Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Multi-Channel Personalization for Internet Banking

Eugenio Cesario[2], Francesco Folino[2],
Riccardo Ortale[2], Domenico Saccà[1,2]

RT-ICAR-CS-04-03 **Marzo 2004**

Consiglio Nazionale delle Ricerche, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR)
– Sede di Cosenza, Via P. Bucci 41C, 87036 Rende, Italy, URL: *www.icar.cnr.it*
– Sezione di Napoli, Via P. Castellino 111, 80131 Napoli, URL: *www.na.icar.cnr.it*
– Sezione di Palermo, Viale delle Scienze, 90128 Palermo, URL: *www.pa.icar.cnr.it*

Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Multi-Channel Personalization for Internet Banking

Eugenio Cesario[2], Francesco Folino[2],
Riccardo Ortale[2], Domenico Saccà[1,2]

Rapporto Tecnico N.:
RT-ICAR-CS-04-03

Data:
Marzo 2004

---

[1] Istituto di Calcolo e Reti ad Alte Prestazioni, ICAR-CNR, Sede di Cosenza, Via P. Bucci 41C, 87036 Rende(CS)
[2] Università degli Studi della Calabria, DEIS, Via P. Bucci 41C, Rende (CS)

Multi-Channel Personalization for Internet Banking

Eugenio Cesario[2], Francesco Folino[2], Riccardo Ortale[2], and Domenico Saccà[1,2]

[1] ICAR-CNR
Via Bucci 41c
I87036 Rende (CS) - Italy
e-mail: sacca@icar.cnr.it
[2] DEIS, University of Calabria
Via Bucci 41c
I87036 Rende (CS) - Italy
e-mail: cesario,ffolino,ortale@si.deis.unical.it, sacca@unical.it

Abstract. This study deals with the individualization of users' interaction through the Web with contents and services from traditional bank information systems. In particular, we focus on a model of personalization that provides users with three orthogonal functionalities, namely adaptive content delivery, navigation support and form filling. The latter is a novel feature for the discovery of typical exploitation patters originated by the user while interacting with a given Web service. This allows to notably increase the service usability, by automatically filling in its form fields, at successive visits of that user, with input values for which a high probability exits that they are manually supplied. The proposed approach is at the heart of a personalization engine, developed around a suite of Web mining techniques, that acts as an adaptive front-office for traditional bank information systems. Precisely, the engine configures as an adaptive Web interface through which the user can perform a number of tasks, such as on-line payments and bank transfers, monitoring account balances and reading personalized news. Two main befits characterize our personalization engine: the ease of deploying it within preexistent bank contexts and the flexibility with which the engine can interact with the user across distinct access channels, ranging from the Web to either Wap-compliant mobile phones or the human voice carried out by the telephone system.

1 Introduction

The increasing number of services available from the Web has determined a substantial change in the nature of the Web itself, which has evolved from a publishing medium into a pervasive platform for transaction processing. In such a new scenario, both traditional industrial businesses and *ad-hoc* Web operators act recognizing the potential economic value of end users, who may turn into customers. The attempt to attract new visitors requires to overcome the limitations of the flat delivery of Web contents and services, that are indistinctly served to all end users. Personalization is a fertile research area whose methods and techniques address the enhancement of the interaction with Web sites. The idea is exploiting Web users' browsing behavior to infer a model of their distinct peculiarities, such as their requirements, tastes and preferences. This allows Web sites to tailor both the delivery and the presentation of products, services and information to the profile of their target users. Many distinct application domains, ranging from e-learning to i9nformation retrieval and from tourism to e-commerce, have benefited from the adoption of an adaptive paradigm for the delivery of Web contents/services. This has materialized into a number of different solutions addressing personalization from different points of views, such as recommender systems and adaptive Web sites. We propose a novel personalization engine, based on data mining techniques, which automatizes an adaptation process consisting of three main steps: *user categorization*, that aims at grouping users into homogeneous clusters from a behavioral point of view; *behavioral modelling*, that is the act of learning reliable models of the preferences, requirements and browsing behaviour of the users within each cluster; *personalization*, that coincides with the actual delivery of services and contents tailored to the profile of the target user. In particular, personalization is achieved as a combination of three orthogonal functionalities: *adaptive content delivery*, i.e. the recommendation of pages with contents pertaining to visitor interests and requirements; *adaptive navigation support*, i.e. the act of suggesting personalized navigation paths; *adaptive form filling*, i.e. the process of learning the typical input from a user to a Web service in order to fill in its form fields at successive requests from that visitor. Our proposal has been developed in the context of the ITB@NK system, a research project targeted at improving users' Web experience with contents and services from traditional bank information-systems. Notwithstanding, its modularity and extensibility features make it an effective solution to the problem of Web individualization across a variety of distinct application environments.

