LNAI 3169

Bamshad Mobasher
Sarabjot Singh Anand (Eds.)

Intelligent Techniques for Web Personalization

IJCAI 2003 Workshop, ITWP 2003
Acapulco, Mexico, August 2003
Revised Selected Papers

Springer

Bamshad Mobasher Sarabjot Singh Anand (Eds.)

Intelligent Techniques for Web Personalization

IJCAI 2003 Workshop, ITWP 2003
Acapulco, Mexico, August 11, 2003
Revised Selected Papers

Springer

Volume Editors

Bamshad Mobasher
DePaul University, Center for Web Intelligence
School of Computer Science, Telecommunication and Information Systems
Chicago, Illinois, USA
E-mail: mobasher@cs.depaul.edu

Sarabjot Singh Anand
University of Warwick, Department of Computer Science
Coventry CV4 7AL, UK
E-mail: s.s.anand@warwick.ac.uk

Library of Congress Control Number: 2005935451

CR Subject Classification (1998): I.2.11, K.4.1, K.4.4, C.2, H.3.4-5, H.5.3, I.2

ISSN 0302-9743
ISBN-10 3-540-29846-0 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-29846-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11577935 06/3142 5 4 3 2 1 0

Preface

Web personalization can be defined as any set of actions that can tailor the Web experience to a particular user or set of users. The experience can be something as casual as browsing a Web site or as (economically) significant as trading stock or purchasing a car. The actions can range from simply making the presentation more pleasing to anticipating the needs of a user and providing customized and relevant information. To achieve effective personalization, organizations must rely on all available data, including the usage and click-stream data (reflecting user behavior), the site content, the site structure, domain knowledge, user demographics and profiles. In addition, efficient and intelligent techniques are needed to mine these data for actionable knowledge, and to effectively use the discovered knowledge to enhance the users' Web experience. These techniques must address important challenges emanating from the size and the heterogeneity of the data, and the dynamic nature of user interactions with the Web.

E-commerce and Web information systems are rich sources of difficult problems and challenges for AI researchers. These challenges include the scalability of the personalization solutions, data integration, and successful integration of techniques from machine learning, information retrieval and filtering, databases, agent architectures, knowledge representation, data mining, text mining, statistics, user modelling and human–computer interaction. Throughout the history of the Web, AI has continued to play an essential role in the development of Web-based information systems, and now it is believed that personalization will prove to be the "killer-app" for AI.

The collection of papers in this volume include extended versions of some of the papers presented at the ITWP 2003 workshop as well as a number of invited chapters by leading researchers in the field of intelligent techniques for web personalization. The first chapter in the book provides a broad overview of the topic and a comprehensive bibliography of research into Web personalization that has been carried out in the past decade. The rest of the chapters are arranged in five parts each addressing a different aspect of the topic. Part I consists of three chapters focussed on user modelling. In the first of these chapters, Craig Miller describes the current state of our understanding of how users navigate the Web and the challenges in modelling this behavior. Further, the necessary capabilities of a working cognitive model of Web navigation by a user, an implementation of such a model and its evaluation are described. Next, Naren Ramakrishnan describes his view of personalization based on capturing the interactional aspects underlying a user's interaction with the Web in an attempt to model what it means for a website to be personable. The final chapter in this part of the book, by Bettina Berendt and Max Teltzrow, rather than modelling the user per se, discusses results from a user study aimed at understanding the privacy concerns of users and the effect of these concerns on current personalization strategies. They argue for improved communication of privacy practice and benefits to the users resulting from data disclosure and a better understanding of the effect of various types of data on the performance of the resulting personalization.

The second part of the book consists of three chapters on recommender systems. In the first of these chapters Fabiana Lorenzi and Francesco Ricci provide a survey of case-based approaches to recommendation generation and propose a unifying framework to model case-based recommender systems. In the following chapter Lorraine McGinty and Barry Smyth describe a novel approach to item selection, known as adaptive selection, that balances similarity and diversity during a user interaction with a reactive recommender system. They show how adaptive selection can dramatically improve recommendation efficiency when compared with standard forms of critiquing. Finally, Robin Burke surveys the landscape of possible hybrid systems for personalization, describing several ways in which base recommenders can be combined to form hybrid systems.

