaliser		*CSUQ 1*	*CSUQ 3*	*CSUQ 6*	*CSUQ 7*	*CSUQ 16*	*CSUQ 17*	*CSUQ 18*	*CSUQ 19*
Imager	Mean	4.80	5.05	5.25	5.20	4.20	3.55	3.95	4.35
	Std. Deviation	1.542	1.638	1.410	1.473	1.795	1.791	1.791	1.496
Bimodal	Mean	4.78	4.65	4.87	5.30	3.61	3.65	4.00	4.30
	Std. Deviation	1.476	1.526	1.486	1.222	1.559	1.526	1.537	1.490
Verbaliser	Mean	4.55	4.55	4.45	5.64	4.27	3.91	4.00	4.36
	Std. Deviation	1.293	1.368	1.368	1.027	1.489	1.221	1.183	1.502

From a gender perspective (Table 4.22 and Table 4.23), female users felt it was harder to learn to operate and explore the system than male users (QUIS 17 and18). Nevertheless, although females are less satisfied with the interface than males, this fact did not translate into an increase in the amount of time needed to solve the questions as it is highlighted in Table 4.14 and Table 4.15. The reason why females are less satisfied is probably the lack of learning elements, instead of the functionalities offered by the interface. This is in accordance which other studies that show that females have more problems when interacting with the web

Table 4.22: Mean and Standard Deviation for Selected QUIS Questions and Gender

Gender		*QUIS 1*	*QUIS 2*	*QUIS 4*	*QUIS 6*	*QUIS 17*	*QUIS 18*	*QUIS 27*
Male	Mean	5.59	6.94	5.59	5.12	6.94	6.76	5.59
	Std. Deviation	2.425	1.560	2.425	1.965	1.391	1.480	2.399
Female	Mean	4.77	6.23	4.54	4.54	5.77	4.23	4.85
	Std. Deviation	2.127	2.279	2.367	2.727	2.803	3.059	2.703

Table 4.23: Mean and Standard Deviation for Selected CSUQ Questions and Gender

Gender		*CSUQ 1*	*CSUQ 3*	*CSUQ 6*	*CSUQ 7*	*CSUQ 16*	*CSUQ 17*	*CSUQ 18*	*CSUQ 19*
Male	Mean	5.28	5.38	5.31	5.76	4.07	3.97	4.24	4.72
	Std. Deviation	1.279	1.147	1.137	1.057	1.438	1.267	1.455	1.251
Female	Mean	4.12	4.08	4.48	4.84	3.84	3.32	3.68	3.88
	Std. Deviation	1.394	1.631	1.636	1.344	1.864	1.796	1.626	1.590

(Brosnan, 1998; Morahan-Martin, 1998), which somehow implies that females weight the learning functionalities higher than males.

Regarding the levels of experience, as showed in Table 4.24 and Table 4.25, the results indicated that the higher the level of experience of the user the lower the degree of satisfaction is (QUIS 1 and2 and specially CSUQ 19). The fact that expert users are able to solve the questions faster than any other levels of experience (Table 4.13) and that they need the minimum number of transactions (Table 4.12) is not enough for them to have a good opinion about the system. This is probably because expert users expect extra services not offered by existing BLC. This may be the same reason why novice users have a better opinion to the system, because novices are actually quite happy to avoid more complex services. Nevertheless, it is noticeable that novice users find the system extremely rigid (2.5 in QUIS 6), and that it has an inadequate power (3.0 in QUIS 7), while at the same time they are pretty satisfied with the interface as it stands (7.0 and 8.5 in QUIS 1 and 2) compared with medium and expert users that have milder opinions (around 5.0 in all cases). Again, the reason for this is probably the simplicity of the interface.

Table 4.24: Mean and Standard Deviation for Selected QUIS Questions and Level of Experience

Brunel Experience		*QUIS 1*	*QUIS 2*	*QUIS 4*	*QUIS 6*	*QUIS 17*	*QUIS 18*	*QUIS 27*
Never used the system	Mean	4.40	5.40	5.00	4.60	6.40	6.20	6.60
	Std. Deviation	1.949	2.302	2.121	1.949	2.074	1.924	.548
Novice	Mean	7.00	8.50	3.00	2.50	7.00	4.00	2.50
	Std. Deviation	.000	.707	5.657	4.950	2.828	2.828	4.950
Medium	Mean	5.38	6.88	5.75	5.31	6.88	5.88	5.50
	Std. Deviation	2.729	1.784	2.113	2.182	1.628	2.553	2.221
Expert	Mean	5.00	6.43	4.43	4.71	5.29	5.29	4.57
	Std. Deviation	1.633	1.813	2.370	2.138	3.094	3.302	3.047

Table 4.25: Mean and Standard Deviation for Selected CSUQ Questions and Level of Experience

Experience Brunel		*CSUQ 1*	*CSUQ 3*	*CSUQ 6*	*CSUQ 7*	*CSUQ 16*	*CSUQ 17*	*CSUQ 18*	*CSUQ 19*
Never used the system	Mean	4.63	4.88	4.88	5.25	3.50	3.63	3.88	4.50
	Std. Deviation	1.768	1.356	1.356	1.389	1.069	1.061	1.553	1.414
Novice	Mean	4.86	4.43	5.00	5.57	4.14	3.86	3.71	4.36
	Std. Deviation	1.069	1.902	1.732	1.272	1.952	1.773	1.113	1.215
Medium	Mean	4.77	4.80	4.97	5.27	4.10	3.70	4.13	4.33
	Std. Deviation	1.524	1.584	1.450	1.258	1.689	1.705	1.776	1.561
Expert	Mean	4.67	4.89	4.78	5.44	3.78	3.44	3.78	3.56
	Std. Deviation	1.323	1.364	1.481	1.424	1.787	1.424	1.093	1.509

As in the case of the behavioural analysis, the perception analysis show that the bigger differences in perception can be seen in the FD/FI dimension, and although other dimensions show different perception, these differences are not backed by different behaviour. These cases that show different perception but without different behaviour are probably influenced by external factors, such as more functionalities expected (in the case of the level of experience) and a tendency to feel lost in hyperspace (as when using gender as human factor).

4.5 Conclusions

The goal of this chapter was to analyse the behaviour and perception of BLC users in order to identify which human factor (FD/FI, V/I, level of experience or gender) is more relevant to personalise the interaction between users and the digital library. To achieve this goal, the chapter has first presented the design of the experiment needed to capture user behaviour. The experiment was based on: (1) a set of seven questions to capture the interaction with the interface, (2) a proxy architecture storing the interaction between users and the library catalogue and (3) a set of questionnaires capturing user perception. The interaction data captured was then processed to represent the behaviour and perception of each user.

Behavioural analysis showed that from all the human factors, the FD/FI dimension of the cognitive styles showed the biggest difference among various human factors. Users with different FD/FI values clearly showed different types of behaviour:

- Field Dependent Users tend to choose *Basic Search* option with *Word or Phrase (SE)* and *Author/Title/Series (ATS)*. The use of the Backward/Forward button is relevant.
- Intermediate Users mainly choose *Basic Search* although *Advance Search* also plays an important role in searching for information. *Word or Phrase* and *Author/Title/Series* are used in the same proportion.
- Field Independent Users use the *Advanced Search* option 25% of the times while the *Basic Search* option is used 75% of the times. They also rely more on *Author/Title/Series* than on *Word or Phrase*.

These differences give the basis to personalise BLC interface based on FD/FI dimension. The perception analysis confirmed some of the results found with the behavioural data. Focusing on the FD/FI dimension, it also showed that while Intermediate users are quite happy with the system as it stands, Field Dependent and Field Independent users are not really satisfied with the interface. This highlights the importance of personalising the interface for these two kinds of users.

Although the results presented in this chapter already gives indications on how to personalise BLC interface, the approach used in this chapter lacks the formality and robustness needed for taking such a decision. There is a need to use intelligent technologies, such as data mining, to enhance the robustness. The following chapter focuses on using data mining techniques to cluster users with similar behaviour and perception, identifying characteristics of the clusters and examining which human factors have been more relevant to form these clusters.

Chapter 5

The Role of Human Factors in Determining Behaviour and Perception of DL Users

5.1 Introduction

The previous chapter has highlighted the importance of FD/FI for determining the behaviour and perception of BLC users when compared with other human factors such as V/I, gender differences and levels of experience. This approach has divided the data according to the classes defined by a given human factor and then found statistical differences among them. Such an approach is typically used in HCI studies (Chen and Macredie, 2004; Yi and Hwang, 2003; Roy and Chi, 2003). The main problem of this approach is a lack of an integration description for user behaviour and user perception because it focuses on a link between a human factor and a single feature (e.g., time spent for completing tasks). In this chapter, a novel approach is proposed to overcome this problem. In this novel approach, unsupervised learning techniques are used to clusters of users that share similar behavior or perception, which are usually called stereotypes [Kobsa 2001]. Subsequently, statistical significances are examined to relevant human factors of each stereotype. This approach can provide obust evidence about which human factors are responsible for the perception and the behavior of users because it shows a direct relationship between a human factor and an integrated stereotype.

The chapter first starts by clustering the behavioural data by using the unsupervised learning techniques presented in Chapter 2 (k-means, hierarchical clustering and fuzzy clustering) and by identifying if there are significant relationships between the clusters (stereotypes) identified and any of the human factors considered. This chapter also presents to use robust clustering for user modelling as one of the novelties of this thesis and discusses the main advantages of using this approach. The same approach is then applied to the analysis of perceptional data. The conclusions of the chapter are built on the importance of human factors for personalisation and on the advantages of using data mining for identifying user preferences.

5.2 Relevance of Human Factors in User Behaviour

In this section k-means, hierarchical clustering and fuzzy clustering are going to be used to cluster BLC users in stereotypes according to their common behaviour. Once users are clustered, the relevance of each human factor in determining behaviour clusters will be studied.

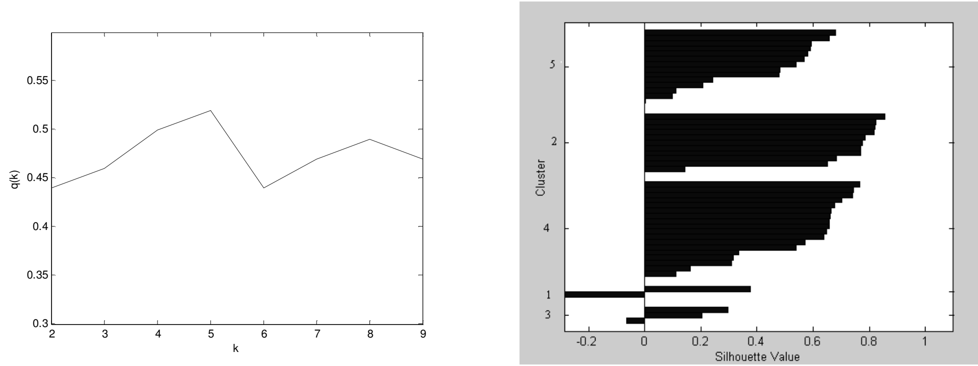
Matlab's Statistics Toolbox 5.1 (http://www.mathworks.com/products/statistics/) was used for k-means and hierarchical clustering, while Fuzzy Logic Toolbox 2.2 (http://www.mathworks.com/products/fuzzylogic/) was used for Fuzzy Clustering. The information used to represent the behavior of each user consists of all cases of a vector containing dimensions 1 through 7 of the elements presented in Table 4.5. The information used to cluster users does not contain any indication of any human factor.

5.2.1 Stereotyping with K-means

The inputs needed by K-means are the number k of clusters used to partition the original data, the concept of distance used to measure the distance between two elements, and, if desired, k cluster centres used to initialise each cluster.

The algorithm was executed for k=2,…, 9 without giving any initial value for the cluster centres and using Euclidean distance. To avoid that the solution given for a given k is a local minima, k-means was run for each value of k 100 times and the solution used was the one that minimised the objective function. In order to determine the optimal number of clusters, the technique presented in section 2.3.1 (subsection 5) was used, with N, the number of users in this case being 48. For each user i, an indication ϕ_i representing how similar the behaviour of that user was with users of the same cluster compared with the behaviour of users of all the other clusters was obtained.

Figure 5.1(a) presents the evolution of the quality of the partitions obtained for the values of k tested. As can be seen, the optimum partition is obtained with a value of k=5. Figure 5.1(b) presents a representation of the five clusters produced and ϕ_i for each user within that cluster. It seems that, from the five clusters, there are three easily distinguishable, with a high



(a) (b)

Figure 5.1(a): Evolution of the Quality of the Clusters, and (b): Representation of the Optimum Five Cluster Partition Found

number of users and a high ϕ_i value for its elements, indicating well defined behaviour of its users, and two clusters with a low number of users and lower ϕ_i values.

Table 5.1 presents the centre of each cluster, indicating the value of each dimension and also the total number of users included in the cluster. Those centres of clusters can be translated into the behaviour of the users. Cluster 1 and cluster 3 are not detailed because they do not contain a relevant number of users, and because they are not compact clusters based on ϕ_i. The behaviour of Cluster 2, Cluster 4, and Cluster 5 is described below.

- Cluster 2: Users use exclusively the *Basic Search* option in combination with *Author/Title/Periodical*.
- Cluster 4: Users use exclusively the *Basic Search* option and use *Word or Phrase* twice as much as the *Author/Title/Periodical*.
- Cluster 5: Users use *Basic Search* three times as often as *Advance Search* and *Word or Phrase* and *Author/Title/Series* in the same proportion.

Table 5.1: Cluster Centres Obtained by k-means

	Users	*BS*	*AS*	*SE*	*ATS*	*NS*	*GB*
Cluster 1	2	2.42	0.71	1.21	1.21	0.92	0.78
Cluster 2	11	1.61	0.02	0.09	1.44	0.09	0.23
Cluster 3	3	1.09	1.57	0.28	0.5	0	0.5
Cluster 4	19	1.6	0.04	0.89	0.4	0.07	0.1
Cluster 5	15	0.94	0.36	0.3	0.54	0.17	0.03

In order to identify the role of human factors on determining behaviour, ANAlysis Of Variance (ANOVA) was used to obtain the significance values (p-value) of gender, levels of experience, FD/FI and V/I in forming the clusters found. In general, only p-values of 0.05 or less are considered as an indication of relevance. The results showed that V/I, level of experience and gender differences did not have any relevance in determining behaviour clusters, with p=0.442, p=0.593 and p=0.238 respectively. However, FD/FI played a role in determining user's behaviour with a p-value of p=0.006.

Table 5.2 presents for each cluster and for each FD/FI value: (1) the percentage of users within each cluster that are of each FD/FI type (% in Cluster), and (2) the percentage of users of each FD/FI type included in each cluster (% in Sample). As can be seen, there is not a very strong relationship between clusters and FD/FI. Only C2 seems to capture FI users, with 63.3% of users being FI, representing 38.9% of all FI users, and C5 groups captured Intermediate users, with 78.6 % of users of the cluster being Intermediate, representing 50% of the total Intermediate users of the pool. From these initial results, it seems that there is not a direct relation between FD/FI and behavioural clusters. Nevertheless, k-means, as a clustering technique, has some biases, for example the concept of distance used, and these biases may affect the results obtained.

Table 5.2: Cognitive Styles of the Clusters Generated with k-means

CLUSTER	*Field Independent (FI)*		*Intermediate*		*Field Dependent (FD)*	
	% in cluster	*% in sample*	*% in cluster*	*% in sample*	*% in cluster*	*% in sample*
1	50.0%	5.6%	0%	0%	50.0%	12.5%
2	63.6%	38.9%	18.2%	9.1%	18.2%	25.0%
3	66.7%	11.1%	33.3%	4.5%	0%	0%
4	27.8%	27.8%	44.4%	36.4%	27.8%	62.5%
5	21.4%	16.7%	78.6%	50.0%	0%	0%

5.2.2 Stereotyping with Fuzzy Clustering (FC)

The only input needed by FC is the number of clusters in which the data is going to be classified. In order to estimate the number of clusters, subtractive clustering (Chiu, 1994) presented in section 2.3.1 (subsection 5) was used. Subtractive clustering was run for values of radii from 0.25 to 0.55 in order to determine the optimum number of clusters. Figure 5.2 presents the evolution of the number of clusters for each one of the radii values. As showed in Figure 5.2, values between [0.25, 0.45] produce a high number of clusters, specially if it is considered that the total number of users considered is 50. Nevertheless, in the range [0.45, 0.55], the value of clusters stabilises to five, which was selected as the number of clusters.

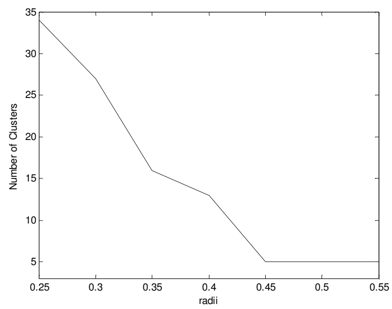


Figure 5.2: Evolution of the Number of Cluster Depending on the *Radii* Value

Fuzzy clustering assigns to each user a degree of inclusion in each cluster. In this study, each user has been included in the cluster that has the highest degree of truth. Table 5.3 presents the centre of each cluster, indicating the value of each dimension. Those centres of clusters can be translated into the behaviour of the users:

Table 5.3: Cluster Centres Obtained by Fuzzy Clustering

	Users	*BS*	*AS*	*SE*	*ATS*	*NS*	*GB*
Cluster 1	8	1.6	0.06	1.12	0.31	0.08	0.11
Cluster 2	10	1.59	0.06	0.07	1.4	0.09	0.21
Cluster 3	5	1.64	0.08	0.7	0.65	0.13	0.11
Cluster 4	15	1.26	0.23	0.44	0.64	0.19	0.12
Cluster 5	12	0.74	0.62	0.3	0.25	0.07	0.036

- Cluster 1: Users that exclusively use *Basic Search* in combination with *Word or Phrase* and *Author/Title/Periodical* occasionally.
- Cluster 2: Users that exclusively use *Basic Search* in combination with *Author/Title/Periodical*.
- Cluster 3: Users that exclusively use *Basic Search* in combination with *Word or Phrase* and *Author/Title/Periodical* in the same proportion.
- Cluster 4: Users that use mainly *Basic Search* and *Advance Search* occasionally and that use *Word or Phrase* and *Author/Title/Periodical* in the same proportion.
- Cluster 5: Users that use *Basic Search* and *Advance Search* in the same proportion in combination with *Word or Phrase* and *Author/Title/Periodical*.

The significance of FD/FI with the clusters identified by Fuzzy Clustering has a value of p=0.010. This shows that FD/FI can play a role in determining user behaviour. Again, V/I, Level of experience and gender do not have any relevance in determining behaviour clusters, with p=0.792, p=0.852 and p=0.177 respectively.

Table 5.4 shows that some clusters capture a relation with a FD/FI dimension: for example cluster 3, where 80% of the users are FD, representing 50% of all the users of this dimension, and cluster 4, which 76% of its members are Intermediate, representing the 45.5% of the total number of Intermediate users. As for the rest of the clusters, there is not a predominant type, and they group users with different FD/FI dimension.

Table 5.4: Cognitive Styles of the Cluster Generated with Fuzzy Clustering

Cluster	*Field Independent (FI)*		*Intermediate*		*Field Dependent (FD)*	
	% in cluster	*% in sample*	*% in cluster*	*% in sample*	*% in cluster*	*% in sample*
1	50.0%	22.2%	50.0%	18.2%		
2	60.0%	33.3%	30.0%	13.6%	10.0%	12.5%
3			20.0%	4.5%	80.0%	50.0%
4	28.6%	22.2%	71.4%	45.5%		
5	36.4%	22.2%	36.4%	18.2%	27.3%	37.5%

5.2.3 Stereotyping with Hierarchical Clustering

Hierarchical clustering has been used to identify users that share a common behaviour, using a Euclidean distance to construct clusters. Figure 5.3 presents the hierarchical clustering tree obtained, where the X-axis presents the users (some of them have been grouped to present a clear representation) and the Y axis illustrates the distance between the two objects being connected. The hierarchical tree has been cut at a height of 1.1, which creates five different clusters. This has been motivated by: (1) graphically from Figure 5.3, it can be inferred that the systems identifies five different groups and (2) the two previous techniques, for the same data set, have identified that the optimum number of clusters is five.