Traditional attempts to Web personalization available from the literature divide into four main approaches, namely solutions based on *manual decision rules*, *content-based filtering*, *collaborative filtering*, and *Web usage mining*. Manual decision rule systems [13], apply fixed adaptation rules to static user profiles, typically obtained at registration time. Their major limitations are the considerable effort required to define and update these rules, the progressive degrade of personalization accuracy due to the aging of static profiles and the involvement of end users in providing their preferences, that typically diminishes the attractiveness of Web sites. Content-based filtering systems [12, 16, 11], first learn a model of user preferences in Web contents, and then try to estimate their interest in unseen Web documents on the basis of some content similarity between these documents and the inferred profiles. A main drawback of such an approach is the inherent difficulty in actually catching a semantic similarity between the contents of two Web pages. Collaborative filtering systems [9, 15], do not rely on any a priori knowledge of Web content. Rather they guess visitor interests in a specific Web item (either a textual content or a service) by detecting a neighborhood of users, whose known preferences in that specific item are then somehow correlated to make suitable interest predictions. However, users with no explicit interests do not benefit from recommendations. Also, the outcome of personalization is affected by the correlation step, that penalizes visitors with few interests. Finally, efficiency and scalability limitations emerge with large numbers of users and items. Systems relying on Web usage mining methods [14, 7, 17] apply machine learning and statistical techniques to visitor clickstream data to automatically reconstruct their browsing activities and, hence, a model of their requirements, preferences and tastes, without any user intervention or a priori knowledge of the specific Web setting. Our approach to personalization conciliates the main benefits of the above studies without suffering from their individual limitations. Specifically, suitable Web usage mining methods permit to infer and regularly update behavioural profiles from raw usage data, without corrective interventions from a human expert being necessary. Cooperation is only required for unknown users when they first access the ITB@NK hyperspace: a concise questionnaire serves to detect visitors' interest in the main content categories. Though potentially disappointing, such a task effectively requires very few clicks and should not be perceived as a deterrent to either ongoing or successive browsing activities. Finally, the adoption of a collaborative approach to refine preference profiles mitigates the effects of poorly answered questionnaires. An overview about methods and techniques for adaptation can be found in [2]. [8] exhaustively surveys recent research efforts in the field of Web usage mining for personalization.

This paper is organized as follows. Section 2 briefly overviews the main functionalities of the ITB@NK system and then provides an insight into the software architecture of its underlying personalization engine. A formal model of the automated adaptation process is proposed in section 3, that also deals with the specific data mining techniques exploited at the different stages of the process. Finally, section 4 draws some conclusions and highlights future developments.

2 A Novel Personalization Engine

ITB@NK is an adaptive front-office that provides personalized access to the contents and services from traditional bank information-systems. Precisely, it provides users with an adaptive Web interface for performing tasks such as online payments and bank transfers, accessing information about their account balances and bank activities, and reading personalized news. ITB@NK exploits a modular and extensible personalization engine, shown in fig. 1. The following analysis focuses on its data sources, main modules and individual components, while section 3 formally deals with the technicalities of its adaptation process.