The third part of the book consists of three chapters on enabling technologies. The first of these, by Chuck Lam, introduces the use of associative neural networks for user-based as well as item-based collaborative filtering. It also discusses the use of principal component analysis for dimensionality reduction. In the next chapter Tiffany Tang et al. propose the use of heuristics to limit the size of the candidate item set, hence improving the performance of traditional user-based collaborative filtering. Finally, Birgit Hay et al. propose a new algorithm for mining interesting Web navigational patterns that can be used for personalizing future interactions.

The fourth part of the book consists of three chapters on personalized information access. The first of these chapters, by Kevin Keenoy and Mark Levene, surveys the current state of the art in personalized Web search. Apostolos Kritikopoulos and Martha Sideri follow this with a chapter describing an approach to personalizing search engine results using Web communities. Finally Tingshao Shu et al. present an approach to predicting a user's current information needs using the content of pages visited and actions performed.

The final part of the book consists of four chapters on systems and applications. The first chapter in this part, by Barry Smyth et al., describes the application of personalized navigation to mobile portals to improve usability. Next, Magdalini Eirinaki et al. present their system for personalization based on content structures and user behavior. Arif Tumer et al. then present a privacy framework for user agents to negotiate the level of disclosure of personal information on behalf of the user with Web services. Finally, Samir Aknine et al. present a multi-agent system for protecting Web surfers from racist content.

August 2005

Bamshad Mobasher
Sarabjot Singh Anand

Author Index

Table of Contents

User Modelling

Recommender Systems

Enabling Technologies

Personalized Information Access

Systems and Applications

Intelligent Techniques for Web Personalization

Sarabjot Singh Anand[1] and Bamshad Mobasher[2]

[1] Department of Computer Science, University of Warwick, Coventry CV4 7AL, UK
s.s.anand@warwick.ac.uk
[2] Center for Web Intelligence, School of Computer Science, Telecommunications
and Information Systems, DePaul University, Chicago, Illinois, USA
mobasher@cs.depaul.edu

Abstract. In this chapter we provide a comprehensive overview of the topic of Intelligent Techniques for Web Personalization. Web Personalization is viewed as an application of data mining and machine learning techniques to build models of user behaviour that can be applied to the task of predicting user needs and adapting future interactions with the ultimate goal of improved user satisfaction. This chapter survey's the state-of-the-art in Web personalization. We start by providing a description of the personalization process and a classification of the current approaches to Web personalization. We discuss the various sources of data available to personalization systems, the modelling approaches employed and the current approaches to evaluating these systems. A number of challenges faced by researchers developing these systems are described as are solutions to these challenges proposed in literature. The chapter concludes with a discussion on the open challenges that must be addressed by the research community if this technology is to make a positive impact on user satisfaction with the Web.

1 Introduction

The term *information overload* is almost synonymous with the Internet, referring to the sheer volume of information that exists in electronic format on the Internet and the inability of humans to consume it. The freedom to express oneself through publishing content to the Web has a number of advantages, however, the task of the consumer of this content is made more difficult not only due to the need to assess the relevance of the information to the task at hand but also due to the need to assess the reliability and trustworthiness of the information available.

Information retrieval technologies have matured in the last decade and search engines do a good job of indexing content available on the Internet and making it available to users, if the user knows exactly what he is looking for but often, search engines themselves can return more information than the user could possibly process. Also, most widely used search engines use only the content of Web documents and their link structures to assess the relevance of the document to the user's query. Hence, no matter who the user of the search engine is, if the same query is provided as input to the search engine, the results returned will be exactly the same.

The need to provide users with information tailored to their needs led to the development of various information filtering techniques that built profiles of users and

B. Mobasher and S.S. Anand (Eds.): ITWP 2003, LNAI 3169, pp. 1–36, 2005.

attempted to filter large data streams, presenting the user with only those items that it believes to be of interest to the user.