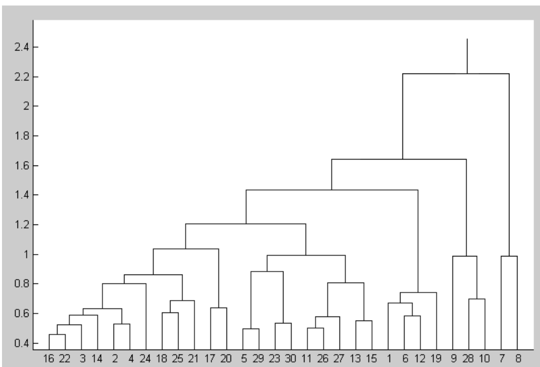


Figure 5.3: Schematic Representation of the Hierarchical Tree Constructed Using Behavioural Data

Table 5.5 presents the number of users included in each cluster and the centre of each cluster. In this case, each dimension of the cluster centre has been obtained as the mean of all the users included in that cluster. Cluster 4 and Cluster 5 are not detailed because they do not contain a sufficient number of users while the behaviour of Cluster 1, Cluster 2, and Cluster 3 are described below.

- Cluster 1: Users who exclusively use *Basic Search* in combination with *Word or Phrase* and *Author/Title/Periodical*.
- Cluster 2: Users who exclusively use *Basic Search* in combination with *Author/Title/Periodical*.
- Cluster 3: Users who use *Basic Search* and *Advanced Search* in the equal proportion in combination with *Word or Phrase* and *Author/Title/Periodical*.

Table 5.5: Cluster Centres Generated by Hierarchical Clustering

	Users	BS	AS	SE	ATS	NS	GB
Cluster 1	22	1.59	0.02	0.7	0.3	0.17	0.5
Cluster 2	14	1.6	0.06	0.07	1.44	0.09	0.03
Cluster 3	9	0.86	0.45	0.55	0.89	0.08	0.09
Cluster 4	3	1.26	0.04	0.32	1.16	0	0.1
Cluster 5	2	2.1	0.9	1.12	1.12	0.03	0.2

The significance of FD/FI with the clusters identified by Hierarchical Clustering has a value of p=0.005. The V/I dimension with p=0.523, Level of experience, with p=0.366, and gender, with p=0.645, do not play a relevant role. In Table 5.6, it can be identified a relation between some clusters with FD/FI, for example, cluster 3, where 75% of the users are FI, representing the 33% of all FI users, and cluster 2, with 69% of Intermediate users, representing 40% of all Intermediate users.

Table 5.6: Cognitive Styles of the Clusters Generated with Hierarchical Clustering

Cluster	*Field Independent (FI)*		*Intermediate*		*Field Dependent (FD)*	
	% in cluster	*% in sample*	*% in cluster*	*% in sample*	*% in cluster*	*% in sample*
1	27.3%	33.3%	40.9%	40.9%	31.8%	87.5%
2	30.8%	22.2%	69.2%	40.9%		
3	75.0%	33.3%	25.0%	9.1%		
4	66.7%	11.1%	33.3%	4.5%		
5			50.0%	4.5%	50.0%	12.5%

5.2.4 Comparative Analysis of the Stereotypes

The previous techniques roughly identify the same behaviour for the pool of users. Also, there are similarities between the behaviour discovered by each clustering technique and the set of behaviours expressed in section 4.3.1. However, while in that case the knowledge about the human factors was used, in this approach no knowledge of the different human differences has been used.

Nevertheless, although the different techniques identify similar set of behaviour, and they agree in the number of clusters created, they do not necessarily agree in the classification within a cluster of a given user. Different techniques have been developed to measure the similarity of two partitions, i.e. compare the level of agreement of two classifiers. These techniques can also be presented as a way of assessing the consistency of a partition. A method for comparing two data partitions is the Kappa metric (Altman 1997; Uebersax 1987; Valiquette 1994). This metric rates the agreement between the classification decisions made by two observers. The metric has a value in the range [-1, +1], where -1 indicates that there is no concordance between the observers, and +1 indicates that there is complete concordance. From a clustering perspective, a high kappa value indicates that the two arrangements are similar, while a low value indicates that there are dissimilar.

Table 5.7: Kappa Values for Each Technique Comparison When Using Behavioural Data

	k-means	*Hierarchical*	*Fuzzy Clustering*
K-means	1	0.764	0.509
Hierarchical	-	1	0.429
Fuzzy Clustering	-	-	1

Table 5.7 presents the *Kappa* value of each pair of clustering techniques used. As showed in this table, k-means and hierarchical clustering have good agreement strength of 0.764, while k-means with fuzzy clustering and hierarchical with fuzzy clustering only have moderate agreement strength. These results show that the partitions created are not very consistent. The main reasons for that are: (1) the bias of each technique has a direct impact in the classification results of the users, (2) the information that represents users contain a lot of noise that affects the clusters created.

The low consistency of the partitions may suggest that human factors in general, and FD/FI in particular, play only a minor role in determining the behaviour of a user, and thus in determining the stereotypes created by clustering techniques. This is because, if a given human factor, especially FD/FI as showed in our previous results, played a relevant role in determining behaviour, the different clustering techniques will have produced similar partitions, because all of them used the same information. Nevertheless, the partitions obtained are not very similar and this inconsistency may heavily be affected by the bias of each technique and the noise of the original data. That implies that, because the techniques do not filter any users, and because users can show behaviour that actually is not relevant in modelling user behaviour, the behaviour captured by each cluster is blurred by the addition of these ill-defined users.

In order to check the role that human factors play in determining user behaviour, a technique that counteracts the bias of the techniques and the noise of the data is needed. In this context, Robust Clustering (Swift et al., 2004), as presented in section 2.3.2, is a suitable tool because: (1) it eliminates the bias of the techniques, due to the fact that clusters are created only if all techniques agree, and (2) it filters users that do not have a well-defined behaviour, because one or more than one technique will not agree for dealing with these users.

5.2.5 Robust Clustering for User Stereotyping

The results obtained from using k-means, hierarchical and fuzzy clustering are used to apply Robust Clustering. The Agreement Matrix was of dimension *50x50*, with $C=3$. After applying the algorithm, eight clusters were obtained. A total of 11 users were filtered, 6 FD, 2 Intermediate and 3 FI, which represented 33% of all FD users, 9% of Intermediate users and 27% of FI users.

The significance of FD/FI with the clusters identified by Robust Clustering has a value of $p=0.000$. Level of experience had a significance of $p=0.656$, V/I of $p=0.231$ and gender of $p=0.317$. This shows that FD/FI actually determines the behaviour of a user, or, at least, plays a strong role.

Table 5.8 highlights a strong relationship between each FD/FI dimension and the clusters:

- 80% of all users of cluster 7 are FD, which represent 80% of all FD users. It is clear that this cluster groups users for which their behaviour is determined by a FD cognitive style.
- 85% of all users of cluster 1 are FI, which represent 50% of all FI users. This cluster groups users whose behaviour is determined by a FI cognitive style.
- 80% of users of cluster 6 are Intermediate, which represent 20% of all Intermediate users. In addition, 71% of the users of cluster 4 and cluster 3 are Intermediate, which represent, in both cases, 25% of all Intermediate users. These three clusters, which are determined by grouping Intermediate users, represent in total 70% of all Intermediate users. Unlike Field Independence and Field Dependence, who have well defined behaviour, Intermediate is defined as a cognitive style, which combines the characteristics of Field

Table 5.8: Cognitive Styles of the Clusters Generated With Robust Clustering

Cluster	*Field Independent (FI)*		*Intermediate*		*Field Dependent (FD)*	
	% in cluster	*% in sample*	*% in cluster*	*% in sample*	*% in cluster*	*% in sample*
1	85.7%	50.0%	14.3%	5.0%		
2			100.0%	10.0%		
3	28.6%	16.7%	71.4%	25.0%		
4	28.6%	16.7%	71.4%	25.0%		
5	50.0%	8.3%	50.0%	5.0%		
6	20.0%	8.3%	80.0%	20.0%		
7			20.0%	5.0%	80.0%	80.0%
8			50.0%	5.0%	50.0%	20.0%

Independence and Field Dependence, so it makes sense that Intermediate users are grouped in more than one cluster.

- As for clusters 2, 5 and 8, they only group two users, which are actually just one element of the Agreement List, so they do not represent relevant behaviour.

Once a technique that eliminates the bias of individual clustering techniques and that filters users that do not show a well defined behaviour has been used, the clusters obtained have a straightforward relation with FD/FI dimension. Although the information used to model the behaviour of individual users did not contain any indication about their FD/FI dimension, users with the same FD/FI dimension have been grouped in the same cluster. Such results imply that FD/FI dimension plays a key role in determining the behaviour of a user when interacting with BLC.

1) Analysis of the Behaviour of Each Cognitive Style

Table 5.9 presents the centre of the relevant clusters where each dimension of the cluster centre has been obtained as the mean of all the users included in that cluster. According to this table, the behaviour of each one of these clusters can be derived:

- Cluster 1 (which represents the behaviour of FI users): Users that exclusively use *Basic Search* in combination with *Word or Phrase* and *Author/Title/Periodical*.
- Cluster 3 (which represent the behaviour of Intermediate users): Users that use Basic *Search* and *Advance Search* in the same proportion, in combination with *Word or Phrase* and *Author/Title/Periodical*.
- Cluster 4 (which represent the behaviour of Intermediate users): Users that use Basic *Search* four times more often that *Advance Search* and *Word or Phrase* and *Author/Title/Periodical* in the same proportion.

Table 5.9: Cluster Centres Obtained with Robust Clustering

	Users	*BS*	*AS*	*SE*	*ATS*	*NS*	*GB*
Cluster 1	8	1.74	0	0.8	0.5	0.08	0
Cluster 3	8	0.65	0.54	0.31	0.17	0.08	0
Cluster 4	7	1.12	0.24	0.3	0.7	0.2	0.06
Cluster 6	5	1.57	0.02	1.19	0.35	0.07	0.19
Cluster 7	5	1.53	0.01	0.017	1.46	1.1	1.19

- Cluster 6 (which represent the behaviour of Intermediate users): Users that exclusively use *Basic Search* and *Word or Phrase* much more often than *Author/Title/Periodical*.
- Cluster 7 (which represent the behaviour of FD users): Users that exclusively use *Basic Search* in combination with *Author/Title/Periodical*.

In this case, because there are relationships between each cluster and each FD/FI dimension, it can also be said that the behaviour of each cluster has interactions with the characteristics that each FD/FI type has. In the approach presented in chapter 4, the behaviour for each FD/FI dimension was identified using the FD/FI information of each user. One of the limitations of that approach is that there is not a filtering of users that do not have clear defined behaviour from a FD/FI perspective, so they can pollute the final behaviour identified for each dimension. To compare the results showed in section 4.3.1 with those obtained in this section, some characteristics have been filtered:

- FD users: The behaviour identified by robust clustering does not use the *Word or Phrase* functionality, while in the behaviour identified in section 4.3.1 users use it.
- FI users: In the behaviour identified by robust clustering, there is no use of *Advance Search*, while in the behaviour identified in section 4.3.1 there is a very small use of that option.
- Intermediate: In both cases, because intermediate users use all the functionalities provided, both approaches identify the same behaviour.

These differences show the advantage of having a technique that filters users that do not have a strong behaviour of a given human factor in order to avoid the design of personalised interfaces where functionalities would not have been really used.

The behaviour results identified by robust clustering are also supported, to some extent, by other previous studies that examined the relationships between cognitive styles and user behaviour in a hypermedia environment. FI users like to explore the systems by themselves (Liu and Reed 1992) and jump from one point to another in hyperspace (Chen and Macredie, 2002), which explains that they use Basic Search in combination with Word or Phrase, which provides information without a predefined structure. This also explains that FI users do not use Advance Search because it provides a more organised environment. This also echoes the results obtained from Graff (2003) which shows that FI users do not favour using AND and OR operators for information searching purposes, which are the options provided by Advance Search. FD users prefer a more guided approach when accessing a hypermedia system (Liu and Reed 1992; Chen and Macredie, 2002) to avoid feeling lost in hyperspace (Wang et al., 2000). This justifies why they made use of the Go Back button more often than any other group and why they prefer to use Basic Search in combination with Author/Title/Periodical, which provides a clear access to information.

2) Analysis of the Users Filtered by Robust Clustering

One of the main advantages of using Robust Clustering for user modelling is the ability to filter users that do not show a clearly defined behaviour. This is especially important for a field such as user modelling where inherently the data available is noisy. It is interesting to analyse which users have been filtered in order to identify the characteristics of the users that actually define the behaviour identified in the clusters (thus also defining at the same time what is a user with an ill-defined behaviour).

In order to get a better idea of the users that have been filtered by Robust Clustering, their WA ratio was studied. As it was said previously, the concept of cognitive style (FD/Intermediate/FI) is actually constructed using the concept of WA ratio (a real number in the range of 0.6-3.0), in which WA scores below 1.03 denote FD individuals; scores of 1.36 and above denote FI individuals; and scores between 1.03 and 1.35 are classified as Intermediate. Such classification is given by Riding (1991), but other values for the classification of cognitive styles are also possible because the borders between cognitive styles are fuzzy.

Studying the users filtered, it was found that 90% of them were within a 0.1 margin of the cognitive styles borders, i.e. they are include within the ranges [0.93-1.13] which defines the FD-Intermediate border and [1.25-1.45] which defines the Intermediate-FI border. This explains our fist assumption that users that did not show a clear behaviour were filtered. From a cognitive style perspective, that is translated into users whose WA ratio is near the border of a cognitive style. While users that are far from the border will show a well defined behaviour, users near the border can have mixed properties. In other words, they do not have a well defined behaviour, which leads them to be filtered.

Table 5.10 presents the mean and standard deviation values for the WA ratios of the users included in cluster 7 (FI users), cluster 1 (FD user), and clusters 4, 5 and 6 combined (Intermediate Users). As showed in this table, those values are far way from the borders, i.e., the behaviour of each FD/FI dimension is defined by users that have a strong definition of that dimension, while users with weaker definitions are filtered. From the results provided by the

Table 5.10: WA Values of the Users Included in Cluster 7, 1 and 4+6+7 Obtained by Robust Clustering

	Cluster 7 (FD)	*Cluster 1 (FI)*	*Cluster 4+6+7 (Intermediate)*
Mean	0.88	1.79	1.17
std	0.06	0.19	0.09

clustering techniques, it seems that a user has an ill defined behaviour if the WA ratio is included in the ranges [0.93-1.13] or [1.25-1.45]. The conclusion is that users with WA values near borders tend to add noise to the behaviour characteristics so it is better to filter them before studying the characteristics of each FD/FI dimension.

5.3 Relevance of Human Factors in User Perception

This section examines the relationships between human factors and user perception. A similar approach used in the previous section was applied for this case. The data obtained from the satisfaction questionnaires was analysed by using clustering techniques and the significances of the corresponding human factors for each cluster were examined. The questionnaires provided for each user a vector containing the answer to the corresponding question in each dimension, making a total of 49 dimensions. It is convenient to identify a reduced form of representation of the perception vector in order to: (1) avoid the dimensionality problem of some clustering techniques and (2) better understand the results provided by the clustering techniques. In order to reduce the dimensionality of the perception vector, a common technique is the study of the significance of each dimension, filtering the dimensions that do not have a relevant significance.

In order to study the significance of each dimension, typically, ANalysis Of VAriance (ANOVA) models are implemented as a precursor to clustering. This approach has been typically used in bioinformatics (Wolfinger et al., 2001; Park et al., 2003; Liu et al., 2005) to identify the genes that are statistically more meaningful. This pre-processing of data usually reduces the dimension of the original data and improves performance of the ANOVA protected clustering method (Liu et al., 2005). In this case, the 49 questions are used to describe the perception of a user in order to identify which questions were more significant, and these questions were then used for further clustering. From the original 49 questions, 27 were identified to be significant, with an ANOVA model p-value of $p<0.05$ (Liu et al., 2005). Analysing the semantic of each question, in order to avoid similar questions and/or concepts, the set of questions was reduced to 17. The questions selected are, questions 1, 2, 4, 6, 9, 10, 17, 18 and 27 from QUIS and questions 2, 3, 6, 11, 13, 16, 17, 18 and 19 of the CSUQ.

These questions have been presented in Table 5.11 and Table 5.12. It can be seen that the questions selected by the significance have a high degree of coincidence with the questions manually selected in section 4.4.

Table 5.11: QUIS Questions Selected by the Relevance Filter

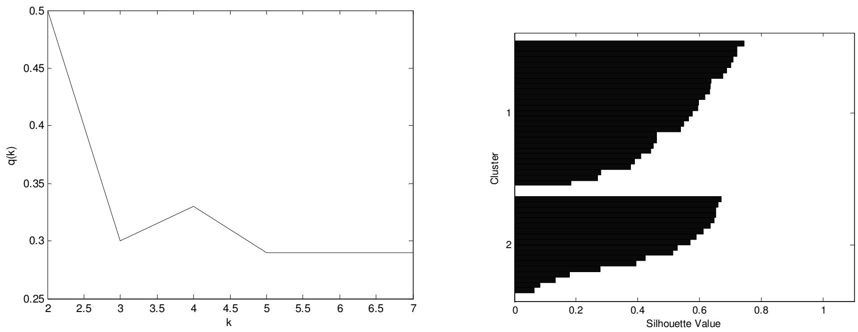
Question	Question	Area
1	The interface is: terrible (0) – wonderful (9)	Overall reaction to the software
2	The interface is: Difficult (0) – Easy (9)	Overall reaction to the software
4	The interface has: Inadequate Power (0) – Adequate Power (9)	Overall reaction to the software
6	The system is: Rigid (0) – Flexible (9)	Overall reaction to the software
9	Organization of Information: Confusing(0) – Very Clear (9)	Screen
10	Sequence of Screens: Confusing (0) – Very Clear (9)	Screen
17	Learning to operate the system is: Difficult (0) – Easy (9)	Learning
18	Exploring new features by trial an error is: Difficult (0) – Easy (9)	Learning
27	The system is designed for all level of users: Never (0) – Always (9)	System Capabilities

Table 5.12: CSUQ Questions Selected by the Relevance Filter

Question	Question
2	It was simple to use this system: 1 (Strongly disagree) – 7 (Strongly agree)
3	I can effectively complete my work using this system: 1 (Strongly disagree) – 7 (Strongly agree)
6	I feel comfortable using this system: 1 (Strongly disagree) – 7 (Strongly agree)
11	The information provided with this system is clear: 1 (Strongly disagree) – 7 (Strongly agree)
13	The information provided for the system is easy to understand: 1 (Strongly disagree) – 7 (Strongly agree)
16	The interface of this system is pleasant: 1 (Strongly disagree) – 7 (Strongly agree)
18	This system has all the functions I expect it to have: 1 (Strongly disagree) – 7 (Strongly agree)
19	Overall, I am satisfied with this system: 1 (Strongly disagree) – 7 (Strongly agree)

5.3.1 K-means, Hierarchical Clustering and Fuzzy Clustering for Identification of Perception

Firstly, K-means was applied to cluster perceptional data. The exact same mechanism presented in section 5.2.1 was used here. Considering this approach, the optimum number of clusters was 2, as can be seen in Figure 5.4(a). Figure 5.4(b) presents the optimum partition and the value of ϕ_i for each user included in that partition. Subsequently, the significant values of FD/FI, V/I, Levels of Experience and Gender Differences were obtained, being respectively *p=0.632, p=0.890*, *p=0.321 and p=0.114*. The results show that FD/FI and V/I do not play any role in determining the perception of a user and there is a slight indication that Levels of Experience and Gender Differences may be relevant.