Data sources. The *Repository* stores visitor clickstream and personal information. It also includes a set of service logs, that keep trace of users' input values to the available Web services. The *Data Warehouse* maintains meaningful information such as visitor browsing sessions, behavioural and interest profiles for disjoint subsets of users, the recent exploitation history of Web services and a suitable index for locating topics scattered on different Web pages. The *CSS/XSL Repository* contains resources detailing the presentation aspects of the hypermedia front-end. XSL shyle sheets contribute to the overall layout of the application front-end, while CSS style sheets specify the presentation properties for front-end elements, such as buttons, scroll bars and hyperlinks.

The personalization modules. The adaptation process relies on three functionalities, namely adaptive content delivery, navigation support and form filling, that are entrusted to as many personalization modules. The *Adaptive Content Delivery* module aims at recommending contents presumably of interest to the visitor. The *DataWarehouse Content Manager* accesses information within the *Data Warehouse* on behalf of the whole module. It also interacts with the *User Preferences* and *Preference Clustering* components. The former holds the preferences of the generic user in specific topics. The latter builds partitions of users with homogeneous interests in the *Data Warehouse*. The *Content Indexer* constructs an inverted table representing the relevance of each topic within any available Web page. The *Content Search* exploits information retrieval techniques to efficiently detect pages with contents related to some query topic.

The *Adaptive Navigation Support* module adapts hyperlinks within the application front-end, in such a way to propose personalized navigation paths. The *DataWarehouse Navigation Manager* reconstructs visitor browsing sessions within the *Data Warehouse*, by applying some preprocessing techniques (i.e. data cleaning, user identification, session identification and page-view identification [6]) to the raw usage data within the *Repository*. The *E. M. Clustering* component forms navigational clusters, i.e. groups of users with similar browsing behaviour, and associates a probabilistic characterization to the navigational behaviour of the users within each such cluster. The *Adaptation Algorithm* implements a suitable technique for tailoring links to visitor browsing profile.

The *Form filler* module improves the usability of Web services by automatically filling in their form fields with input values that best describe the typical exploitation pattern of each individual user. The *DataWarehouse Form Manager* builds suitable *Data Warehouse* tables from the service log information within the *Repository*: any such table stores the recent input values from all visitors to a given Web service. The *Service Descriptor* keeps trace of those services that are subjected to form filling. For each such service, it manages details that specify what fields are actually manipulated by the *Form Filler*. The *Association Miner* computes the most typical set of input values from any user to a Web service.

Two main components. The *Dispatcher* is a core component, since it encodes the control flow of any Web application built on top of the proposed engine. It receives user requests and dynamically associates them with a specific personalization workflow, that depends on a number of characteristics of the request, such as its emitting device and the referred hypermedia resource. The *Dispatcher* coordinates the behaviour of personalization modules and eventually activates

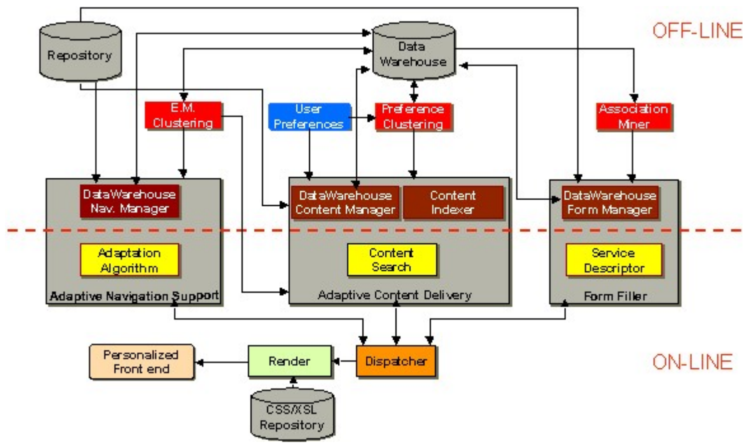

Fig. 1. The software architecture of our personalization engine.

the *Render*. This module assembles the contributions from personalization modules into a response, whose mark-up language and presentation are targeted at the access device from which the initial request was originated.