The goal of personalization is to provide users with what they want or need without requiring them to ask for it explicitly [1]. This does not in any way imply a fully-automated process, instead it encompasses scenarios where the user is not able to fully express exactly what the are looking for but in interacting with an intelligent system can lead them to items of interest.

Intelligent Techniques for Web Personalization is about leveraging all available information about users of the Web to deliver a personal experience. The "intelligence" of these techniques is at various levels ranging from the generation of useful, actionable knowledge through to the inferences made using this knowledge and available domain knowledge at the time of generating the personalized experience for the user. As such, this process of personalization can be viewed as an application of data mining and hence requiring support for all the phases of a typical data mining cycle [2] including data collection, pre-processing, pattern discovery and evaluation, in an off-line mode, and finally the deployment of the knowledge in real-time to mediate between the user and the Web.

In this chapter we provide an overview of the topic of Intelligent Techniques for Web Personalization. In Section 2 we describe the process of personalization in terms of an application of a data mining to the Web. Section 3 provides a classification of approaches to Web personalization while in Section 4 we describe the data available for mining in the Web domain, specifically for the generation of user models. Section 5 describes the various techniques used in generating a personalized Web experience for users highlighting the advantages and disadvantages associated with each approach. Issues associated with current approaches to Web personalization are discussed in Section 6. The important issue of evaluating Web personalization is discussed in Section 7. Finally the chapter concludes in Section 8 with a discussion on the current state and future direction of research in Web personalization.

2 The Personalization Process

Personalization aims to provide users with what they need without requiring them to ask for it explicitly. This means that a personalization system must somehow infer what the user requires based on either previous or current interactions with the user. This in itself assumes that the system somehow obtains information on the user and infers what his needs are based on this information.

In the context of this book, we focus on personalization of the Web or more generally, any repository of objects (items) browseable either through navigation of links between the objects or through search. Hence, the domain we address includes Intranets and the Internet as well as product/service catalogues. More formally, we assume that we are given a universe of n items, $I = \{i_j : 1 \leq j \leq n\}$, and a set of m users, $U = \{u_k : 1 \leq k \leq m\}$, that have shown an interest, in the past, in a subset of the universe of items. Additionally, each user, u_k, may be described as a t-dimensional vector $(a_1^k, a_2^k, ..., a_t^k)$ and each item, i_j, by an s-dimensional vector $(b_1^j, b_2^j, ..., b_s^j)$. Further domain knowledge about the items, for example, in the form of an ontology, may also be available. We will assume the existence of a function $r_{u_k} : I \rightarrow [0,1] \cup \perp$ where $i_j = \perp$ signifies that the item i_j has not been rated by the user, u_k [1] that assigns a rating to each item in I. Let $I_k^{(u)}$ be the set of items currently unrated by the user u_k, i.e. $I_k^{(u)} = \{i_j : i_j \in I \wedge r_{u_k}(i_j) = \perp\}$. Similarly let $I_k^{(r)}$ be the set of items rated by the user u_k, i.e. $I_k^{(r)} = I - I_k^{(u)}$.

The goal of personalization is to recommend items, i_j, to a user u_a, referred to as the *active user*, where $i_j \in I_a^{(u)}$ that would be of interest to the user.

Central to any system capable of achieving this would be a user-centric data model. This data may be collected implicitly or explicitly but in either case must be attributable to a specific user. While this seems obvious, on the Web it is not always straightforward to associate, especially implicitly collected data with a user. For example, server logs provide a rich albeit noisy source of data from which implicit measures of user interest may be derived. Due to the stateless nature of the Web, a number of heuristics must be used along with technologies such as cookies to identify return visitors and attribute a sequence of behaviours to a single user visit/transaction [3].