(a) (b)

Figure 5.4(a): Evolution of the Quality of the Clusters, and (b): Representation of the Optimum Two Cluster Partition Found

Subsequently, Hierarchical clustering was applied to the same data, using a Euclidean distance to construct clusters. Figure 5.5 presents the hierarchical clustering tree obtained. The hierarchical tree has been cut at a height of 11.5, which creates two different clusters. This has been motivated by: (1) graphically from Figure 5.5, it can be inferred that the systems identifies two different groups and (2) k-means, for the same data set, has established that the optimum number of clusters is two. With this partition, the significant values of FD/FI, V/I, Levels of Experience and Gender Differences were *p=0.897, p=0.432*, *p=0.217* and *p=0.169* respectively. The two clusters obtained when applying Fuzzy Clustering produce very similar results, with significance values of *p=0.719, p=0.643*, *p=0.310* and *p=0.451* for FD/FI, V/I, Levels of Experience and Gender Differences.

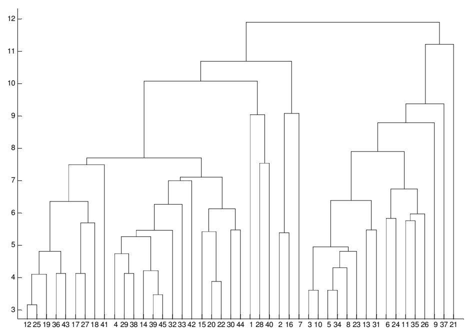


Figure 5.5: Schematic Representation of the Hierarchical Tree Constructed Using Perceptional Data

The previous results state that there is no relationship between the FD/FI and V/I of a user and his/her perception of the interface, and that there is a possible relationship between perception and gender differences/levels of experience. The filtering of noisy users and the elimination of the bias of each technique will help to highlight this possible relation. The kappa values, as presented in Table 5.13, indicate that there is a good agreement between all the techniques, which implies that the partitions created are consistent.

Table 5.13: Kappa Values for Each Technique Comparison When Using Perceptional Data

	k-means	*Hierarchical*	*Fuzzy Clustering*
K-means	1	0.906	0.509
Hierarchical	-	1	0.717
Fuzzy Clustering	-	-	1

5.3.2 Robust Clustering for the Identification of User Perception

The results obtained from k-means, hierarchical and fuzzy clustering were used for Robust Clustering. The Agreement Matrix was of dimension *50x50*, with *C=3*. After applying the algorithm, five clusters were obtained, with only three users filtered.

In order to identify the role that Gender Differences, Levels of Experience, FD/FI and V/I played in determining the robust clusters, their significance was obtained, having values of *p=0.730, p=0.651, p=0.009 and p=0.239*. This shows that Levels of Experience is the main human factor that determines the perception of a user, while at the same time it is confirmed that FD/FI and V/I do not play any role, while gender plays a minor role.

In order to further study how different levels of experience have been clustered, Table 5.14 presents the percentage of users within that cluster that are of that experience level (% in Cluster), and the total percentage of users of that experience level included in that cluster (% in Sample).

Table 5.14: Experience Level of the Users of Each Cluster Generated with Robust Clustering

Cluster	*I do not use catalogue*		*Novice*		*Medium*		*Expert*	
	% in cluster	*% in sample*	*% in cluster*	*% in sample*	*% in cluster*	*% in sample*	*% in cluster*	*% in sample*
1					100%	8.3%		
2	13.0%	50%	13.0%	60%	69.6%	66%	4.3%	10.0%
3	12.5%	33%	12.5%	40%	31.3%	20%	43%	70.0%
4					50.0%	4.2%	50%	10.0%
5	50.0%	16%					50%	10.0%

From the table, it can be observed that some clusters have a strong relationship with the Levels of User Experience:

- Cluster 2 groups users with a Medium experience level. In this cluster, 70% of users are of this type, representing 66% of all the users of that experience level.
- Cluster 3 groups users with an Advance experience level. In this cluster, 43% of users are of this level, grouping 70% of all the expert users of the pool.

Clusters 1, 4 and 5 are not considered because they only group a small number of users. After analysing these results, it has to be considered that: (1) the value of level of experience is given by each user, i.e. it has not been assigned by a test or by an expert, so it faces the subjectivities of each user, and (2) the pool of users was slightly unbalanced, with only three users saying that they do not usually use the catalogue.

It is possible to identify the relationships between the perception of users of each cluster and the levels of user experience. Table 5.15 presents the mean value of each dimension that characterises the perception of a user for each relevant cluster (clusters 2 and 3). The first table presents the values for QUIS (which has a range of [0-9]) and the second for CSUQ (which has a range of [1-7]). Using the semantic of the questionnaire, the following perceptions can be identified:

- Users of cluster 2 (medium experience users) think that the system is acceptable (6.29 in QUIS1) and that it is easy to use (7.25 in QUIS2), that has an adequate power (6.25 QUIS4) and that is flexible enough (6.18 QUIS6). In general, these users are moderately satisfied with BLC, and think that they have the tools they need to efficiently accomplish a task.
- Users of cluster 3 (expert users) think that the system is not up to what they would expect (2.94 in QUIS1), they have a neutral opinion regarding if it is difficult or easy (5.22 QUIS2), they think that BLC has inadequate power (3.05 in QUIS4) and that the system is rigid (2.5 in QUIS6). In general, the users of this cluster are dissatisfied with all the characteristics of BLC.

Table 5.15: Cluster Centres Obtained by Robust Clustering when Using Perception Data

	QUIS1	*QUIS2*	*QUIS4*	*QUIS6*	*QUIS9*	*QUIS10*	*QUIS17*	*QUIS18*	*QUIS27*
Cluster 2	6.29	7.25	6.25	6.18	6.55	5.59	7.25	7.00	6.44
Cluster 3	2.94	5.22	3.05	2.5	3.27	4.33	5.5	4.94	4.61

	CSUQ2	*CSUQ3*	*CSUQ6*	*CSUQ11*	*CSUQ13*	*CSUQ16*	*CSUQ18*	*CSUQ19*
Cluster 2	5.40	5.29	5.44	4.51	5.37	4.81	4.4	4.96
Cluster 3	4.00	3.77	3.83	2.88	3.66	2.44	2.88	3.22

From the results, it can also be observed that users do not have a well defined perception of BLC until they reach some degree of experience (Medium or Expert). The other two types of users, *I do not usually used system* and *novice*, do not have a coherent opinion of the interface and they do not group themselves in any clusters.

It seems that expert users are more dissatisfied with BLC than medium users. One of the main areas of dissatisfaction is the flexibility of the system (QUIS6), for which expert users consider to be very poor. This behaviour, to some extent counterintuitive, can be explained by the fact the expert users know BLC better, usually use it for more defined or high-level tasks (90% of expert users were from the group Lecturer/Researcher while 85% of medium users were Students/Graduate Students), and are probably more aware of other library catalogues and functionalities. Therefore, they may have a higher expectation to the BLC and are not satisfied with the functionality provided by existing BLC.

5.4 Conclusions

The preliminary results of Chapter 4 showed that when considering the FD/FI dimension of cognitive styled, users showed different behaviour. The goal of this chapter was to formally and robustly confirm these preliminary results in order to decide which human factor is suitable for personalising the interface of Brunel Library Catalogue.

The chapter first focuses on identifying human factors that have the highest relevance for determining user behaviour. The data analysis was conducted with k-means, hierarchical clustering and fuzzy clustering. These approaches revealed that FD/DI had some significance in determining user behaviour. Nevertheless, these approaches had a series of problems, mainly the bias of each technique and the lack of any kindd of filtering of users that did not show a well defined behaviour.

In order to avoid these problems, a solution was the use of robust clustering (RC). RC is based on constructing clusters based on the agreement of a set of clustering techniques. The main benefits are the elimination of the bias of the techniques and the filtering of users that do not have a well defined behaviour. The clusters obtained using RC and behavioural data showed the high relevance of FD/FI and a direct relation between each cluster and a FD/FI type. This result reinforced the idea that the FD/FI dimension plays a very relevant role in determining the behaviour of a user in BLC. From a personalisation perspective, this result also implied that one of the elements that a BLC user model should have is FD/FI because of the relevance in determining user behaviour.

The chapter also studied the characteristics of the users filtered by RC. The users that were filtered were those that were near the borders of FD/FI definitions. This conclusion suggested the idea that in order to study the behaviour of FD and FI users, it is recommended to filter the users that are near the borders of the different FD/FI types and focus on users that have a strong type of a FD/FI dimension.

The chapter presented the same approach to analyse perceptional data. The results indicated a strong significance of the levels of experience in determining the different perception clusters. The results showed that expert users had more negative opinions than intermediate users, mainly because expert users are more aware of other search functionalities excluded in BLC. On the other hand, the results also show that novice users and users that do not use the system do not have a homogeneous perception of the system (i.e. do not form a cluster). Due to this reason, the levels of experience may not be considered as a suitable human factor for the design of a personalised interface.

In summary, the results from Chapter 4 and this Chapter indicate that FD/FI plays a significant role in determining user behaviour and this parameter should be considered for personalisation. The following chapter will present the design of personalised interfaces for Brunel Library Catalogue based on FD/FI and will capture the behaviour and perception of the users with their new personalised interfaces in order to compare their evolution.

Chapter 6

User Satisfaction in Adaptable and Adaptive Digital Libraries

6.1 Introduction

The previous chapter showed that FD/FI plays a relevant role in determining the behaviour of a user in BLC. It also showed that FD, Intermediate and FI users develop different behaviour when interacting with BLC. These behavioural differences can be used to design a personalised interface. More specifically, the result of the design would be an adaptive interface for BLC based on the needs of each cognitive style.

In general, personalisation is supposed to increase user satisfaction because the system is tailored to the needs of each user. Nevertheless, it is not clear to which extent user satisfaction will be increased as a consequence of introducing adaptable and/or adaptive interfaces. The main goal of this chapter is to check to what extent the introduction of an adaptive interface designed using the behavioural results of Chapter 5 increases user satisfaction and if the introduction of the adaptive interface changes user behaviour and perception. The chapter will also focus on examining how an adaptable interface affects user satisfaction. In other words, the following questions will be examined in this chapter:

- whether adaptivity can increase user satisfaction and to which extent.
- whether adaptivity changes user behaviour and perception.
- whether adaptability can increase user satisfaction and to which extent.
- whether adaptability is really used by users.

These results will also allow answering one of the key questions that this thesis presents: to which extent is data mining able to automatically capture user preferences. The answer to this question can be obtained by comparing the increase in user satisfaction between adaptive and adaptable interfaces.

The chapter first presents the design of the adaptive cognitive interface and the adaptability added to the interface, followed by describing the experiment designed to capture the perception and behaviour of users when interacting with the adaptive interface and with the adaptable interface. Subsequently, it uses the data collect from the experiment to:

- Study the perception of the user with the adaptive interface and compare the results with the perception of the original BLC interface.
- Study the behaviour of users with the adaptive interface and compare the results with the behaviour with the original BLC interface.
- Study the perception with the adaptable interface and compare the results with the perception given for the adaptive interface.
- Study the impact that adaptability has in the interface.

6.2 Experiment Design for BLC Adaptable and Adaptive Interface

This section describes the experiment designed to capture users' behaviour and perception when using adaptive and adaptable interfaces based on BLC. The following subsections present the description related to the participants, the research instruments used, the tasks designed and the data collection techniques used.

6.2.1 Participants

A total of 50 individuals participated in this study. Participants were students at Brunel University and they volunteered to take part in the study. All participants had the basic computing and Internet skills necessary to operate BLC. From the pool of users that participated in this experiment, only 15% of them also participated in the pervious experiment described in Chapter 4. The time elapsed between the two experiments was 18 months.

The classification of users according to their FD/FI dimension is: 20 FI, 15 Intermediate and 15 FD. As for the rest of human factors: (1) considering the V/I dimension of CS: 16

imagers, 17 Bimodals and 17 Verbalisers; (2) considering gender: 28 males and 22 females and (3) considering levels of experience: 2 users have never used the BLC, 16 are novices, 16 are medium and 17 are experts.

6.2.2 Research Instruments

The research instruments used include: (1) Cognitive Style Analysis (Riding, 1991), (2) an adaptive and adaptable interface designed for BLC, (3) CSUQ and QUIS for capturing perception, and (4) Privoxy server for personalising the interface between each user and his/her FD/FI dimension. Among these instruments, CSA, Webquilt, CSUQ and QUIS have already been described for the previous experiment.

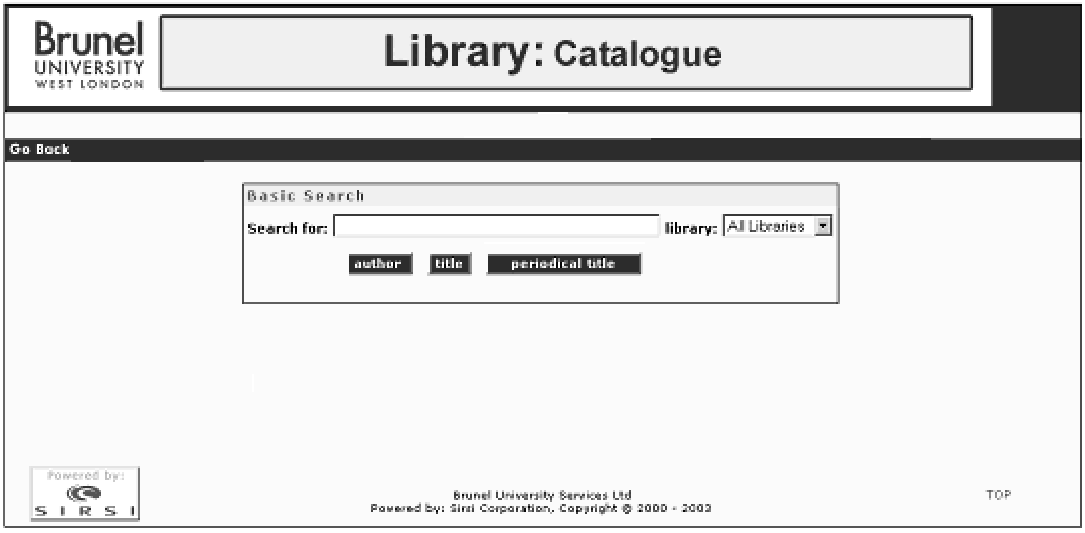
1) Design of an Adaptive Interface for BLC

The previous chapter has showed the relevance of FD/FI in determining user behaviour and proposed it as the relevant human factor to personalise BLC interface. The behavioural trends described in section 5.2.5 have been used to redesign BLC for each FD/FI dimension. The results presented in section 4.3.1 regarding the preferences of presenting information ordered by relevance or by alphabetical order have also been used. Figure 6.1 presents the three adaptive interfaces designed.

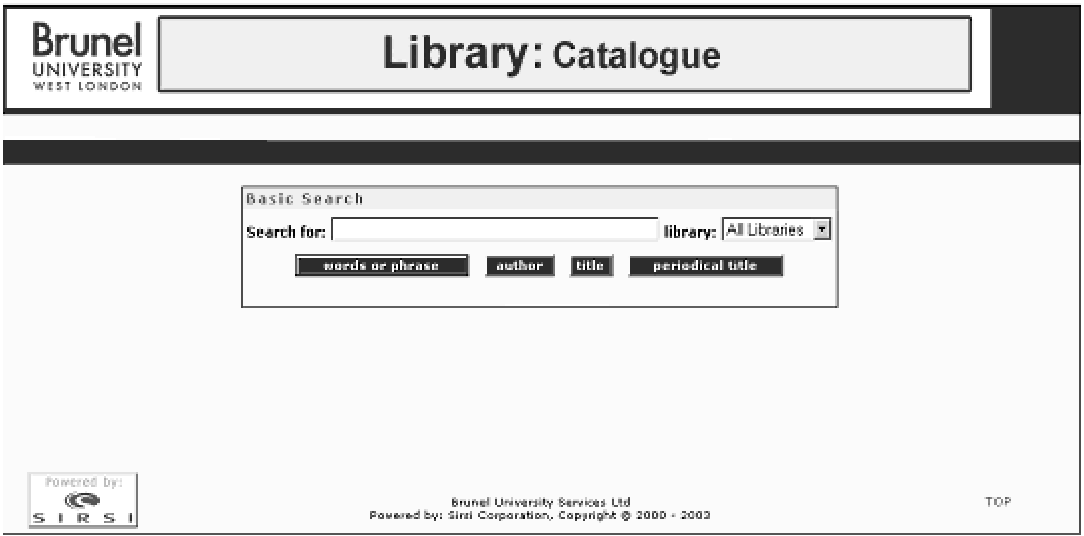
In general, when compared with the generic interface presented in Figure 4.1, all three adaptive interfaces have been simplified and there is no information regarding the library or links to other resources. The personalisation done in each case is described below:

- FD users have only access to *Basic Search* and the *Author/Title/Periodical* and *Go Back* buttons. The *Advanced Search* option is not offered. The results of the searching are presented in alphabetical order.
- FI users have only access to *Basic Search* and the *Word or Phrase* and *Author/Title/ Periodical* buttons. The *Advanced Search* option is not offered. The results of the searching are presented by relevance.
- Intermediate users have an interface which is basically as the original one but without the extra links. Users have access to *Basic Search* and the *Word or Phrase* and *Author/Title/Periodical* search functionalities. The *Advance Search* option is offered for Intermediate users. The results of the search are presented by alphabetical order.

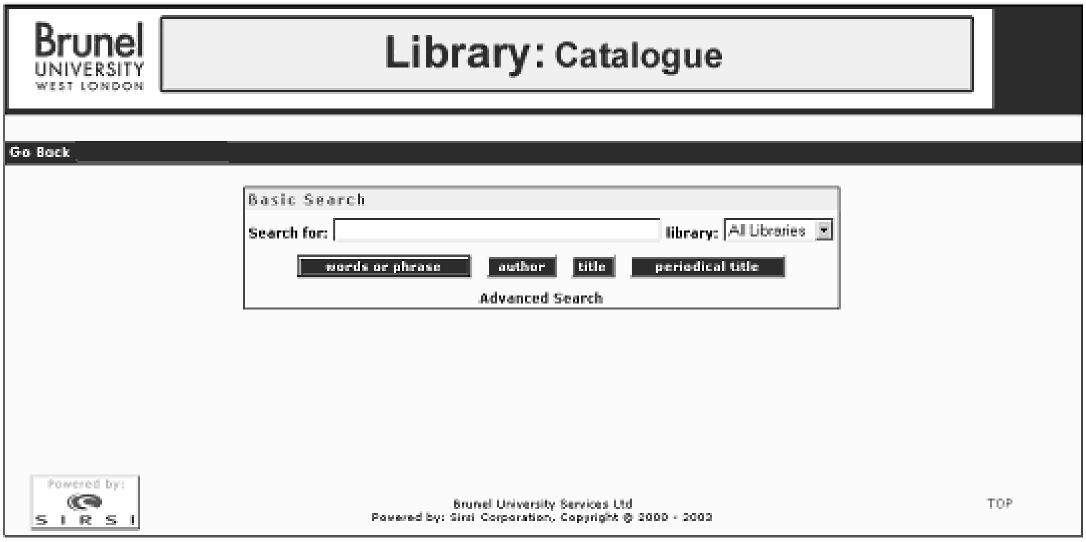
Each interface was generated automatically according to FD/FI dimension of each individual using Privoxy (Privoxy, 2006), a programmable proxy server that filtered the elements of the interface that were not relevant for each FD/FI dimension.



(a)



(b)



(c)

Figure 6.1 (a): Adaptive FD Interface, (b): Adaptive FI Interface and (c): Adaptive Intermediate Interface

2) Privoxy

Privoxy (Privoxy, 2006) is a web proxy with advanced filtering capabilities for filtering web page content, managing cookies and controlling access. Privoxy has a very flexible configuration and can be customised to suit individual needs. The flexibility of Privoxy lies within the configurable filter file, which contains the definitions of the elements that are to be filtered and the conditions that should be verified in order to filter them.

In our context, Privoxy is ideal to personalise the content of BLC in an unobtrusively way. The result is that the user receives an adaptive version of BLC interface based on his/her FD/FI dimension through Privoxy. In order to generate each adaptive interface, three filters were defined, one for each FD/FI dimension. The definition of the filters is done with regular expressions using a Perl-style language.

Each filter basically deleted the references to the links that were not to be used by each FD/FI dimension. Figure 6.2 presents the basic connection between a user and BLC through the proxy server used to personalise the interface. Privoxy also stores logs regarding the time of connection and the number of pages visited. Such information will be useful to study how behaviour has changed in an adaptive environment.

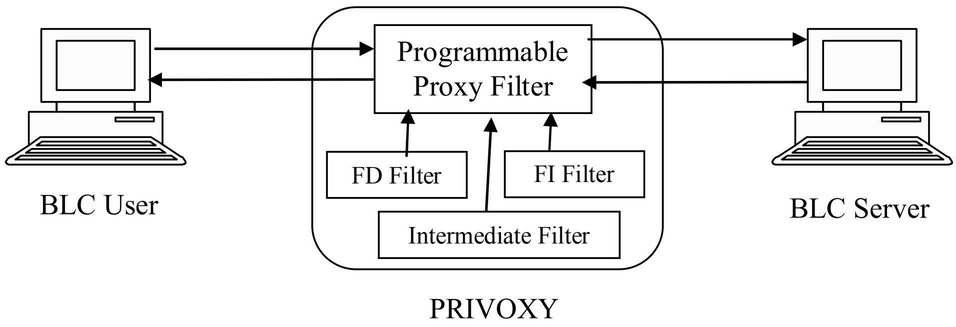


Figure 6.2: Architecture for the Implementation of Adaptive Interfaces

3) Design of an Adaptable Interface for BLC

Adaptability refers to the capability that an interface offers to a user to change the functionalities and general aspects of the interface. In this case, each user was offered the capability of changing the interface with a menu that provided access to all the functionalities of BLC interface. The menu was divided in two parts: Search Functionalities and Result Presentation. Search Functionalities included the following options: Search by Word or Phrase, Search by Author, Search by Title, Search by Periodical, Advance Search, Limit Search, Back Button, Backward/Forward button and Help button. Result presentation had two options: (1) by alphabetical order and (2) by relevance order.

When the user was presented with the adaptability menu, the options that were already being used (according to his/her FD/FI dimension) were already activated. Figure 6.3 presents an example of the adaptable interface presented to a FD user. The rest of the options were deactivated. Each option could be activated or deactivated as desired by the user. When the user accepts the changes made, a filter expressing the preferences of the user is generated and given to Privoxy to reflect the changes made and to allow the user to work with the new interface.

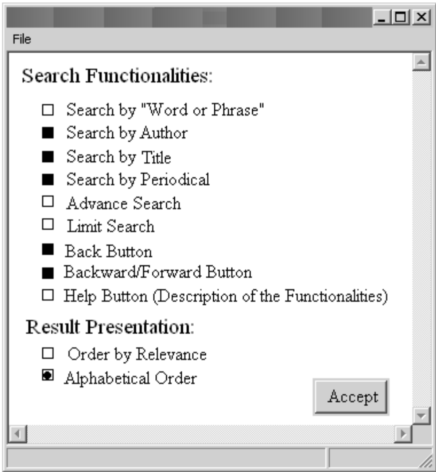


Figure 6.3: Adaptable Interface Presented to FD Users

6.2.3 Task Design

The design of the tasks has followed the same idea expressed in Section 4.2.2. The set of questions presented in the first experiment have also been used in this case but divided into two subsets: (1) the first one concerning Questions 1-4 of Table 4.4 and (2) the second set of questions concerning Questions 5-7 of Table 4.4. A new search question has been added to the second set in order to have also four questions: “Find the Call Number of the book Homage to Catalonia by G. Orwell”.

The reason of having two sets of questions is to solve each one of them with different interfaces, the first set of questions with the adaptive interface and the second set of questions with the adaptable interface.

6.2.4 Experimental Procedure

The experiment to evaluate user satisfaction in adaptive and adaptable interfaces is comprised of eight different steps:

(1) The CSA was used to classify participants' cognitive styles into Field Independent, Intermediate or Field Dependent.
(2) Participants started by answering the following questions: (a) gender, (b) levels of experience in BLC (Never used the system, Novice, Medium or Expert), and (c) their positions in the university: researcher/professor, graduate student, undergraduate student or other.
(3) Users had to complete the first set of tasks using the adaptive interface that matched their FD/FI dimension. Privoxy provided the adaptive interface needed and stored the interaction between each user and BLC.
(4) Users answered the QUIS and CSUQ questionnaires for the adaptive interface. The complete set of questions is presented in Appendix A.
(5) Users were presented with the adaptability menu and were given the possibility of changing their adaptive interface. The changes made by each user were stored and a specific filter was generated and given to Privoxy.
(6) Users solved the second set of the tasks with the adaptable interface. Privoxy provided the adaptable interface needed and stored the interaction between each user and BLC. The system also checked if the user actually used the changes made with the adaptability menu.
(7) Users answered again QUIS and CSUQ to measure user satisfaction with the adaptable interface. Although the semantics of the questions were as same as those used in step 4, the order of the questions was changed and the questions were reformulated with a different text to avoid that the answers given in step 4 could contaminate the results of this step. The complete set of modified questions for QUIS and CSUQ is presented in Appendix B.
(8) Users expressed if they preferred the first, second or had no preference of interface (i.e. if they preferred the adaptive or the adaptable interface or had no preference).

6.2.5 Data Collection and Summarisation

The information captured for each participant was used to construct an 89-dimensional vector that contained all the information from each user. The data captured for each participant is presented in Table 6.1. Dimensions 1 to 8 indicate the amount of time that a user spent in solving each one of the tasks. Dimensions 9-16 indicate the number of transactions that the user needed to solve each one of the tasks.

Table 6.1: Dimensions of a BLC User Vector for the Adaptive and Adaptable Interfaces

No.	Variable	Information
1-8	T(i)	Time in microseconds needed to solve task i, i=1…8.
9-16	Trans(i)	Number of transactions needed to solve task i, i=1…7.
17-48		Answers to QUIS and CSU for the adaptive interface.
49-80		Answers to QUIS and CSU for the adaptable interface.
81	CS	User cognitive style obtained using CSA test.
82	WA	WA ratio of the user provided by the CSA test.
83	VI	VI ratio of the user provided by the CSA test.
84	LE	Level of experience indicated by the User (Expert, Medium, Novice or Do not Usually use the system).
85	P	Position within the university.
86	G	Gender.
87	Pref	Preference of the adaptive interface, adaptable interface or no preference.
88	CH	Yes or No, depending if the user changed the original adaptive interface with the adaptability menu.
89	US	Yes or No, depending if the user used the changes made with the adaptability menu.

As for the perceptional data, dimensions 17 to 48 contain the answers to QUIS and CSUQ questions for the adaptive interface and dimensions 48 to 80 for the adaptable interface. Human factors were stored in dimensions 81 to 86: users' cognitive style (CS), WA ratio, VI ratio, level of experience (LE), position within the university (P) and gender (G). Dimension 87, variable Pref, indicates whether the user preferred the original adaptive interface or the adaptable interface. Dimension 88, variable CH, indicates if the user made changes to the adaptive interface using the adaptable menu and, dimension 89, variable US, if these changes were used for solving the second sets of tasks.

6.3 Perception of BLC Adaptive Interface

This section analyses the satisfaction of BLC users when interacting with the adaptive interface. The objective is to check if an adaptive interface increases user satisfaction and to which extent that satisfaction is increased. Table 4.16 and Table 4.17 presented the mean value of the answers given by the users to the original (non-personalised) BLC interface to the selected QUIS and CSUQ questions. Table 6.2 and Table 6.3 summarise the same information but for the three adaptive interfaces presented in Figure 6.1.

Table 6.2: QUIS Average User Answers for the Adaptive Interface

	QUIS1	*QUIS2*	*QUIS4*	*QUIS6*	*QUIS17*	*QUIS18*	*QUIS27*
Mean	5.84	6.58	5.58	5.36	6.91	6.05	5.67
Std	1.511	1.562	1.829	2.058	1.616	2.011	2.032

Table 6.3: CSUQ Average User Answers for the Adaptive Interface

	CSUQ1	*CSUQ3*	*CSUQ6*	*CSUQ7*	*CSUQ16*	*CSUQ17*	*CSUQ18*	*CSUQ19*
Mean	5.44	5.07	5.60	5.60	4.49	4.42	4.51	4.98
Std.	1.385	1.549	1.158	1.294	1.454	1.607	1.454	1.504

As showed in these tables, there is an increase in user satisfaction when going from the original BLC interface to the adaptive interface. This increase is mainly stated by: (1) CSUQ 19, which indicates how satisfied the users are with the system, has gone from 4.33 to 4.98, a 9.2% increase in satisfaction over the scale of CSUQ and by (2) QUIS 1, which measures if the interface is terrible or wonderful, has gone from 5.23 to 5.84, a 6.5% increase in the scale of QUIS. In general, all questions have increased in the area of 5% to 10%. A variety of questions demonstrate the impact of the adaptive interface:

- QUIS 6, which measures if the system is rigid or flexible, has gone up from 4.87 to 5.36, a 5% increase. It has to be considered that the adaptive interface actually offers fewer functionalities than the original BLC interface. Thus, it can be surprising to find out that users find the adaptive interface more flexible. The main reason may be that the interface has been adapted to the needs of their FD/FI dimension.
- CSUQ 1, which measures how easy it is to use the system, has gone from 4.74 to 5.44. It implies that there is a 10% increase. This appears to indicate that the adaptive interface is much simpler to use than the original interface.

To examine the standard deviation, there is another interesting improvement of the adaptive interface over the original BLC interface. In the original interface, the average standard deviation for the QUIS and CSUQ questionnaires was 2.203 and 1.58 respectively, while in the adaptive interface it has gone down to 1.8 and 1.4 respectively. These values imply that in the adaptive interface users are not only more satisfied but also agree more on their perception. These results show that an adaptive interface increases user satisfaction. Nevertheless, this increase is somehow limited to around 10% of the range of the questionnaires.

After assessing the global impact of introducing adaptivity, it is relevant to study the impact for each FD/FI dimension for which the personalisation was designed. Table 6.4 and Table 6.5 present the mean values for the adaptive interface for each FD/FI dimension. Table 4.18 and Table 4.19 presented the same mean values but for the original (non-personalised) interface.

Table 6.4: QUIS Mean Values for the Adaptive Interface for Each FD/FI Dimension

FD/FI Dimension		*QUIS1*	*QUIS2*	*QUIS4*	*QUIS6*	*QUIS17*	*QUIS18*	*QUIS27*
Field Dependent	Mean	5.83	6.58	5.42	4.17	7.08	6.58	6.42
	Std	1.267	1.730	1.782	2.082	1.782	2.314	2.193
Intermediate	Mean	5.25	7.83	5.58	4.83	6.58	5.42	5.83
	Std	1.712	1.642	1.832	2.517	1.782	1.975	2.125
Field Independent	Mean	6.21	7.05	5.68	6.00	7.00	6.11	5.74
	Std	1.475	1.268	1.945	1.374	1.453	1.823	1.759

Table 6.5: CSUQ Mean Value for the Adaptive Interface for Each FD/FI Dimension

FD/FI Dimension		*CSUQ1*	*CSUQ3*	*CSUQ6*	*CSUQ7*	*CSUQ16*	*CSUQ17*	*CSUQ18*	*CSUQ19*
Field Dependent	Mean	5.75	5.17	5.83	6.08	4.42	4.67	4.42	5.25
	Std	1.288	1.832	1.115	1.084	1.564	1.670	1.730	1.215
Intermediate	Mean	5.08	4.83	5.17	4.92	4.42	4.17	5.00	4.58
	Std	1.832	1.642	1.586	1.564	1.240	1.403	.603	1.676
Field Independent	Mean	5.47	5.32	5.74	5.74	4.58	4.42	4.26	5.05
	Std	1.124	1.336	.806	1.098	1.575	1.742	1.628	1.580

6.3.1 Field Dependent Users

FD users have clearly benefited from the introduction of an adaptive interface. Except for QUIS6 (rigid-flexible), the answers to all questions have improved in the region of 5% to 14%, with an average increase of 8%. The increase is especially noticeable in QUIS1 (satisfied with how easy is to use the system) which increases by 14%; QUIS13 (the system is defined for all level of users) which increases by 13% and CSUQ19 (I am satisfied with this system,) which increases by 10%. Only in QUIS6 (rigid-flexible), the satisfaction is reduced by 2% (-2%). Nevertheless, the value of only -2% shows that the difference in flexibility for users is not relevant.

6.3.2 Intermediate Users

When evaluating the increase of user satisfaction for Intermediate users, it has to be considered that the adaptive and the original BLC interfaces offer basically the same functionalities. The result is that, when comparing the answers for the adaptive interface with those for the original interface, there is not an appreciable variation in the perception, with all the questions having a variation between 2% and –1%. These variations can probably be considered noise because the different users evaluated the original and the adaptive interfaces.

6.3.3 Field Independent Users

FI users experience an increase in user satisfaction even bigger than FD users, ranging the increase between 4% and 15%, with an average increase of 11%. It is especially noticeable for QUIS1 (interface terrible-wonderful) which increases by 14%; QUIS6 (rigid-flexible) which increases by 15% and QUIS19 (I am satisfied with this system) which increases by 15%.

In summary, the aforementioned results indicated that the introduction of an adaptive cognitive interface in BLC increases user satisfaction of all FD/FI dimensions, on average by 10%.

6.4 Behaviour of BLC Adaptive Interface

This section analyses the behaviour of BLC users when interacting with the adaptive interface presented in section 6.2. The study is going to focus on the time needed to solve a question and the number of transactions needed. The goal of this study is to highlight if there is a link between a variation in user satisfaction and the reduction or increase of the time needed to solve a task.

Table 6.6 presents the average time needed and the number of transactions needed for all users. When compared with the results presented in Table 4.7 for the original interface, there is a reduction of 20% (12 seconds) in the time needed to solve a question and also a reduction of 20% in the number of transactions needed (1 transaction less). It is clear that there is a link between the increase in user satisfaction and the reduction on the amount of time needed to complete the tasks. In other words, if users are able to solve a task faster, their satisfaction will be increased. This reduction is also true when considering searching and browsing questions individually. The time needed to solve a search question is reduced by 25% (14 seconds), and by almost 30% (25 seconds) for a browsing question. Nevertheless,

Table 6.6: Global Mean and Standard Deviation for the Time and Number of Transactions Needed to Solve the Experimental Questions with the Adaptive Interface

	Time	*Trans*	*TimeSearch*	*TransSeacrh*	*TimeBrowse*	*TransBrowse*
Mean	51436	4.1	41267	3.97	61627	4.2
Std	11248	1.09	9345	0.69	13076	1.49

solving search questions are still much faster than browsing questions (41 seconds compared to 61 seconds). It is also important to highlight that the standard deviation of all the concepts studied is considerably reduced for the adaptive interface, implying that in general users show a more homogeneous behaviour.

Table 6.7 presents the average time and average number of transactions needed for FD, Intermediate and FI users. When compared with Table 4.9 that presented the same information but for the original BLC interface, it seems that there has been a reduction in the time and transactions needed for FD and FI users. However, for Intermediate users, although the average time is smaller in the adaptive case, the reduction is not relevant. This fact again highlights the relationships between user satisfaction and the time needed to solve a task. While users who do not have a reduction in the time or transactions do not have an increase in their satisfaction, those who are able to reduce the time needed to complete a task have an increase in their satisfaction. On a more detailed level, FD users experience a reduction of 24% (17 seconds) in the average time needed to solve a question. The same reduction applies if searching and browsing questions are considered separately. FI users experience an average reduction of 23% (16 seconds). There is also the same reduction if searching and browsing questions are considered separately.

Table 6.7: Behaviour of Each User According to Each FD/FI Dimension

	Field Dependent	*Intermediate*	*Field Independent*
AVTime	52439	47832	55067
AvTrans	4.1	4.4	4
AvTimeSearch	53482	41432	39652
AvTransSearch	4	4.5	3.8
AvTimeBrowse	51389	52657	69856
AvTransBrowse	4.2	4.3	4.3

The same behavioural characteristics that were observed in the original BLC interface can also be observed in the adaptive interface: (1) Intermediate users solve the question faster than FD and FI users, (2) FD users can solve browsing and searching questions using the same amount of time and (3) FI users spend less time in solving searching questions than they do in browsing questions. Basically, the behaviour that users show in the original interface is present in the adaptive interface but with a reduction in the amount of time needed to complete the tasks. This implies that adaptivity does not change the behaviour of users, but it reinforces their tendencies.

6.5 Comparison of Perception of BLC Adaptive and Adaptable Interfaces

This section studies how the introduction of adaptability affects user satisfaction compared with a system that only offers adaptivity. The results presented in this section are based on the answers given by users after solving the second set of tasks with the adaptable interface. The purpose is to identify which part of the increase in user satisfaction is caused by the adaptive interface and which one is by the adaptable interface. In addition, it will answer to which extent data mining techniques are able to identify user preferences.

Table 6.8 and Table 6.9 present the mean value of the answers for the QUIS and CSUQ questions after using the adaptable interface. After conducing the comparison between Tables 6.2 and 6.3 (which present the results for the adaptive interface), it seems that there is no clear variation. All questions in the adaptable interface are within a range of +2% and –2% of the adaptive interface. It implies that there is not a change in the satisfaction of the users between these two types of interfaces.

Table 6.8: QUIS Average User Answers for the Adaptive and Adaptable Interface

	QUIS1	*QUIS2*	*QUIS4*	*QUIS6*	*QUIS17*	*QUIS18*	*QUIS27*
Mean	5.95	6.59	5.52	5.27	6.94	6.05	5.91
Std	1.759	1.533	2.130	2.261	1.965	2.171	2.021

Table 6.9: CSUQ Average User Answers for the Adaptive and Adaptable Interface

	CSUQ1	*CSUQ3*	*CSUQ6*	*CSUQ7*	*CSUQ16*	*CSUQ17*	*CSUQ18*	*CSUQ19*
Mean	5.55	5.86	5.42	5.70	4.44	4.44	4.62	4.95
Std	1.307	1.457	1.139	1.389	1.563	1.593	1.484	1.344

Table 6.10 and Table 6.11 present the mean values of answers given to QUIS and CSUQ questions for each FD/FI dimension for the adaptable interface.