Once the data has been cleansed and stored within a user-centric model, analysis of the data can be carried out with the aim of building a user model that can be used for predicting future interests of the user. The exact representation of this user model differs based on the approach taken to achieve personalization and the granularity of the information available. The task of learning the model would therefore differ in complexity based on the expressiveness of the user profile representation chosen and the data available. For example, the profile may be represented as vector of 2-tuples $u_k^{(n)}(< i_1, r_{u_k}(i_1) >, < i_2, r_{u_k}(i_2) >, < i_3, r_{u_k}(i_3) > .... < i_n, r_{u_k}(i_n) >)$ where i_j's $\in I$ and r_{u_k} is the rating function for user u_k. In the presence of a domain ontology, the user profile may actually reflect the structure of the domain [4], [5], [6]. Recently, there has been a lot of research interest in generating aggregate usage profiles rather than individual user profiles [7], that represent group behaviour as opposed to the behaviour of a single user. The distinction between individual and aggregate profiles for personalization is akin to the distinction between lazy and eager learning in machine learning.

The next stage of the process is the evaluation of the profiles/knowledge generated. The aim of this stage is to evaluate how effective the discovered knowledge is in predicting user interest. Common metrics used during this phase are coverage, mean absolute error and ROC sensitivity. See Section 7 for a more detailed discussion on evaluation metrics.

The deployment stage follows evaluation, where the knowledge generated and evaluated within the previous two stages of the process is deployed to generate recommendations in real-time as the users navigate the Web site. The key challenge at this stage is scalability with respect to the number of concurrent users using the system.

An essential, though often overlooked, part of the personalization process is the monitoring of the personalization. Anand et al. suggest that the success of the person-

[1] Note that a while we assume a continuous scale for rating, a number of recommender systems use a discrete scale. However, our formalisation incorporates this case as a simple linear transformation can be performed on the scale to the [0,1] interval.

alization should be based on lift in business process based metrics [8]. Other than just monitoring the effectiveness of the knowledge currently deployed, an essential aspect of monitoring the effect of personalization is profile maintenance. User interests are dynamic and their evolution must be detected and adapted to for effective personalization to take place. Additionally, personalization itself can influence user behaviour. Techniques for identifying this change and adapting the personalization system to it are not well understood, requiring further research.

In terms of the learning task, personalization can be viewed as a

- Prediction Task: A model must be built to predict ratings for items not currently rated by the user. Depending on whether the user ratings are numeric or discrete, the learning task can be viewed as a being one of regression or classification.
- Selection Task: A model must be built that selects the N most relevant items for a user that the user has not already rated. While this task can be viewed as one of post processing the list of predictions for items generated by a prediction model, the method of evaluating a selection based personalization strategy would be very different from that of a prediction based strategy (see Section 7).

3 Classifications of Approaches to Personalization

In this section we discuss various dimensions along which personalization systems can be classified based on the data they utilize, the learning paradigm used, the location of the personalization and the process that the interaction takes with the user.

3.1 Individual Vs Collaborative

The term personalization impresses upon the individuality of users and the need for systems to adapt their interfaces to the needs of the user. This requires data collected on interactions of users with the system to be modelled in a user-centric fashion. Typically, data is collected by the business with which the user is interacting and hence the business has access to data associated with all its customers.

A personalization system may choose to build an individual model of user likes and dislikes and use this profile to predict/tailor future interactions with that user. This approach commonly requires content descriptions of items to be available and are often referred to as *content-based filtering systems*. NewsWeeder [9] is an example of such a system that automatically learns user profiles for netnews filtering. In the case of NewsWeeder the user provides active feedback by rating articles on a scale of 1 to 5. The process of building a profile for a user requires the transformation of each article into a bag or words representation, with each token being assigned a weight using some learning method such as *tfidf* [10] or minimum description length [11]. The profile is then used to recommend articles to the user.

An alternative approach to recommendation is to not only use the profile for the active user but also other users with similar preferences, referred to as the active user's neighbourhood, when recommending items. This approach is referred to as *social or collaborative filtering*. An example of such a system is GroupLens, also aimed at recommending netnews articles [12]. GroupLens defines a user profile as an n-dimensional vector, where n is the number of netnews articles. If an articles has been rated by the user, its corresponding element in the vector contains the rating. Note that as opposed to content-based filtering, the actual content descriptions of the articles is not part of the profile. Articles not currently rated by the active user but rated highly by users in the neighbourhood of the active user are candidates for recommendation to the active user. While GroupLens only uses rating data, collaborative approaches that utilise both content and user rating data have also been proposed [13], [14].

A major disadvantages of approaches based on an individual profile include the lack of serendipity as recommendations are very focused on the users previous interests. Also, the system depends on the availability of content descriptions of the items being recommended. On the other hand the advantage of this approach is that it can be implemented on the client side, resulting in reduced worries for the user regarding privacy and improved (multi-site) data collection for implicit user preference elicitation.

The collaborative approach also suffers from a number of disadvantages, not least the reliance on the availability of ratings for any item prior to it being recommendable, often referred to as the new item rating problem. Also, a new user needs to rate a number of items before he can start to obtain useful recommendations from the system, referred to as the new user problem. These issues along with others such as sparseness are discussed in greater detail in Section 6.

3.2 Reactive Vs Proactive

Reactive approaches view personalization as a conversational process that requires explicit interactions with the user either in the form of queries or feedback that is incorporated into the recommendation process, refining the search for the item of interest to the user. Most reactive systems for personalization have their origins in case-based reasoning research [15], [16], [17]. Reactive systems can be further classified based on the types of feedback they expect from the user. Common feedback mechanisms used by these systems include value elicitation, critiquing/tweaking [17], rating and preference feedback [18]. Value elicitation and tweaking/critiquing are feature based approaches to feedback. While in value elicitation the user must provide a rating for each feature of each recommendation object presented to the user, based on its suitability to the users needs, in tweaking/critiquing the user only provides directional feedback (for example, "too high", "too low") on feature values for the recommended object. Rating and preference are feedback approaches at the object level. In rating based feedback, the user must rate all the recommendations presented to him, based on their 'fit' with his requirements. In preference feedback the user is provided with a list of recommendations and is required to choose one of the recommendations that best suits his requirement. The system then uses this feedback to present the user with other, similar objects. The iterations continue until the user finds an object of interest or abandons the search. Examples of such recommender systems include Entree [19], DIETORECS [20] and ExpertClerk [21]. For a more detailed discussion on these feedback mechanisms see [16], [17].

Proactive approaches on the other hand learn user preferences and provide recommendations based on the learned information, not necessarily requiring the user to provide explicit feedback to the system to drive the current recommendation process. Proactive systems provide users with recommendations, which the user may choose to select or ignore. The users feedback is not central to the recommendation process as is the case in reactive systems. Examples of proactive systems include the recommendation engine at Amazon.com [22] and CDNOW, Web mining based systems such as [23], [24], [25], GroupLens [26], MovieLens [27] and Ringo [28].

3.3 User Vs Item Information

Personalization systems vary in the information they use to generate recommendations. Typically, the information utilized by these systems include:

- Item Related Information: This includes content descriptions of the items being recommended and a product/ domain ontology
- User Related Information: This includes past preference ratings and behaviour of the user, and user demographics

Systems that use item related information generally deal with unstructured data related to the items [29], [9]. Once this data has been processed, into relational form such as a bag-of-words representation commonly used for textual data, a user profile is generated. The profile itself may be individual as in the case of NewsWeeder [9] or based on group behaviour [13].

Most systems that use user related information, tend to be based on past user behaviour such as the items they have bought or rated (implicitly or explicitly) in the past. Fewer systems use demographic data within the recommendation process. This is due to the fact that such data is more difficult to collect on the Web and, when collected, tends to be of poor quality. Also, recommendations purely based on demographic data have been shown to be less accurate than those based on the item content and user behaviour [30]. In his study of recommender systems, Pazzani collected demographic data from the home pages of the users rather than adding the additional burden on the user to provide data specifically for the system. Such data collection outside of a controlled environment would be fraught with difficulties. In Lifestyle Finder [31], externally procured demographic data (Claritas's PRIZM) was used to enhance demographic attributes obtained from the user, through an iterative process where the system only requests information pertinent to classifying the user into one of 62 demographic clusters defined within the PRIZM classification. Once classified, objects most relevant to that demographic cluster are recommended to the user.