Table 6.10: QUIS Average Answers for the Adaptive and Adaptable Interface for Each FD/FI Dimension

FD/FI Dimension		*QUIS1*	*QUIS2*	*QUIS4*	*QUIS6*	*QUIS17*	*QUIS18*	*QUIS27*
Field Dependent	Mean	6.08	7.08	5.58	4.25	7.42	6.83	6.75
	Std. Deviation	1.311	1.505	2.314	2.454	1.443	2.250	1.485
Intermediate	Mean	5.58	7.67	5.42	5.08	6.17	5.67	5.67
	Std. Deviation	1.621	1.073	1.443	2.234	2.623	1.723	1.670
Field Independent	Mean	6.11	6.63	5.32	5.58	6.68	5.79	6.16
	Std. Deviation	2.105	1.640	2.451	2.116	1.734	2.347	2.218

Table 6.11: CSUQ Average Answers for the Adaptive and Adaptable Interface for Each FD/FI Dimension

FD/FI Dimension		*CSUQ1*	*CSUQ3*	*CSUQ6*	*CSUQ7*	*CSUQ16*	*CSUQ17*	*CSUQ18*	*CSUQ19*
Field Dependent	Mean	5.83	5.33	5.83	5.83	4.75	4.75	4.50	5.25
	Std	1.030	1.303	1.030	1.337	1.422	1.712	1.679	1.422
Intermediate	Mean	5.00	4.83	5.28	4.95	4.47	4.33	4.75	4.83
	Std	1.477	1.337	1.165	1.215	1.030	1.155	.866	1.193
Field Independent	Mean	5.26	4.58	5.37	5.32	4.42	4.32	4.16	4.84
	Std	1.327	1.610	1.165	1.455	1.924	1.797	1.675	1.425

6.5.1 Field Dependent Users

FD users experience a slight increase in user satisfaction. For all questions, the satisfaction increases in the range of 1% to 6%, having an average increase of 3%. For example, it is noticeable that because of the introduction of adaptability, QUIS5 (interface is difficult or easy) increases by 5%, CSUQ3 (I can effectively complete my work with this interface) increases by 6% and CSUQ19 (I am satisfied with this system) increases by 6%.

6.5.2 Intermediate Users

As in the case of the adaptive interface, there is not a relevant change in the user perception. The answers given in this case are all within a range of 2% and –1% of the answers given for the adaptive interface, i.e. user perception does not change because of the introduction of adaptability.

6.5.3 Field Independent Users

For FI users, it can be seen that there has been a reduction in user satisfaction as a result of introducing adaptability. The value of the answers for all questions is reduced in the range of [-2%, 0%] except QUIS27 (which measures if the users feel that the interface is designed for all level of users) that increases almost 6%. These results indicate that although FI users do not favour adaptable interfaces, they recognise that the approach is much more suitable for all kind of users. When interpreting these values, it has to be considered the way in which the experiment was done: all users were presented with the adaptability menu and were “forced” to make changes if they felt those changes could improve their interaction with the BLC. In other words, the approach was guided and such an approach is not suitable to FI users, which are known to prefer to explore the system by themselves (Wang et al., 2000). This guided approach to adaptability appears to be responsible for the slight decrease in user satisfaction for FI users.

The aforementioned results may lead two conclusions. The first conclusion is that adaptability, designed as an option of the interface, can help to improve user satisfaction slightly. FD users will probably use that option, especially if it is presented to them in a guided way while Intermediate and FI user may not favour it. The second conclusion is that data mining is able to capture user preferences to a great extent (at least from a FD/FI perspective), due to the fact that the addition of adaptability does not significantly change the degree of user satisfaction.

6.6 Impact of Adaptability

As showed in the previous section, adaptability can slightly increase user satisfaction, especially for FD users. It is important to have a deep understanding of the changes made to the interface and to identify whether those changes are actually used. These issues will be examined in this section.

The first indication of the impact of adaptability is given by studying the answer to the question of which interface is preferred by the users. The answers showed that 62% of users did not have a preference, and, from the rest, 14% prefer the adaptive interface and 24% the adaptable interface. These results reinforce the idea that adaptability does not have a big impact on user satisfaction.

Considering the changes made through the adaptable menu, 62% of users made some change to the interface; while 38% kept the interface as it was originally presented to them (they preserved the original adaptive interface). From those 62% of people who actually made changes, 44% used those changes to solve the second set of questions, while 56% did not use the changes they made. Two conclusions can be obtained from these results: (1) Adaptability is not very attractive to users, and (2) more than half of the users that made changes using the adaptable menu did not use the additional functionalities. These results are in accordance with other studies in personalisation (Manber et al., 2000). These results also highlight the main problems of adaptability: (a) users are not aware of their own preferences and (b) they do not necessarily understand the options that are being offered.

Tables 6.12 to 6.14 present the percentage of users who made changes for each FD/FI dimension, the percentage of users who used those changes and the percentages of users who preferred each interface.

Table 6.12: Percentage of User, by FD/FI Dimension, That Made Changes

FD/FI Dimension	*Did the user made changes to the interface?*			
	NO		YES	
	% FD/FI dimension	% NO group	% FD/FI dimension	% YES group
Field Dependent	41.7%	31.3%	58.3%	25.9%
Intermediate	66.7%	50.0%	33.3%	14.8%
Field Independent	15.8%	18.8%	84.2%	59.3%

Table 6.13: Percentage of Users, by FD/FI Dimension, That Uses Changes

FD/FI Dimension	*Did the user use the changes made?*					
	NO		YES		No Changes Made	
	% FD/FI dimension	% NO group	% FD/FI dimension	% YES group	% FD/FI dimension	% No Changes
Field Dependent	33.3%	26.7%	25.0%	25.0%	41.7%	31.3%
Intermediate	16.7%	13.3%	16.7%	16.7%	66.7%	50.0%
Field Independent	47.4%	60.0%	36.8%	58.3%	15.8%	18.8%

Table 6.14: Interface Preference of Users by FD/FI Dimension

FD/FI Dimension	*Interface Preference*					
	No Preference		Adaptive		Adaptable	
	% FD/FI dimension	% No Preference	% FD/FI dimension	% Adaptive	% FD/FI dimension	% Adaptable
Field Dependent	58.3%	25.9%	8.3%	16.7%	33.3%	40.0%
Intermediate	83.3%	37.0%	8.3%	16.7%	8.3%	10.0%
Field Independent	52.6%	37.0%	21.1%	66.7%	26.3%	50.0%

Table 4.15 highlighted the tendency of FI users to learn the environment using an active approach. As showed in Table 6.12, 84% of FI users made changes to the interface, which represented 59% of all the users who made any change. This shows a counter intuitive conclusion that while FI users are the ones who use the adaptability provided by the interface more extensively (probably because of their tendency to explore the system by themselves), such use is not translated into an increase in user satisfaction. As for the preference of interface, 26% of FI users preferred the adaptable interface, while the rest preferred the adaptive interface or had no preference.

On the other hand, FD users did not use adaptability as extensively as FI users, with only 58% of them making some changes (Table 6.12), and 25% of users (Table 6.13) actually using those changes to solve the second set of questions. Nevertheless, the increase in user satisfaction of FD users as a consequence of using the adaptable interface was much higher than the increase experienced by FI users. This is probably due to the fact that FD users feel much more comfortable in a guided environment, i.e., guiding users to make changes in the interface, which was used in the experiment.

As for Intermediate users, only 33% made changes to the adaptable interface. From those users who made changes to the interface, only half of them actually used those changes to solve the second set of tasks. The big majority have no preference for the adaptive or adaptable interfaces.

Table 6.14 shows that the introduction of adaptability does not really have a big impact for any FD/FI dimensions, only 33.3% of FD users and 26% of FI users prefer the adaptable interface. This does not mean that adaptability is not an important tool for users. Actually, as it has been showed, there is a relevant group of users who used the adaptability provided by the interface and the satisfaction of some of those users is increased because of adaptability, especially for FD users. On the other hand, what these results imply is that automatic user modelling approaches to capture user behaviour with data mining is efficient because the inclusion of the adaptability in an adaptive interface will not, greatly change user satisfaction.

Table 6.15 presents the interface preference of users depending on the changes made. Most people who do not change the interface, as expected, have no preference between the adaptive or adaptable interface (because they are actually the same interface). There are a small percentage of people (12.5% of FD users and 12.5% of Intermediate users) who prefer the adaptable interface although they did not make changes. As for the users who made changes, the majority of FD users prefer the adaptable interface. On the other hand, almost 70% of FI users had no preference or preferred the adaptive interface. This leads to one of the problems related to adaptability: that users are not necessarily aware of their preferences, which in turns highlights again the necessity of implementing data mining approaches to identify user preferences for personalisation.

Table 6.15: Percentage of Users Classified According to FD/FI Dimension, Their Preference of Interface And If They Have Made or Not Any Changes to the Adaptive Interface

FD/FI Dimension	*Did the user made changes using the Adaptability menu?*					
	No			Yes		
	Interface Preference			Interface Preference		
	No Preference	Adaptive	Adaptable	No Preference	Adaptive	Adaptable
Field Dependent	87.5%	.0%	12.5%	28.6%	14.3%	57.1%
Intermediate	87.5%	.0%	12.5%	75.0%	25.0%	.0%
Field Independent	100.0%	.0%	.0%	43.8%	25.0%	31.3%

6.7 Conclusions

The chapter has presented the design of adaptive and adaptable interfaces for BLC and the experiment conducted to capture user behaviour and user perception with the new interfaces. This chapter has also analysed the impact of introducing adaptivity and adaptability on users' behaviour and perception for BLC interface. A set of very relevant conclusions has been made:

(1) Adaptivity increases user satisfaction. Globally, the increase is 10% of CSUQ and QUIS although FD and FI users have higher increases in satisfaction than Intermediate users.

(2) Adaptivity reduces the interaction needed between users and BLC. It has been showed that there is a link between the time needed to solve a task and user satisfaction.

(3) Adaptability does not have the same impact as adaptivity. The inclusion of adaptability did not globally increase user satisfaction. Nevertheless, adaptability is needed for some specific users, for example FD users.

(4) Users are not usually aware of their preferences and this fact limits the importance of adaptability and increases the need for adaptivity.

(5) Data mining is able to capture user preferences to a great extent (at least from a FD/FI perspective) because that the inclusion of adaptability does not really change the degree of user satisfaction.

In summary, adaptability can potentially improve user experience but it does not have the same impact in increasing user satisfaction as adaptive interfaces. This justifies the importance of implementing adaptive interfaces based on FD/FI dimension. Up to now in this thesis, the identification of the FD/FI dimension of a user is based on the results of CSA test. However, taking the CSA test is time consuming. In addition, some users may be not willing to do it. In order to practically implement an adaptive interface based on FD/FI, the next chapter details the construction of a classification system that automatically identifies the FD/FI dimension based on the interactions between BLC and the users.

Chapter 7

Automatic Cognitive Identification of Digital Library Users for Adaptivity

7.1 Introduction

The previous chapters have shown that FD/FI is key in determining user behaviour, and that the introduction of an adaptive FD/FI interface lead to an increase in user satisfaction. In order to implement this approach, it is necessary that the system knows in advance each user's FD/FI dimension in order to present suitable interfaces. The direct solution is to let each user take the CSA test. Nevertheless, this solution is problematic because the CSA test is time consuming and users may not be willing to take it.

In order to make it feasible to implement an adaptive FD/FI interface, there is a need to automatically identify the FD/FI dimension of each user. This problem is not only important for DLs, but also for any adaptive hypermedia systems based on cognitive styles. There are different applications, mainly learning environments, that adapt presentation and content using a cognitive style approach (Triantafillou et al., 2002; Papanikolau et al., 2003; Bajraktarevic et al. 2003), but all of them assume that the cognitive style of each user is known in advance. While this assumption can be valid for testing environments or systems with a reduced number of users, it is not valid for large-scale adaptive systems. In this context, the idea of automatically identifying users' cognitive styles is essential.

The goal of this chapter is to design a classification system that automatically identifies the FD/FI dimension of each user. Two questions need to be solved: (1) which data is going to be used to produce the classification system and (2) how to obtain such data. Regarding the first question, Chapter 5 showed the connection between user behaviour and FD/FI dimension. Therefore, the behavioural data contains all the necessary information in order to identify the FD/FI dimension of a user. As for the second question, the experiment described in Chapter 4 already collected the data needed. Table 7.1 presents the subset of variables that will be used to generate the classification system.

Table 7.1: Table of Variables That Describe User Behaviour

	Variable Name	Information
1	BS	Number of times that *Basic Search* was used to solve a generic task.
2	AS	Number of times that *Advance Search* was used to solve a generic task.
3	SE	Number of times *Word or Phrase* was used to solve a generic task.
4	ATS	Number of times that *Author, Title* and *Periodical* were used to solve a generic task.
5	NS	Number of times that *New Search* was used to solve a generic task..
6	GB	Number of times that *Go Back* was used to solve a generic task..
7	BF	Number of times *Backward/Forward* was used to solve a generic task.
8	CS	User FD/FI dimension obtained using CSA test
9	WA	*WA* ratio of the user provided by the CSA test.

Note that the definition of FD/FI was constructed using a real number, the WA ratio, where WA scores below 1.03 denote Field Dependent individuals; scores of 1.36 and above denote Field Independent individuals; and scores between 1.03 and 1.35 are classified as Intermediate (Riding, 1991). Considering the FD/FI definition, automatically identifying the FD/FI dimension of a user has two main approaches:

(1) Create a classification system that assigns each user to a FD/FI dimension, i.e. Field Dependent (FD), Intermediate (I) or Field Independent (FI).

(2) Create a classification system that assigns to each user a *WA* ratio from which the FD/FI can be directly obtained. This approach is usually called regression or function approximation.

A classification system can be implemented using a variety of supervised learning techniques and also soft computing techniques. Details of these techniques were presented in Chapter 2. This chapter first presents the generation of systems to automatically identify the FD/FI dimension of a user using neural networks and decision trees. For both cases, classification and regression are considered, and their results compared. Subsequently, a more flexible approach, Neuro-Fuzzy systems (NFS), is considered and implemented.

7.2 Cognitive Identification using Supervised Learning Techniques

In this section, the construction of a FD/FI identification system using neural networks and decision trees is detailed. The reasons for using these techniques are that both have been extensively used for user modelling, as presented in Chapter 2, and both have implementations for classification and regression.

7.2.1 FD/FI Identification using Classification: C4.5 and MLP

Two cognitive classification systems were constructed using: (1) C4.5 as an example of decision trees and (2) Multi-Layer Perceptron (MLP) as an example of neural networks (NNs). Both systems were constructed using Weka Data Mining Software (Witten and Frank, 1999). The training vectors consisted of seven dependent variables which describe the behaviour of the user when interacting with BLC (variables 1-7 in Table 7.1) and one independent variable, the FD/FI dimension (variable 8 in Table 7.1). In order to test the classification systems, two techniques were applied: (1) splitting and (2) 3-fold cross-validation. Splitting divided the file into 66% for training and 33% for testing. The other testing technique, 3-fold cross-validation, divided the data into three parts and each part was used for testing against the classification system obtained with the other two used for training. The classification result considered was the average of the three classification systems.

C4.5 (Quinlan, 1993) has as main parameters: (1) the confidence factor used for pruning, where in general smaller values incur in more pruning, and (2) the minimum number of instances (*MinObj*) that a leaf of the tree should have, where, in general, smaller values create bigger trees. Figure 7.1 and Figure 7.2 present the correct identification rates when using C4.5 with 3-fold cross validation and 66% split respectively, for values of *[0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35]* for the confidence factor and *[1,2,3,4]* of *MinObj*. In both cases, the better classification rate was of 58%.

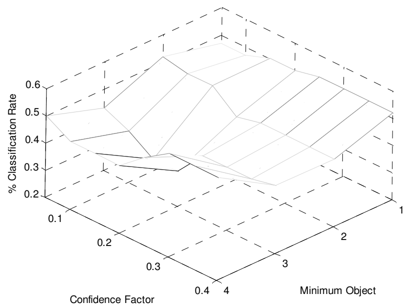


Figure 7.1: C4.5 Correct Identification Rate for Different Confidence Factor and MinObj Values When Using 3-cross Validation

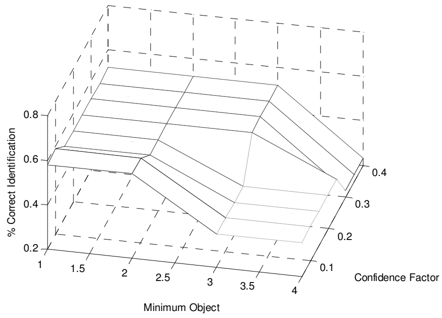


Figure 7.2: C4.5 Correct Identification Rate for Different Confidence Factor and *MinObj* Values When Using 66% Split

MLP was designed with a three-layer network, a sigmoidal transfer function, and a back-propagation learning algorithm. The input layer contained seven neurons, one for each dimension of the input vector, the hidden layer a variable number of neurons, and the output layer three neurons, one for each class. The algorithm was run for 500 epochs, and the learning process stopped when the root mean squared error RMSE<0.01. If that condition was not achieved, the MLP selected was the one that minimised RMSE. Figure 7.3 presents the correct recognition rate for a variable number of neurons in the hidden layer ranging from 5 to 100 in increments of 5, both for 3-fold cross validation and 66% split. As showed in the Figure, the best results obtained are again of 58% for a number of neurons in the hidden layer between 20 and 30. The system was also tested with more complex architectures by testing the recognition capabilities of MLP using two hidden layers. Considering that in the previous case the best results were obtained for a number of neurons in the range 20-30, a MLP with two hidden layers, where each one of those layers had as possible values [10, 20, 30, 40] neurons, was tested. Again, the maximum correct identification rate was 58%. Figure 7.4 presents the correct classification rate for 3-fold cross-validation. Table 7.2 summarises the best classification results of MLP and C4.5 used as classification techniques.

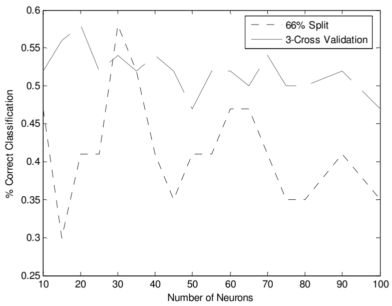


Figure 7.3: MLP Identification Rate for 3-cross Validation and 66% Split Using One Hidden Layer

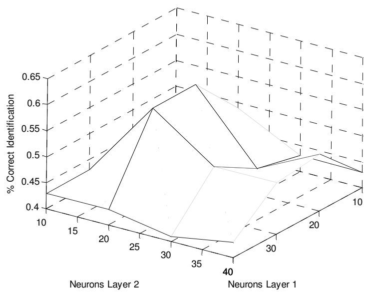


Figure 7.4: MLP Identification Rate for 3-fold Cross-validation and Two Hidden Layers

Table 7.2: Classification Results

	3- Cross Validation	*66% Split*
C4.5	58 %	58 %
MLP	58 % (20 neurons)	58 % (30 neurons)

The classification results obtained are not satisfactory: basically, only one out of two users is assigned a correct FD/FI dimension. Possible reasons for these low recognition rates are: (1) although it seems that this is a traditional classification problem in which all the instances (users) have a class assigned (FD/FI dimension), this is not entirely true, because each user, originally, is assigned a number (*WA* ratio) which is then translated into a class, (2) the definition of FD/FI dimension is fuzzy and the concept is not completely understood, (3) the behaviour of users within a class is not necessarily constant: two users with the same FD/FI dimension can have very different behaviour; for example, if one of them has a *WA* near a border with another dimension and the other has a *WA* away from all borders; the first one will share some behaviour patterns with their neighbours while the second will show a pure behaviour of that FD/FI dimension.

Taking into account these reasons, a regression approach in which the classification system predicts the *WA* ratio (and from that the FD/FI dimension) would possibly produce better results. The main reason is the fact that the regression approach avoids the ill-definition of the FD/FI dimension.