In addition to systems that depend solely on item related or user related information, a number of hybrid systems have been developed that use both types of information. Section 5.4 discusses these systems in greater detail. An example of such a system is the bibliographic system proposed by Haase et al. [5]. In addition to data on user behaviour, two domain ontologies are also available to the system describing the content of the items in a more structured form than that used by NewsWeeder. Hasse et al. define a user model based on user expertise, recent queries, recent relevant results (implicitly obtained by user actions on previous recommendations), a vector of weights for content features and a similarity threshold.

3.4 Memory Based Vs Model Based

As described in Section 2, the process of personalization consists of an offline and online stage. The key tasks during the offline stage are the collection and processing of data pertaining to user interests and the learning of a user profile from the data collected. Learning from data can be classified into memory based (also known as lazy) learning and model based (or eager) learning based on whether it generalizes beyond the training data when presented with a query instance (online) or prior to that (offline).

Traditional Collaborative filtering (see Section 5.2) and content based filtering based systems (see Section 5.1) that use lazy learning algorithms [32], [33] are examples of the memory-based approach to personalization, while item-based and other collaborative filtering approaches that learn models prior to deployment (see Section 5.3) are examples of model-based personalization systems.

As memory based systems simply memorise all the data and generalize from it at the point of generating recommendations, they are more susceptible to scalability issues. Section 6.3 discusses some of the solutions proposed in literature to address the scalability of memory based personalization systems. As the computationally expensive learning occurs offline for model-based systems, they generally tend to scale better than memory based systems during the online deployment stage. On the other hand, as more data is collected, memory based systems are generally better at adapting to changes in user interests compared to model based techniques that must either be incremental or be rebuilt to account for the new data.

Memory based systems generally represent a user profile using a vector representation though more expressive representations such as associative networks [34] and ontological profiles [35] have also been proposed.

3.5 Client Side Vs Server Side

Approaches to personalization can be classified based on whether these approaches have been developed to run on the client side or on the server-side. The key distinction between these personalization approaches is the breadth of data that are available to the personalization system. On the client side, data is only available about the individual user and hence the only approach possible on the client side is *Individual.*

On the server side, the business has the ability to collect data on all its visitors and hence both Individual and Collaborative approaches can be applied. On the other hand, server side approaches generally only have access to interactions of users with content on their Web site while client side approaches can access data on the individuals interactions with multiple Web sites.

Given these characteristics, most client side applications are aimed at personalized search applicable across multiple repositories [36], [37]. The lack of common domain ontologies across Web sites, unstructured nature of the Web and the sparseness of available behavioral data currently reduce the possibilities for personalization of navigational as opposed to search based interactions with the Web.

4 Data

Explicit data collection has typically been modelled as ratings of items, personal demographics and preference (including utility) data. Preference data refers to information that the user provides that can help the system discern which items would be useful to the user. When declared explicitly it can take the form of keywords/product categories (e.g. genres in movie/music databases) or values for certain attributes that describe the objects (e.g. cotton as the preferred material in an apparel store). Utility data refers to information regarding how the user would measure the fit of the objects recommended with his requirements. For example, if two suppliers for the same product exist, with supplier A providing the product at a premium rate over supplier B but with the advantage of free insurance for a predefined period, different users will have different thresholds for the extra cost of purchasing the product from supplier A [38], [39]. We refer to data that defines these preferences as utility data. Rating data may take the form of a discrete numeric value or an unstructured textual form such as reviews of products. While using numeric values is computationally easier to leverage, they are also less reliable as users associate these discrete values subjectively, for example, three stars according to one user may be equivalent to two stars for another user.

Implicit data collection refers to any data that can be collected on the user unobtrusively by "watching" their interaction with the system. Once again the objective is to obtain ratings from various discernable actions of the user. The actions and the associated inferences are dependent on the type of system being personalized. For example, in the Web domain in general, the linger time [2] is taken to be an implicit indicator of interest in the object [26]. Additionally, in an e-commerce context, actions such as adding an item to the basket, purchasing an item, deleting an item from the basket can all imply differing levels of interest in the item [40] as could bookmarking of pages [41], visit frequency, following/passing over a link and saving a page on a news/content site [42]. Claypool et al. [43] evaluated a number of possible implicit interest indicators and concluded that linger time and amount of scrolling can be useful indicators of interest. They also provided a useful categorization of interest indicators.

One issue with implicit data collection is that most observations are positive in nature and it is up to the system to use some heuristics to decide on what defines a negative observation. For example, the use of the back button after the user spends only a short time on a page can be inferred as being a negative observation or the choosing of a document from a list may render the other items in the list as being classified as not interesting [44], [45]. Even when certain negative actions are observed such as the deletion of an item from a shopping trolley, heuristics must be used to decide on how the initial interest in an item, i.e. inserting of the product in the shopping basket, must be amended when the item is deleted from the basket. Schwab et al. [46] propose a system that only employs positive feedback data to avoid the use of such heuristics. Hotle and Yan [47] showed that implicit negative feedback data can greatly improve the effectiveness of a conversational recommendation system, however, care must be taken in deciding what feedback can be attributed as being negative.

It is worth noting at this point that some of the implicit interest indicators used in these evaluations required data to be collected on the client side, while other data can be collected on the Web server, albeit with some inaccuracy, servicing the user request.

Explicit data input has a cost associated with it as it requires users to detract from their principle reason for interacting with the system and provide data, the benefits of which are intangible to the user. A number of studies carried out by the IBM User Interface Institute in the early 1980's confirm that, in general, users are motivated to get

[2] The time spent viewing an item and its associated content.

started with using a system and do not care about spending time up front on setting up the system, as is required by personalization systems that are dependent on explicit data being provided by the user. Carroll and Rosson [48] refer to this phenomenon as the "paradox of the active user" as users would save time in the long term by taking some initial time to optimize the system but that's not how people behave in the real world. While the studies were not aimed at personalization systems per se, the conclusion of the studies that engineers must not build products for an idealized rational user, rather they must design for the way users actually behave is just as valid for personalization systems. Studies in personalization show that without tangible benefits for the user, the user tends to read a lot more documents than they bother ranking [49]. By generating data that indicates a users interest in an object without the user needing to provide this information would result in more data and a reduction in sparsity, that exists especially in large information resources, typical of the Web. Additionally, privacy concerns also imply that users on the Internet tend to only provide accurate information that is deemed essential. Berendt and Teltzrow [50] suggest that users on the Internet exhibit varying degrees of privacy concerns and a large percentage of users would be happy to impart with various degrees of private information based on the perceived benefit to them in doing so. An interesting implication for designing personalization systems.

5 Personalization Techniques

In this section we describe the various approaches used for generating a personalized Web experience for a user.

5.1 Content-Based Filtering

Content based filtering systems have their roots in information retrieval. The approach to recommendation generation is based around the analysis of items previously rated by a user and generating a profile for a user based on the content descriptions of these items. The profile is then used to predict a rating for previously unseen items and those deemed as being potentially interesting are presented to the user. A number of the early recommender systems were based on content-based filtering including Personal WebWatcher [45], InfoFinder [51], NewsWeeder [9], Letizia [44] and Syskill and Webert [52]. Mladenic [53] provides a survey of the commonly used text-learning techniques in the context of content filtering, with particular focus on representation, feature selection and learning algorithms.

Syskill and Webert learns a profile from previously ranked Web pages on a particular topic to distinguish between interesting and non-interesting Web pages. To learn the profile, it uses the 128 most informative words, defined using expected information gain, from a page and trains a naïve Bayes classifier to predict future, unseen pages as potentially interesting or not for the user. The user may provide an initial profile for a topic, which in the case of Syskill and Webert, requires the definition of conditional probabilities for each word, given a page that is (not) interesting to the user. As pages get rated, these initial probabilities are updated, using conjugate priors [54], to reflect the rating of the pages by the user.