7.2.2 FD/FI Identification using Regression: CART and MLP

Two regression systems were constructed using: (1) CART as an example of classification trees used for regression and (2) MLP as an example of NN architecture used for regression. Both systems were designed with MATLAB, using the NN Toolbox (www.mathworks.com/products/neuralnet) for MLP and the Computational Statistics Toolbox (Martinez and Martinez, 2001) for CART. The training vectors consisted of seven dependent variables (variables 1-7 in Table 7.1) and one independent variable, the *WA* ratio (variable 9 in Table 7.1). Again, in order to test the classification system, two testing techniques were applied: (1) splitting and (2) 3-fold cross-validation. In this case, in order to obtain the correct classification rate, first, the *WA* ratio predicted was transformed into the FD/FI dimension, and then the comparison with the correct FD/FI value was made.

CART (Breiman et al, 1984) has as main parameter the minimum number of items (*SplitMin*) that an impure node, a node that contains more than one item, must have in order to be split. In general, smaller values of *SplitMin* produce bigger and more complex trees. Figure 7.5 presents the correct identification rate of CART for different values of *SplitMin* both for 3-fold cross-validation and 66% split. The best identification results are around 75%.

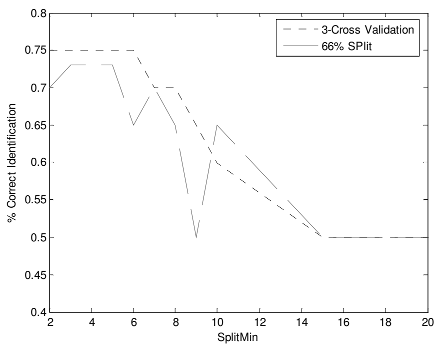


Figure 7.5: CART Correct Identification Rate for Different Values of *SplitMin*

MLP was designed as a three-layer network with back-propagation learning. The first layer contained seven neurons, one for each dimension of the input vectors, the hidden layer a variable number of neurons with a tan-sigmoid transfer function and the output layer one neuron with a linear transfer function. This structure is typical for regression problems (Demuth and Beale, 1998). Each MLP was trained for 500 epochs or until the root mean square (RMSE) error was smaller than 0.01. If that error was not achieved, in order to avoid over fitting, the MLP selected was the one that minimised RMSE. Considering the previous results, Figure 7.6 presents the correct identification rate when using one hidden layer and a number of neurons in the range of [5-40] with increments of five neurons. The best result obtained in this case is of 66% correct identification rate.

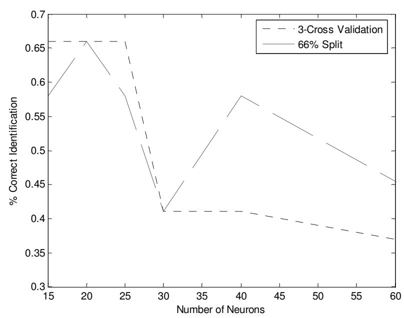


Figure 7.6: MLP Correct Recognition Rate for Different Number of Neurons in the Hidden Layer

Table 7.3 presents the best correct identification results for CART and MLP. In general, there is an increment in the correct identification rate, for both MLP and decision trees compared with the classification approach. This result is in accordance with what Peterson et al. (2003) stated, that the use of category information on the CSA is considerably less reliable than the use of the *WA* ratios. The best results are obtained using the decision tree approach with CART, which achieves up to a 75% correct classification rate. One of the limitations of using a decision tree approach is that the technique is not able to capture the inherent uncertainty that modelling human behaviour entails. In this context, a combination of decision trees with a soft computing technique (like NNs for example) could increase the correct classification rate. Considering that decision trees can also be expressed in the form of classification rules, in the literature there are a variety of algorithms that combine classification rules with neural networks, generally called neuro-fuzzy systems (Jang and Sun, 1995). Neuro-fuzzy systems (NFS) provide an excellent framework to automatically create rules that learn from examples using neural networks and that are able to handle the uncertainty and fuzziness of the concepts and data being used.

Table 7.3: Regression Results

	3- Cross Validation	*66% Split*
CART	75%	73%
MLP	66% (25 Neurons)	66% (20 Neurons)

7.3 FD/FI Identification using NFS

NFS combines fuzzy logic with the learning capabilities of NNs (Jang et al., 1997). One of the most typical NFS is Adaptive-Network-based Fuzzy Inference Systems (ANFIS) (Jang, 1993). The main limitation of ANFIS is the training time needed, which is exponential with the dimension of the input space. This complexity is also a problem when executing the model, which, in this case, is of critical importance due to the necessity of the system to identify the FD/FI dimension of a user in real-time in order to present an adaptive interface. The conclusion is that a reduction of the dimensionality is usually needed for providing personalisation.

7.3.1 Feature Selection

The objective of this section is to identify the subset from the seven original behavioural variables that better characterises the FD/FI dimension within the context of a NFS classifier. The result of the study should indicate which dimensions of the behaviour vector are more relevant to produce a good approximation to WA. Similar techniques have been used very successfully to related problems, as car mileage prediction (MathWorks, 2002).

To identify which variables are more relevant, for all subsets of one, two, three and four variables, a NFS has been trained using data splitting, with 66% for training and 33% for testing (NFS of higher dimensionality were not able to be trained due to the dimensionality problem). All NFS were implemented using MATLAB's Fuzzy Logic Toolbox (www.mathworks.com/products/fuzzylogic). The original fuzzy logic knowledge base was automatically generated using grid partition and each input was assigned two labels. The grid partition divided the input space into a grid generated by the labels. For each system, its Root Mean Square (RMS) training error (the error when the testing file is the same 66% used for training) and its RMS testing error (the error when the testing items used were the 33% of the file not used for training) were collected.

Figure 7.7 presents the training and testing error for each subset of one variable ordered using the training error. The value for IN7 (*BF* in Table 7.1) is not presented because its training error is much higher than the rest of variables. Figure 7.8 to Figure 7.10 present the same results for each subset of two variables, three variables and four variables respectively, showing the subsets with smaller training errors. From Figure 7.7, it can be observed that IN2 (*AS* in Table 7.1) provides the better training error, 0.4448, and IN1 (*BS* in Table 7.1) the best testing error, 0.3499. Figure 7.8 corroborates the importance of IN2 for training error, because the smallest training errors are always obtained using IN2 as one of its inputs. Also, again, the smallest testing error is obtained in combination with IN1, by the pair of inputs IN1-IN6 (*GB*

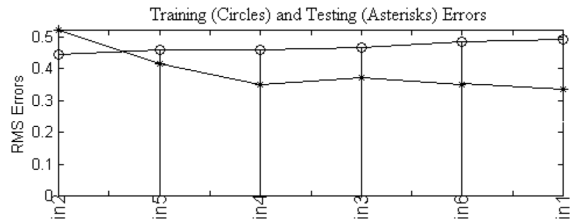


Figure 7.7: RMS Error for One-dimensional Systems

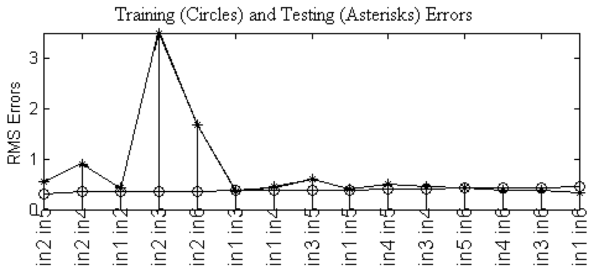


Figure 7.8: RMS Error for Two-dimensional Systems

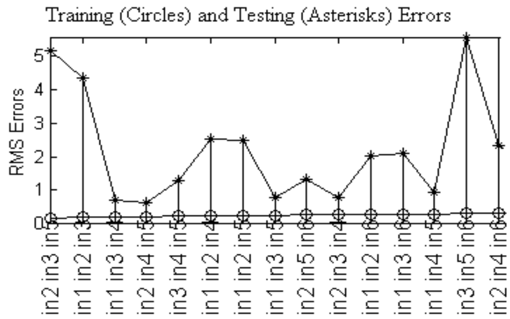


Figure 7.9: RMS Error for Three-dimensional Systems

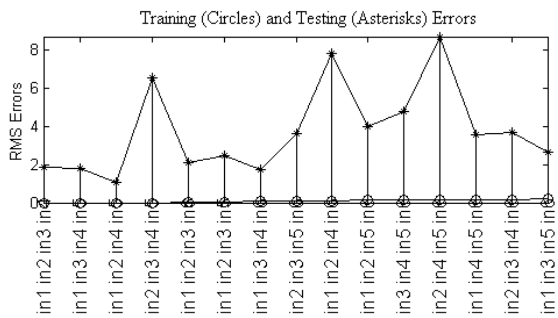


Figure 7.10: RMS Error for Four-dimensional Systems

in Table 7.1). Figure 7.9 and Figure 7.10 show that, although the incorporation of a third and fourth variables can produce subsets of inputs with similar training errors, testing errors are much higher. Typically the set of variables would be chosen according to the smallest testing error, which produces IN1-IN6 as input variables (testing error of 0.3442). Choosing the set of variables with smallest testing error does not imply that these variables will produce the best solution. This motivated the selection of another set of variables that had a testing error similar to IN1-IN6 and also one of the smallest training errors: IN1-IN2, which has a testing error of 0.3597 but a smaller training error than IN1-IN6. Also, this combination of variables is very promising because it combines the variable that minimises the training error (IN2), with the variable that minimises testing error (IN1).

Therefore, a good NFS, in the sense of: (1) size of the fuzzy knowledge base, (2) training and testing time and (3) efficiency of the classification system, can be achieved by a two dimensional system with a possible combination of BS and AS or BS and GB as inputs. The following section checks which one of these combinations produces better results.

7.3.2. NFS for FD/FI Classification

Two NFS were constructed using the Fuzzy Logic Toolbox implementation of ANFIS as learning algorithm. The training vectors were composed of two variables: *BS* and *AS* (variables 1 and 2 of Table 7.1) in the first case, and *BS* and *GB* (variables 1 and 6 of Table 7.1) for the second case. In both cases, the independent variable was the *WA* ratio (variable 8 of Table 7.1). In order to test each NFS, two testing techniques were applied: (1) 66%-33% splitting and (2) 3-fold cross-validation. In both cases, the original fuzzy system was automatically generated using grid partition, with three triangular membership functions per input and singleton outputs. The learning process was run for 50 epochs with the back propagation learning algorithm. To avoid over fitting, the training algorithm selected the fuzzy inference system that minimised the testing error within the first 50 epochs. Figure 7.11, where dots represent testing error and asterisks the training error, shows the learning process when using 66% split, and how the minimum RMS testing error is obtained in epoch 21 and has a value of 0.38 when using *BS-AS* as inputs. Both fuzzy systems have two inputs (*BS-AS* or *BS-GB*), one output (*WA* value), three membership functions per input, three output singletons and only three rules.

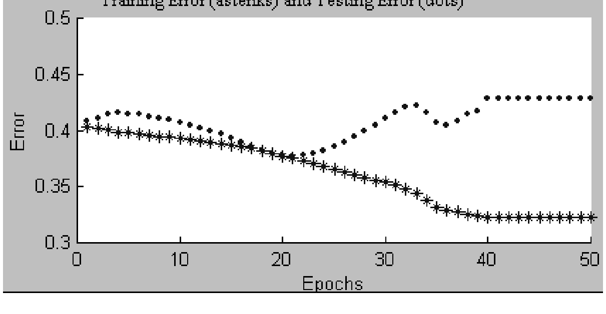


Figure 7.11: Training Error (Asterisks) and Testing Error (Dots) of the Neuro-fuzzy System with 66% Split

Other tests were run for varying number of membership functions and different types of membership functions. While the *WA* ratio produced varied for different values, the correct recognition rate did not change when considering different types of membership functions or varying number of membership functions, except the case of two membership functions which produced worst results. Figure 7.12 shows the output of the testing file when using 66%-33% splitting for the system constructed with *BS-AS*. As showed in the Figure, although the exact *WA* ratio is not predicted, the system is able to give very approximate values that actually classify the user, in general, in the correct FD/FI dimension. Nevertheless, users that have a *WA* ratio near FD/FI have a higher probability of being incorrectly classified. Using splitting the correct classification rate obtained is 86% with *BS-AS* and 79% with *BS-GP* while 3-cross validation provides a 83% correct classification rate with *BS-AS* and 78% with *BS-GB*. The results show that *BS-AS* captures better than *BS-GB* the characteristics of each FD/FI dimension. These results are better than the ones obtained using just a rule-based approach, as done by CART, which implies that the soft computing approach captures, to some extent, the uncertainty of modelling FD/FI.

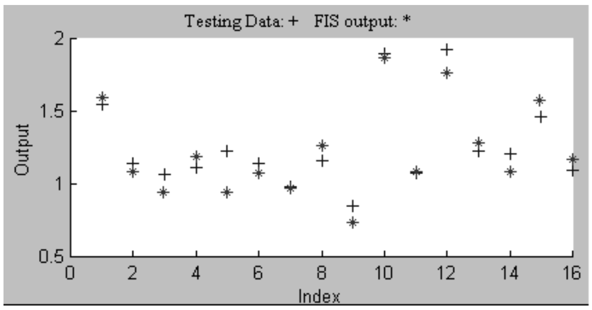


Figure 7.12: Comparison between the Testing WA Ratios (+ Signs) and the Predicted WA Ratios (* Signs)

7.3.3. FD/FI Classification from an Adaptive Perspective

The classification of the *WA* ratio into the FD/FI dimensions given by Riding (1991) is just one possibility of defining the borders, i.e. the borders between FD/FI dimensions are fuzzy. Considering this idea, it can be possible that some of the users that have been assigned to an incorrect FD/FI dimension could find the interface assigned to them useful: i.e. users with *WA* ratio near FD/FI borders have a higher probability of being incorrectly classified. Nevertheless, these users, to some extent, share the behaviour of its neighbours so they can also find their adaptive interface useful. This idea is a natural consequence of the results presented in section 5.2.5 regarding the filtering of users when using Robust Clustering.

Section 5.2.5 concluded that users with an ill defined behaviour were users within a 0.1 margin of the FD/FI borders. In the present context, it would mean that those users will find the interface assigned to each side of the FD/FI border useful. For example, a user with a *WA* ratio of 1.01 could be classified as Field Dependent, or considering that 1.01 is included in [1.03±0.1], with 1.03 the border value between FD and Intermediate, the user can also be classified as Intermediate. In any case, the assumption is that, because the user shares characteristics of FD and Intermediate users, he or she will find both interfaces useful.

Table 7.4 presents the correct classification rates using MLP, CART, and ANFIS with *BS-AS* considering that users within a 0.1 margin of FD/FI borders are correctly classified in any of the two dimensions defined by each side of the border. When using ANFIS, in the worst case, a 91% correct classification rate is achieved. These results, which arouse from an application perspective, show that the automatic identification of a user's FD/FI dimension is feasible and opens the door to build an adaptive interface.

Table 7.4: Classification Results from an Application Perspective

	3- Cross Validation	*66% Split*
CART	80%	82%
MLP	73%	71%
ANFIS – (BS,AS)	91.5%	100%

7.4. Conclusions

The FD/FI dimension of a user is a relevant factor to determine the way in which a user interacts with a web-based application. This importance implies that an adaptive FD/FI interface can be useful to tackle the different problems that users have when interacting with the Web, especially in environments as relevant as digital libraries. The main drawback of building an adaptive interface based on FD/FI is that each user needs to take a cognitive style test (e.g., the CSA) to assign a FD/FI value to a user. This process is time consuming and some users would not be willing to take it.

The main goal of this chapter was to show that it is possible to automatically identify the FD/FI dimension of each user for building an adaptive interface. The main conclusions are:

- In order to better identify the FD/FI dimension of a user, due to the fuzziness of its definition, a regression approach, in which the *WA* ratio is obtained, outperforms a classification approach.

- In general, the use of a soft computing approach can improve the classification rate. Also, a new concept that users near FD/FI borders can have two correct cognitive values has been proposed.
- The results obtained by using a NFS approach in this context have shown that the system can be applied to automatically identify the FD/FI dimension of a user for building an adaptive interface.

These main conclusions may be valid for any web-based applications, as they depend on the nature of the problems, instead of the types of web-based applications. Nevertheless, the correct classification rate for a particular environment will depend on the data that can be collected.

Chapter 8

Conclusions

8.1 Introduction

Digital libraries (DLs) are complex systems that serve users with diverse needs. Personalisation has been recognised as an effective approach to meet the needs of different DL users. However, existing applications are mainly constructed a using user-guided approach in which users need to state their preferences explicitly (Dushay, 2002). In addition, human factors are ignored, such as levels of experience, gender differences, and cognitive styles, in the development of personalised DLs. These limitations in personalised DLs have left two main areas for research: (1) the use of intelligent techniques, such as data mining, for identifying user preferences automatically and (2) the inclusion of human factors for improving the development of personalised DLs. To this end, these two issues have been addressed in this thesis.

In terms of the former, this thesis has showed that data mining is able to automatically identify user preferences. In particular, Robust Clustering is a very effective technique to capture and model user behaviour and user perception. Along with this conclusion, this thesis has also demonstrated that providing adaptivity in DLs can increase user satisfaction while the inclusion of adaptability does not improve user satisfaction. In respects of the latter, the results presented in this thesis have highlighted the relationships between cognitive styles and user behaviour within a DL. It has also showed that there are links between the levels of experience and user perception while novices do not have a homogenous perception. These results have provided the justification for the inclusion of cognitive styles for developing a personalised DL.

The aim of this chapter is to present the conclusions from the investigation. This chapter first begins by describing the results and conclusions presented in this thesis with regards to user behaviour and user perception in DLs (section 8.2). Subsequently, the main conclusions regarding the use of data mining techniques for user modelling are presented (section 8.3). The chapter then moves to describe the limitations of this thesis (section 8.4) and finally finishes with the discussion of future research (section 8.5).

8.2 User Behaviour and Perception in DL

This section summarises the key conclusions from the user behaviour and user perception. In terms of user behaviour, five main patterns have been discovered in this thesis:

(1) Users that exclusively use *Basic Search* in combination with *Word or Phrase* or *Author/Title/Periodical*.

(2) Users that use *Basic Search* and *Advance Search*, in the same proportion, in combination with *Word or Phrase* or *Author/Title/Periodical*.

(3) Users that use *Basic Search* four times as much as they use *Advance Search* and *Word or Phrase* or *Author/Title/Periodical* in the same proportion.

(4) Users that exclusively use *Basic Search* and *Word or Phrase* much more often than *Author/Title/Periodical*.

(5) Users that exclusively use *Basic Search* in combination with *Author/Title/Periodical*.

There are links between each type of user behaviour and the FD/FI dimension: the first pattern represents FI users, and the last pattern represents FD users, patterns (2), (3) and (4) represent Intermediate users. These results reveal that the FD/FI dimension plays a key role in determining user behaviour. From the perspective of personalisation, this finding also implies that one of the important elements that a user model should have is the FD/FI dimension.

In respects of user perception, FD and FI users are not very satisfied with the interface provided by BLC while Intermediate users are happy with existing BLC. In addition, there are relationships between users' perception and their levels of experience. Basically, Intermediate users and expert users demonstrate two different types of perception.

(1) Intermediate users think that BLC is acceptable, easy to use, has an adequate power and is flexible enough. In general, these users were moderately satisfied with BLC, and felt that they had the tools needed to efficiently accomplish a task.

(2) Expert users think that BLC is not up to what they would expect, have a neutral opinion

regarding if it is difficult or easy, think that it has inadequate power and that the system is rigid. In general, these users were dissatisfied with BLC and would expect to have improvement.

These results implies that expert users had more negative perception than intermediate users probably because expert users are more aware of other functionalities excluded in BLC. The results also show that novice users do not have a homogeneous perception.

In addition to the FD/FI dimension and levels of user experience, user perception also has connections with adaptivity and adaptability. The key conclusions include:

(1) Adaptivity increases user satisfaction by around 10%. FD and FI users have higher increases in satisfaction than Intermediate users.

(2) Adaptability does not have the same impact as adaptivity. The inclusion of adaptability does not globally increase user satisfaction. Nevertheless, adaptability is needed for some specific users (e.g., FD users), who, to some extent, benefit from it.

8.3 Data Mining for User Modelling

Data mining has been used as a tool to automatically identify user preferences and generate user models to implement adaptivity. The following conclusions have been produced regarding the use of data mining techniques for modelling user preferences:

- Data mining is able to capture user preferences to the point that the addition of adaptability does not really change the degree of user satisfaction. Therefore, one of the conclusions of this thesis is that data mining can effectively generate reliable user models.
- Robust clustering is an effective data mining technique for capturing user preferences. The advantages that robust clustering has, elimination of bias and filtering of noisy items, are very relevant from a user modelling perspective in which the original data is by definition fuzzy and noisy.

In terms of user modelling with data mining techniques, some relevant results have also been presented:

- In order to better study the different behaviour and perception of FI and FD users, it is recommended to filter users who are near borders of a particular FD/FI dimension and focus on users that have a strong FD/FI behaviour. The results show that in general any value in the area of [+0.1, -0.1] of the borders should be filtered.

- In general, a soft computing approach can be applied to improve the results of modelling user behaviour.
- In order to better identify the FD/FI dimension of a user, due to the fuzziness of its definition, a regression approach, in which the WA ratio is obtained, outperforms a classification approach.

8.4 Limitations of This Study

Any research has a set of limitations that can affect the conclusions found. This section summarises some limitations presented in the thesis:

- One of the main problems of any data mining applications is the amount and quality of the data available. It is arguable that the number of users used for this thesis is not sufficient for capturing the variety of user behaviour and user perception of a DL. Although, in general, it can be considered that the more users the data set has the better, it has to be noted that the study presented in this thesis has used a number of users that is above the average number of users considered in other relevant studies (e.g. Goren-Bar et al., 2001; Semeraro et al., 2001; Tarpin-Bernard and Habib-Mammar, 2005). When confronting the problem of data for user modelling, there is a need to consider the difficulties of collecting data. Some of the main problems are: (1) lack of participants because no reward is offered, and, if a reward is offered, it is not good enough to dramatically increase the numbers, (2) scheduling problems with individuals that want to take part in the experiment, (3) amount of time needed to take the experiment. As an example, each experiment presented in this thesis took about three months.
- The results presented in this thesis are based on the data collected in one particular DL, i.e., BLC. Although in general all DLs offer the same set of functionalities, the results presented in this thesis can be explicitly applied only to BLC. Although it is expected that the results are to a large extent valid for any DLs, there is a need to verify the results by running similar experiments in other DLs.
- Although some human factors are well defined, such as gender difference, others are not so clear. A good example of this case is the levels of experience because the users directly evaluate their experience by themselves. This auto-evaluation may affect the final results of this thesis.
- The instrument for measuring FD/FI dimension used in this thesis was Riding's CSA (1991). This test is just one example of the instruments. There are other instruments such

as Group Embedded Figures Test (GEFT, Witkin et al., 1977). Although both tests measure the same concept, the way of doing it is different. Therefore, their results may also be different.

- The data mining techniques used in Chapter 5 are just a small sample of the possible clustering techniques that can be used to create user models. Each technique has its own limitations so the use of different techniques can produce different results, which, in turn, may affect the results obtained by robust clustering.

Some of these limitations can be used as the starting point for future research, which are discussed in the next section.

8.5 Future Research Directions

The research presented in this thesis has opened up new opportunities for future research, some of which are summarised below.

- Chapter 5 concluded that Robust Clustering is an ideal tool to model user behaviour. Nevertheless, Robust Clustering is based on the combination of a set of standard data mining techniques. Although the combination of techniques used in this thesis has produced very good results, it is not guaranteed that the results produced are optimum. It would be very interesting to study how the combination of different techniques affects the results produced by Robust Clustering.
- Chapter 5 also studied the significance of human factors for creating behavioural and perceptional clusters. It would be interesting to analyse if there are relationships between the set of behavioural clusters and the set of perceptional clusters, i.e. to study if there are relationships between user behaviour and user perception.
- One of the key elements of any projects that use data mining techniques is the amount and quality of the data available. It would be interesting to check if the increase in the number of users would change the results obtained. It also would be interesting to study if the conclusions were different with the data of user behaviour and user perception captured from other digital libraries.
- The research presented in this thesis has assumed that all users interacting with a digital library use a standard computer, instead of a portable device. Nevertheless, portable devices (PocketPCs, PDAs, WAP Phones, etc.) are becoming increasingly important because of the easy and mobile access to information they provide. A possible

direction for future research is to personalise the interface of DLs by considering the characteristics of portable devices and their users.

References

Adams, A., and Blandford, A. (2002) Acceptability of Medical Digital Libraries. *Health Informatics Journal*. 8(2), 58-66. Sheffield Academic Press. ISSN 1460-4582.

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

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

Altman D.G. (1997). *Practical Statistics for medical Research*. Chapman and Hall, London.

Anderson, C. R., Domingos, P., Weld, D.S. (2001). Adaptive Web Navigation for Wireless Devices. *Proceedings of the Seventeenth International Joint Conference on Artifical Intelligence (IJCAI-01)*

Anderson, C., Domingos, P., Weld, D. (2002). Relational Markov Models and their Application to Adaptive Web Navigation. *Proceedings of the Eigth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2002),* 143-152

Ardissono, L., and Goy, A. (2000). Tailoring the Interaction with Users in Web stores. *User Modelling and User Adapted Interaction* 10(4), 251-303.

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

Bajraktarevic, N., Hall, W., and Fullick, P. (2003). Incorporating Learning Styles in Hypermedia Environment: Empirical Evaluation. *The Fourth Conference on Hypertext and Hypermedia, Workshop on Adaptive Hypermedia and Adaptive Web-based systems AH2003*.

Beck, J., Jia, P., Sison, J., and Mostow, J. (2003). Predicting Student Help-Request Behavior in an Intelligent Tutor for Reading. *Proc. of the 9th Int. Conf. on User Modelling, Lecture Notes in Artificial Intelligence vol. 2702*, 303-312.

Beck, J.E., and Woolf, B.P. (1998). Using a Learning Agent with a Student Model. *Proceedings of the 4th International Conference on Intelligent Tutoring Systems LNAI, vol. 1452,* 6-15.

Belkin, N.J., and 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.

Bidel, S, Lemoine, L., and Piat, F. (2003). Statistical machine learning for tracking hypermedia user behavior. *Proceedings of the 2nd Workshop on Machine Learning, Information Retrieval and User Modeling, 9th Int. Conf. in User Modeling,* 56-65.

Blandford, A., Stelmaszewska, H., and Bryan-Kinns, N. (2001). Use of multiple digital libraries: a case study. *Proc. of the JCDL'01*, ACM Press.

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

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

Bonissone,P., Badami, C., and Chiang, X. (1995). Industrial Applications of Fuzzy Logic at General Electric. *Proceedings of the IEEE, 83(3),* 450-465.

Borgman, C.L. (1996) Why are Online Catalogues Still Hard to Use. *Journal of the American Society for Information Science, 47(7),* 493-503

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers *Proceedings of the fifth annual workshop on Computational learning theory*, 144-152

Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. (1984). *Classification and Regression Trees*, Wadsworth Int. Group/Probability Series, Belmont, California, USA,

Brosnan, M.J. (1998). The impact of computer anxiety and self-efficacy upon performance. *Journal of Computer Assisted Learning* , 14, 1998, 223-234.

Brusilovsky, P., and 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.

Bryan-Kinns, N., Blandford, A., and Thimbleby, H. (2000). Interaction Modeling for Digital Libraries, *Workshop on Evaluation of Information Management Systems*.

Callan, J., Smeaton, A., Beaulieu, M., Borlund, P., Brusilovsky, P., Chalmers, M., Lynch, C., Smyth, B., and Straccia, U. (2003). Personalization and Recommender Systems in Digital Libraries*. Proc. of the 2nd DELOS Workshop on Personalization and Recommender Systems in Digital Libraries.*

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

Changchien, S. W. and Lu, T. (2001). Mining association rules procedure to support on-line recommendation by customers and products fragmentation. *Expert Systems with Applications 20*, 325-335.

ChanLin, L. (1998). Students' cognitive styles and the need for visual control in animation. *J. of Educational Computing Research,* 19(4), 351-363.

Chen, S. Y. and Macredie, R. D. (2004) Cognitive Modelling of Student Learning in Web-based Instructional Programmes. *International Journal of Human-Computer Interaction*. 17(3), 375-402.

Chen, S. Y., Ford, N.J. (1997). Towards adaptive information systems: individual differences and hypermedia. *Information Research*, Vol.3 (2)

Chen, S.Y. and Macredie, R.D. (2002). Cognitive Styles and Hypermedia Navigation: Development of a Learning Model. *J. of the American Society for Information Sciences and technolog*. 53(1), 3-15.

Chin, K.K. (1998). Support Vector Machines applied to Speech Pattern Classification. *M.Phil. Thesis, Cambridge University Engineering Department.*

Chin, J.P., Diehl, V.A., and Norman, K.L. (1988). Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface. *ACM CHI'88 Proceedings*, 213-218

Chiu, S. L. (1994). Fuzzy Model Identification based on Cluster Estimation. *J. of Intelligent & Fuzzy Systems 2,* 267-278..

Chuang, Y-R. (1999). Teaching in a Multimedia Computer Environment: A study of effects of learning style, gender, and math achievement. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, Vol. 1(1).

Cohen, S., Fereira, J., Horne, A., Kibbee, B., Mistlebauer, H., and 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.

Conati, C., Gertner, A., Vanlehn, K., and Druzdzel, M. (1997). On-line student modelling for coached problem solving using Bayesian Networks. *Proc. of the 1997 User Modelling Conference, Sardinia.*

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

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

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

Davidson, I., and 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.

Demuth, H., and Beale, M. (1998). Neural Network Toolbox User's Guide. Ver. 3.0, The Mathworks.

Deshpande, M., and Karypis, G. (2001). Selective Markov Models for predicting web-page accesses. *Proceedings SIAM Int. Conference on Data Mining (SDM2001).*

Di Giacomo, M., Mahoney, D., Bollen, J., Monroy-Hernandez, A., and 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.

Doux, A., Laurent, J., Nadal, J. (1997). Symbolic Data Analysis with the k-Means Algorithm for User Profiling. *User Modeling: Proceedings of the sisxth International Conference, UM97*.

Drigas, A., Kouremenos, S., Vrettos, S., Vrettaros and J., Kouremenos, D. (2004). An expert system for job matching of the unemployed. *Expert Systems with Applications 26*, 217-224.

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

Duchamp, D., (1999). Prefetching Hyperlinks. *Proc. Second USENIX Symp. on Internet Technologies and Systems, USENIX, Boulder, CO,* 127-138

Dufresne, A., and Turcotte, S. (1997). Cognitive Style and its Implications for Navigation Strategies. *World Conference on Artificial Intelligence and Education, AI-ED'97.*

Dushay, N. (2002) Models and tools for generating digital libraries: Localizing experience of digital content via structural metadata. *Proceedings of JCDL 02 Conference*, 244-252.

Efron, B., and Tibshirani, R. (1991) Statistical Data Analysis in the Computer Age. *Science, Vol. 253, No. 5018* , 390-395.

Everitt, B.S. (1993). Cluster Analysis. *Arnold. 3rd edition*.

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

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

Fink, J., Kobsa, A., and 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., and Chen, S. (2000). Individual Differences, Hypermedia Navigation and Learning: An Empirical Study. *Journal of Education Multimedia and Hypermedia*, 9 (4), 281-311.

Ford, N., and Miller, D. (1996). Gender differences in Internet perception and use. Electronic Library and Visual Information Research. *The third ELVIRA conference*, 87-202.

Ford, N., Miller, D., and Moss, N. (2001). The role of individual differences in Internet searching: an empirical study, *J. of the American Society Information Science and Technology*, 52(12), 1049-1066.

French, C.J., and Viles, L.C. (1999). Personalized Information Environments. *D-Lib Magazine*, 5(6).

Frias-Martinez, E., Chen, S., and Liu, X. (2006). A Survey of Data Mining Approaches to User Modeling for Adaptive Hypermedia. *IEEE Tran. System, Man and Cybernetics – Part C* 36 (5).

Frias-Martinez, E., Magoulas, G., Chen, S., and Macredie, R. (2005). Modelling Human Behaviour in User-Adaptive Systems: Recent Advances Using Soft Computing Techniques. *Expert Systems with Applications*, *vol. 29(2)*, 2005, 320-329.

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

Fu, Y., Sandhu, K., and Shih, M.-Y. (1999). Clustering of Web users based on access patterns. *Proceedings of the 1999 KDD Workshop on Web Mining, San Diego, CA. Springer-Verlag*

George G., and Cardullo, F. (1999). Application of Neuro-Fuzzy Systems to Behavioral Representation in Computer Generated Forces. *Proceedings of the Eighth Conference on*

Computer Generated Forces and Behavioral Representation (May 11-13, Orlando, Florida), 575-585.

Goldberg, D. (1989). *G*enetic Algorithms in Search, Optimization, and Machine Learning. *Addison-Wesley.*

Gonzalves, M.A, and Fox, E.A. (2002). 5SL: a language for declarative specification and generation of digital libraries. *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries Portland, Oregon, USA*, 263 - 272.

Goren-Bar, D., Kuflik, T., Lev, D. and Shoval, P. (2001). Automatic Personal Categorization Using Artificial Neural Networks. *Proceedings of the 8th International Conference on User Modeling 2001, Lecture Notes in Artificial Intelligence, vol. 2109,* 188-198.

Graff, M. (2003). Cognitive Style and Attitudes Towards Using Online Learning and Assesment Methods, *Electronic Journal of e-learning*, Vol. 1, Issue 1, 21-28.

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

Gyenesei, A. (2000). A fuzzy approach for mining quantitative association rules. *TUCS Technical Report 336, University of Turku, Department of Computer Science, Finland.*

Hand D.J. (1997). Construction and assessment of classification rules. *Wiley*.

Hand, D.J., Mannila, H., and Smyth, P. (2001). Principles of data mining, *MIT Press*.

Harrison, A. W. and Rainer, R. K. (1992). The influence of individual differences on skill in end-user computing, *J. of Management Information Systems*, 9(2), 93-111.

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

Hay, B, Wets, G., and Vanhoof, K. (2001) Clustering navigation patterns on a website using a Sequence Alignment Method. *IJCAI 2001 Workshop on Intelligent Techniques for Web Personalization.*

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

Hicks, D.L., and 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.

Hicks, D.L., Tochterman, K., Eich, S. and Rose, R. (1999) Using Metadata to support customization. *Proceedings of the third IEEE Metadata Conference, Bethesda, MD, USA.*

Hong, J., Heer, J. Waterson, S., and Landay, J.A. (2001). WebQuilt: A Proxy-based Approach to Remote Web Usability Testing. *ACM Transactions on Information Systems.* 19, 263-385.

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

Hsieh, N. (2004). An integrated data mining and behavioural scoring model for analysing bank customers. *Expert Systems with Applications 27*, 623-633.

Jain, A.K., and Dubes, R.C. (1999). Data clustering. *ACM Computing Survey*, 31:264-323.

Jang, J.S. (1993). ANFIS: Adaptive-Network-Based Fuzzy Inference Systems. *IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23(3)*, 665-685.

Jang, J.S.R. and Sun, C.T. (1995). Neurofuzzy modelling and control. *Proc. IEEE (3) Vol. 83*, 378-406

Jang, J.S.R., Sun, C.T. and Mizutani E. (1997). Neurofuzzy and Soft Computing: a Computational Approach to Learning and Machine Intelligence. *Prentice Hall*.

Jayawardana, C., Hewagamage, K.P., and 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.

Jonassen, D. H. and Grabowski, B. (1993). Individual Differences and Instruction. *Allen and Bacon, New York.*

Joshi, A., Joshi, K. and Krishnapuram, R. (2000). On mining Web Access Logs. *Proceedings of the ACM-SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, 63-69.

Kass, G.V. (1980). An Exploratory Technique for Investigating Large Quantities of Categorical Data. *Applied Statistics 29: p. 119-127*.

Klir, J. and Yuan, B. (1995). Fuzzy Sets and Fuzzy Logic. Theory and Applications. *Prentice Hall*.

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

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

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

Kuo, R.J. and Chen, J.A. (2004). A decision support system for order selection in electronic commerce based on fuzzy neural network supported by real-coded genetic algorithm. *Expert Systems with Applications 26*, 141-154.

Lampinen, T. and Koivisto, H. (2002). Profiling Network Applications with Fuzzy C-Means Clustering and Self-Organising Map. *Proceedings of the 1st Int. Conf. on Fuzzy Systems and Knowledge Discovery: Computational Intelligence for the E-Age,* 300-304.

Large, A. and Beheshti, J. (1997) OPACs: a research review. *Library & Information Science Research 19(9),* 111-133.

Large, A., Beheshti, J., Rahman, T. (2002). Gender differences in collaborative web searching behavior: an elementary school study. *Information Processing and Management: an International Journal Volume 38 , Issue 3* , 427 - 443

Lazander, A.W., Biemans, H.J.A., and Wopereis, I.G.J.H. (2000). Differences between novice and experienced users in searching information on the World Wide Web. *Journal of the American Society for Information Science* , *51*(6), 576-581.

Lee, R.S.T. (2001). iJADE IWShopper: A New Age of Intelligent Web Shopping System Based on Fuzzy-Neuro Agent Technology. *Web Intelligence: Research and Development,* Zhong, N., Yao, Y., Liu, J., Ohsuga, S. Editors, *Lecture Notes in Artificial Intelligence, vol. 2198,* 403-412.

Lee, W. and Tsai, T. (2003). An interactive agent-based system for concept-based web search. *Expert Systems with Applications 24*, 365-373.

Lewis, J.R. (1995). IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for User. *International Journal of Human-Computer Interaction*. *7*, 57-78.

Li, Y. and Zhong, N. (2004). Web mining model and its applications for information gathering. *Knowledge-Based Systems, 17*, 207–217.

Liaw, S. and Huang, H. (2003). An Investigation of User Attitude toward search engines as an information retrieval tool. *Computers in Human Behavior*, Vol. 19(6), 751-765

Lippmann, R.P. (1987). An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine, April, 4-28.*

Liu, H., Tarima, S., Borders, A.S., Getchell, M.L., and Stromberg, A.J. (2005). Quadratic regression for gene discovery and pattern recognition for non cyclic short time-course microarray experiments. *BMC Bioinformatics 6,* 106

Liu, M., and Reed, W.M. (1995). The effect of hypermedia assisted instruction on second-language learning. *J. Educational Computing Researc 12(2)*, 159-175.

Liu, Y., and Ginther, D. (1999). Cognitive styles and distance education. *Online Journal of Distance Learning Administration,2*(3).

Loia, V., and Luongo, P. (2002). An Evolutionary Approach to Automatic Web Page Categorization and Updating. *Web Intelligence: Research and Development, LNAI 2198*, 292-301.

Looney, C.G. (1997). Pattern Recognition using Neural Networks: Theory and Algorithms for Engineers and Scientists. *Oxford University Press*.

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.D., Papanikolau, K., and Grogoriadou, M. (1999). Towards a computationally intelligent lesson adaptation for a distance learning course. *Proceedings of the IEEE International Conference on Tools with Artificial Intelligence, 5-11 Chicago, IL, USA*.

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

Magoulas, G.D., Papanikolau, K.A. and Grigoriadou, M. (2001). Neuro-fuzzy synergism for planning the content in a web-based course. *Informatica Vol. 25(1),* 39-48.

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

Marchionini, G., Plaisant, C. and Komlodi, A. (1998). Interfaces and tools for the Library of Congress National Digital Library Program, *Information Processing&Management*, 34(5), 535-555.

Marrison, D.L., and Frick, M.J. (1994). The effect of agricultural students' learning styles on academic achievement and their perceptions of two methods of instruction. *Journal of Agricultural Education*, 35(1), 26-30.

Martin, J.K., and 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).*

Martin-Bautista, M. J., Vila, M.A., and Larsen H.L. (2000). Building Adaptive User Profiles by a Genetic Fuzzy Classifier with feature selection. *Proceedings of The 9th International IEEE Conference on Fuzzy Systems, San Antonio, TX, USA.*

Martinez, L. W., and Martinez, A.R. (2001). Computational Statistics Handbook with MATLAB. *Chapman & Hall / CRC*.

Mathworks. (2002). Gas Mileage Prediction with Fuzzy Logic Toolbox 2.2.4. *http://www.mathworks.com/products/fuzzylogic/demos.html?file=/products/demos/shipping/fuzzy/gasdemo.html*

McKeown, K.R., Elhadad, N., and 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.

Min, H., Smolinski, T. and Boratyn, G. (2001). A GA-based Data Mining Approach to Profiling the Adopters of E-purchasing. *Proceedings of the 2001 IEEE 3rd Int. Conf. on Information Reuse and Integration* , 1-6.

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

Mitchell, T. J. F., Chen, S. Y. and Macredie, R. D. (2005) Hypermedia Learning and Prior Knowledge: Domain Expertise vs. System Expertise. *Journal of Computer Assisted Learning* 21, 53-64.

Mobasher, B. Jain, N., and Srivastava, J. (1997). Web Mining: Patterns from WWW Transactions. *Technical Report TR97-050, Department of Computer Science, University of Minnesota, March 1997.*

Mobasher, B., and Cooley, R. (2000). Automatic Personalization Based on Web Usage Mining. *Communications of the ACM, vol. 43(8),* 142-151.

Mobasher, B., Dai, H., Luo, T., and Nakagawa, M. (2001). Effective personalization based on Association Rule discovery from web usage data. *3rd ACM Workshop on Web Information and Data Management.*

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

Morahan-Martin, J., Males, females, and the Internet, *J. Gackenbach (Ed.), Psychology and the Internet: Intrapersonal, interpersonal, and transpersonal implication,* 1998, 169-197.

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

Nasraoui, O. and Petenes, C. (2003). Combining Web Usage Mining and Fuzzy Inference for Website Personalization. *Proceedings of WEBKDD 2003: Web Mining as Premise to Effective Web Applications*, 37-46.

Nasraoui, O., Cardona, C., Rojas, C., and Gonzalez, F. (2003b). Mining Evolving User Profiles in Noisy Web Clickstream Data with a Scalable Immune System Clustering Algorithm. *Workshop Notes of WEBKDD 2003: Web Mining as Premise to Effective and Intelligent Web Applications, Washington DC, 71-81.*

Nasraoui, O., Frigui, H., Joshi, A. and Krishnapuram, R. (1999). Mining Web Access Logs Using Relational Competitive Fuzzy Clustering. *Proceedings 8th Intl. Fuzzy Systems Association World Congress - IFSA 99.*

Nuernberg, P.J.m, Furuta, R., Legget, J.J., Marchal, C.C. and Shipman, F.M. (1995) Digital Libraries: Issues and Architecture. *Proceedings of ACM Digital Libraries*, 147-153

Pal, S.K., Talwar, V. and Mitra, P. (2002). Web Mining in Soft Computing Framework: Relevance, State of the Art and Future Directions. *IEEE Tran.s on Neural Networks, Vol. 13(5)*, 1163 - 1177.

Paliouras, G., Karkaletsis, V., Papathedorou, C., and 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.

Paliouras, G., Papatheodorou, C., Karkaletsis, V., and Spyropoulos, C. (2000). Clustering the users of large web sites into communities. *17th International Conference on Machine Learning,* 719-726.

Palmquist, R.A., and Kim, K.S. (2000). Cognitive style and on-line database search experience as predictors of search performance, *J. of the American Society Information Science* , 51(6), 558-566.

Papanikolau, K.A., Grigoradiou, M., Kornilakis, H., and Magoulas, G.D. (2003). Personalizing the Interaction in a wed-based Educational Hypermedia System: the case of INSPIRE. *User Modelling and User-Adapted Interaction 13*, 213-267.

Park, T.I., Yi, S.G., Lee, S., Yoo, D.H., Ahn, J.I., and Lee, Y.S. (2003). Statistical tests for identifying differentially expressed genes in time-course microarray experiments. *Bioinformatics. 19*, 694-703.

Perkowitz, M. and Etzioni, O. (2000). Adaptive Web Sites. *Communications of the ACM, Vol. 43(8),* 152-158.

Perkowitz, M.,and Etzioni, O. (1999) Towards Adaptive Web Sites: Conceptual Framework and Case Study. *The Eigth International World Wide Web Conference*.

Peterson, E. R., Deary, I. J., and Austin, E. J. (2003) The reliability of Riding's Cognitive Style Analysis test. *Personality and Individual Differences,* 34(5), 881-891.

Privoxy. (2006). http://www.privoxy.org/

Quinlan, J.R. (1993). C4.5: Programs for Machine Learning. *Morgan Kaufmann*.

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

Ramsden, A. (2002) MuOpenLibrary project. *Personalisation and digital libraries seminar, Open University, UK, October 2002*

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

Riding, R. and Rayner, S. G. (1998), *Cognitive Styles and Learning Strategies*, David Fulton , 1998

Riding, R. J. (1991). Cognitive Styles Analysis. *Learning and Training Technology, Birmingham*.

Riding, R. J., and Grimley, M. (1999). Cognitive style, gender, and learning from multimedia materials in 11 year-old children. *British Journal of Education Technology*, 30(1), 43-56.

Riding, R.J. and Cheema, I. (1991). Cognitive Styles. *Educational Psycholog*y, 11(3/4), 193-215.

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

Roh, T. H., Oh. K.J. and Han, I. (2003). The collaborative filtering recommendation base don SOM cluster-indexing CBR. *Expert Systems with Applications 25*, 413-423.

Roy, M. and Chi, M.T.H. (2003). Gender differences in patterns of searching the web. *Journal of Educational Computing Research* 29(3), 335-348.

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.

Sas, C., Reilly, R. and O'Hare, G. (2003). A Connectionist Model of Spatial Knowledge Acquisition in a Virtual Environment. *Proc. 9th Int. Conf. on User Modelling*, 40-48.

Schmitt, C., Dengler, D. and Bauer, M. (2003). Multivariate Preference Models and Decision Making with the MAUT Machine. *Proc.9th Int. Conf. on User Modeling, LNAI, vol. 2702*, 297-302.

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

Semeraro, G., Abbatista, F., Fanizzi, N., and Ferilli, S. (2000). Intelligent Information Retrieval in Digital Library Service. *Proc. 1st 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., and 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., and Abbattista, F. (2001). Learning Interaction Models in a Digital Library Service. *8th Int. Conf. on User Modelling*, LNAI 2109, 44-53.

Shardanand, U., and Maes, P. (1995). Social information filtering: Algorithms for automating word of mouth. Proc. *ACM '95 Conference on Computer Human Interaction (CHI), 210-217.*

Sheperd, A., Watters, C. and Marath, A.T. (2002). Adaptive User Modeling for Filtering Electronic News. *Proc. of the 35th Annual Hawaii Intl. Conf. on System Sciences (HICSS-02), Vol. 4*, 102.2.

Sinha, N.K., Gupta, M.M. and Zadeh, L.A. (eds) (2000). Outline of Computational Theory of Perceptions Based on Computing with Word. *Soft Computing & Intelligent Systems*, 3-22.

Stathacopoulou, R., Grigoriadou, M., and Magoulas, G.D. (2003). A Neuro-fuzzy Approach in Student Modeling. *Proceeding of the 9th Int. Conf. on User Modeling, UM2003, LNAI 2702*, 337-342.

Stelmaszewska, H. and Blandford, A. (2004). From physical to digital: a case study of computer scientists' behaviour in physical libraries. *International Journal of Digital Libraries*. 4.2. 82-92.

Swift, S., Tucker, A., Vinciotti, V., Marin, N., Orengo, C., Liu, X. and Kellam, P. (2004). Consensus Clustering and Functional interpretation of gene-expression data. *Genome Biology*. 5:R94, http://genomebiology.com/2004/5/11/R94.

Tarpin-Bernard, F., Habieb-Mammar, H. (2005). Modeling Elementary Cognitive Abilities for Adaptive Hypermedia Presentation. *User Modeling and User Adapted Interaction 15 (5),* 459-495.

Theng, Y.L., Duncker, E., and Mohd, N. (1999). Design Guidelines for User-Centred Digital Libraries. *Proc. 3rd Conf. on Research and Advanced Technology for Digital Libraries, LNAI 1696*, 167-183.

Torkzadeh, G. and Van Dike, T.P. (2002). Effects of Training on the Internet Self-efficacy and Computer User Attitude. *Computers in Human Behavior, Vol. 18*, 479-494.

Triantafillou, E., Pomportsis, A., and Georgiadou, E. (2003). AEC-CS: Adaptive Educational System based on Cognitive Styles. *WASWBE'02, Workshop on Adaptive Systems for Web-based Education, 2nd Intl. Conf. on Adaptive Hypermedia and Adaptive Web Based Systems*.

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

Uebersax, J.S. (1987). Diversity of decision-making models and the measurement of interrater agreement. *Psycological Bulletin*. 101, 140-146.

Valiquette, C., Lesage, A., Cyr, M., ad Toupin, J. (1994). Computing Cohen's kappa coefficients using SPSS MATRIX. *Behavioral Research Methods, Instruments and Computers*. *26*, 60-61.

Van House, N.A. (1995) User Needs Assesment and Evaluation for the UC Berkeley Electronic Environmental Library Project: A preliminary report. *Proceedings of ACM Digital Libraries*, 71-76.

Vapnik, V., (1995). The nature of statistical learning theory. *Springer-Verlag, New York.*

Vrettos, S. and Stafylopatis, A. (2001). A Fuzzy Rule-Based Agent for Web Retrieval-Filtering. *Web Intelligence: Research and Development,* Zhong, N., Yao, Y., Liu, J., Ohsuga, S. Editors, *Lecture Notes in Artificial Intelligence, vol 2198,* 448-453.

Wang, P., Hawk, W.B., and Tenopir, C. (2000). User's interaction with world wide web resources: an exploratory study using a holistic approach. *Information Processing and management* 36, 229-251.

Webb, A. (1999). Statistical Pattern Recognition. *Arnold*.

Webb, G. I., Chiu, B. C., and Kuzmycz, M. (1997). Comparative Evaluation of Alternative Induction Engines for Feature Based Modelling. *Int. J. of Artificial Intelligence in Education*, 8:97-115, 1997

Webb, G.I., Pazzani, M.J. and Billsus, D. (2001). Machine Learning for User Modeling. *User Modeling and User-Adapted Interaction, Vol. 11, Issue 1-2*, 19-29.

Weller, H. G., Repman, J., and Rooze, G. E. (1994). The relationship of learning, behavior, and CS in hypermedia instruction: Implications for design of HBI. *Computers in the Schools*, 10(3/4), 401-420.

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.

Witkin, H.A., and Goodenough, D.R. (1981). Cognitive styles: Essence and origins. *New York: International Universities Press.*

Witkin, H. A., Moore, C. A., Goodenough, D. R., & Cox, P. W. (1977). Field-dependent and field independent cognitive styles and their educational implications. *Review of Educational Research*, 47(1), 1-64.

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

Witting, F. (2003). ML4UM for Bayesian Network User Model Acquisition. *Second Workshop on Machine Learning, Information Retrieval and User Modeling*, June 22-26, Pittsburgh, PA, USA.

Wolfinger, R.D., Gibson, G., Bennet, L., C., and Paules, R.S. (2001).Assesing Gene Significance from cDNA Microarray Expression Data via Mixed Models. *J. of Computational Biology. 8*, 625-637.

Yan, J., Ryan, M. and Power, J.(1994).Using Fuzzy Logic. *Prentice Hall*.

Yan, T.W., and Garcia-Molina, H. (1995). SIFT --- A tool for wide-area information dissemination. In *Proceesings of the 1995 UNSENIX Technical Conference*, 177-186.

Zhang, X. (2003). Discriminant Analysis as a Machine Learning Method for Revision of User Stereotypes of Information Retrieval Systems. *MLIRUM'032nd Workshop on Machine Learning, Information Retrieval and User Modeling, 9th Int. Conf. On User Modeling, Pittsburgh, PA, USA.*

Zhu, T., Greiner, R., Haubl, G. (2003). Learning a Model of a Web User's Interests. *Proceedings of the 9th International Conference on User Modeling, LNAI 2702.*

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

Zukerman, I., and Albrecht, D.W. (2001). Predictive statistical models for user modeling. *A. Kobsa (ed), User Modeling and User-Adapted Interaction Journal, Vol II, Issues 1-2*, 5-18.

Yi, M.Y. and Hwand, Y. (2003). Predicting the use of web-based systems, self-efficacy, enjoyment, learning goal orientation, and the technology acceptance model. *Int. J. of Human-Computer Studies,* 59, 431-449.

Appendix A

Complete QUIS, CSUQ and ASQ Questionnaires

Questionnaire for User Interface Satisfaction

OVERALL REACTION TO THE SOFTWARE		0	1	2	3	4	5	6	7	8	9		NA
1.	terrible											wonderful	
2.	difficult											easy	
3.	frustrating											satisfying	
4.	inadequate power											adequate power	
5.	dull											stimulating	
6.	rigid											flexible	
SCREEN		**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**		**NA**
7. Reading characters on the screen	hard											easy	
8. Highlighting simplifies task	not at all											very much	
9. Organization of information	confusing											very clear	
10. Sequence of screens	confusing											very clear	
TERMINOLOGY AND SYSTEM INFORMATION		**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**		**NA**
11. Use of terms throughout system	inconsistent											consistent	
12. Terminology related to task	never											always	
13. Position of messages on screen	inconsistent											consistent	
14. Prompts for input	confusing											clear	
15. Computer informs about its progress	never											always	
16. Error messages	unhelpful											helpful	

LEARNING		0	1	2	3	4	5	6	7	8	9		NA
17. Learning to operate system	difficult	○	○	○	○	○	○	○	○	○	○	easy	○
18. Exploring new features by trial and error	difficult	○	○	○	○	○	○	○	○	○	○	easy	○
19. Remembering names and use of commands	difficult	○	○	○	○	○	○	○	○	○	○	easy	○
20. Performing tasks is straightforward	never	○	○	○	○	○	○	○	○	○	○	always	○
21. Help messages on the screen	unhelpful	○	○	○	○	○	○	○	○	○	○	helpful	○
22. Supplemental reference materials	confusing	○	○	○	○	○	○	○	○	○	○	clear	○
SYSTEM CAPABILITIES		**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**		**NA**
23. SYSTEM speed	too slow	○	○	○	○	○	○	○	○	○	○	fast enough	○
24. SYSTEM reliability	unreliable	○	○	○	○	○	○	○	○	○	○	reliable	○
25. SYSTEM tends to be	noisy	○	○	○	○	○	○	○	○	○	○	quiet	○
26. Correcting your mistakes	difficult	○	○	○	○	○	○	○	○	○	○	easy	○
27. Designed for all levels of users	never	○	○	○	○	○	○	○	○	○	○	always	○

Computer System Usability Questionnaire

		1	2	3	4	5	6	7		NA
1. Overall, I am satisfied with how easy it is to use system	DISAGREE								AGREE	
2. It was simple to use system	DISAGREE								AGREE	
3. I can effectively complete my work using system	DISAGREE								AGREE	
4. I am able to complete my work quickly using system	DISAGREE								AGREE	
5. I am able to efficiently complete my work using system	DISAGREE								AGREE	
6. I feel comfortable using system	DISAGREE								AGREE	
7. It was easy to learn to use system	DISAGREE								AGREE	
8. I believe I became productive quickly using system	DISAGREE								AGREE	
9. system gives error messages that clearly tell me how to fix problems	DISAGREE								AGREE	
10. Whenever I make a mistake using system , I recover easily and quickly	DISAGREE								AGREE	
11. The information provided with system is clear	DISAGREE								AGREE	
12. It is easy to find the information I needed	DISAGREE								AGREE	
13. The information provided for system is easy to understand	DISAGREE								AGREE	

			1	2	3	4	5	6	7		NA
14.	The information is effective in helping me complete the tasks and scenarios	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
15.	The organization of information on system screens is clear	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
16.	The interface of system is pleasant	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
17.	I like using the interface of system	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
18.	system has all the functions and capabilities I expect it to have	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
19.	Overall, I am satisfied with system	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
			1	2	3	4	5	6	7		NA

After-Scenario Questionnaire

			1	2	3	4	5	6	7		NA
1.	Overall, I am satisfied with the ease of completing the tasks in this scenario	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
2.	Overall, I am satisfied with the amount of time it took to complete the tasks in this scenario	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
3.	Overall, I am satisfied with the support information (online-line help, messages, documentation) when completing the tasks	DISAGREE	◌	◌	◌	◌	◌	◌	◌	AGREE	◌
			1	2	3	4	5	6	7		NA

Appendix B

Modified QUIS and CSUQ Questionnaires

Note: BLC = Brunel Library Catalogue.

		1	2	3	4	5	6	7		NA
1. Overall, I am satisfied with how easy it is to use BLC	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
2. I think that BLC is simple to use	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
3. BLC has helped me to effectively complete my tasks	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
4. I think that BLC helps me to complete my work quickly	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
5. I think that BLC helps me to efficiently complete my work	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
6. I feel comfortable using BLC	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
7. I think that BLC is easy to learn	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
11. I think that the information provided by BLC is clear	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
12. I think that with BLC it has been easy to find the information I needed	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
13 The information provided for BLC is easy to understand	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
15. BLC's organization of information is clear	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
16. I think that BLC interface is pleasant	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
17. I like using BLC interface	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
18. BLC has all the functions and capabilities I expect	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐
19. Overall, I am satisfied with BLC	strongly disagree	☐	☐	☐	☐	☐	☐	☐	strongly agree	☐

		0	1	2	3	4	5	6	7	8	9		NA
17. I find that learning to operate BLC is	difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	easy	☐
18. I find that with BLC exploring new features by trial and error is	difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	easy	☐
20. I find that in BLC performing tasks is straightforward	never	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	always	☐
21. I find BLC messages	unhelpful	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	helpful	☐

OVERALL REACTION TO BLC		0	1	2	3	4	5	6	7	8	9		NA
1. ☐	terrible	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	wonderful	☐
2. ☐	difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	easy	☐
3. ☐	frustrating	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	satisfying	☐
4. ☐	inadequate power	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	adequate power	☐
5. ☐	dull	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	stimulating	☐
6. ☐	rigid	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	flexible	☐

		0	1	2	3	4	5	6	7	8	9		NA
23. BLC speed is	too slow	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	fast enough	☐
25. BLC tends to be	noisy	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	quiet	☐
26. Correcting your mistakes ☐	difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	easy	☐
27. In my opinion BLC is designed for all levels of users	never	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	always	☐

		0	1	2	3	4	5	6	7	8	9		NA
9. I think that BLC organization of information is	confusing	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	very clear	☐
10. I think that the sequence of screens of BLC is	confusing	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	very clear	☐