The architecture in fig. 1 is built around the *Model-View-Controller* (MVC) design pattern [4]. This allows to the separate the three essential aspects of an application, i.e. its business, presentation and control logic with the purpose of considerably reducing both time and costs for development and maintenance. Precisely, the *Dispatcher* acts as the Controller in the MVC design pattern, the *Render* pays the role of the View, while the personalization modules and associated components behave as the Model. Finally, it is worth noticing that the exploitation of powerful tools for presenting hypermedia documents supports personalization on a variety of access channels such as the Web, Wap-compliant mobile phones, and the human voice carried out by the telephone system.

3 A Formal Model for the Personalization Process

Let $\mathcal{W} = \{p_1, \ldots, p_n\}$ represent the hyperspace of a Web application, consisting of n Web pages. In the following we assume that $\mathcal{W}$ denotes the ITB@NK hyperspace, though our treatment can be applied to a variety of Web settings. Our formalization addresses the contents, links and services within any page $p \in \mathcal{W}$.

A term vector v_p is associated to each page $p \in \mathcal{W}$ in order to represent it into a multidimensional Euclidean space. Term vector cardinality coincides with that of the so-called Web site dictionary SD, i.e. a collection of all topics within the hyperspace $\mathcal{W}$. Technically, SD is a vector of unique term stems resulting from the application of traditional information retrieval techniques (such as term extraction, term deletion and stemming [5]) to the pages in $\mathcal{W}$. Also, the generic coordinate $v_p[i]$ indicates the relevance of topic $SD[i]$ within p and is computed by means of the *TFIDF* technique.

Notation $\mathcal{L}_p =< (l_{p,1}, a^x_{p,1}), \ldots, (l_{p,m_p}, a^x_{p,m_p}) >$ indicates the set of m_p non-contextual links [2] within a page p, i.e. all the available links with the exception of those that are anchored to actual contents. Entities $l_{p,j}$ and $a^x_{p,j}$ respectively denote the j-th link in $\mathcal{L}_p$ and a corresponding set of annotations, i.e. suitable presentation information concerning $l_{p,j}$. Our model currently focuses on link color ($c^x_{p,j}$), font-type ($ft^x_{p,j}$) and font-size ($fs^x_{p,j}$). Formally, $a^x_{p,j} = \{c^x_{p,j}, ft^x_{p,j}, fs^x_{p,j}\}$, where index x represents the relevance of a link to a given user. It takes values in the set $\{n, m, h\}$, whose elements distinguish among three degrees of link reputation, that is navigational (n), moderate (m) and high (h): intuitively, a high link reputation requires brighter colors, more readable font types and larger font sizes. For each x, annotations $a^x_{p,j}$ are suitably defined to address particular application requirements. Fig. 2 shows a sample page p. Links within $\mathcal{L}_p =< (l_{p,news}, a^h_{p,news}), (l_{p,markets}, a^n_{p,markets}), (l_{p,stockportfolio}, a^m_{p,stockportfolio}), (l_{p,yourmoney}, a^n_{p,yourmoney}) >$ reveal a prominent interest of the current visitor into news and portfolio management. Presentation guidelines provide that *arial* is the font type for anchor text; links with high, moderate and normal reputation are respectively rendered in red, green and blue; correspondingly, a large (14 pt), medium (12 pt), small (10 pt) font-size is exploited. In the specific case of fig. 2, annotations become $a^h_{p,news} = \{red, arial, 14\}$, $a^n_{p,markets} = \{blue, arial, 10\}$, $a^m_{p,stockportfolio} = \{green, arial, 12\}$, $a^n_{p,yourmoney} = \{blue, arial, 10\}$.

The set of n_p Web services within a page $p \in \mathcal{W}$ is denoted by $\mathcal{S}_p = \{ws_{p,1}, \ldots, ws_{p,n_p}\}$. In particular, the form associated to the generic Web service $ws_p \in \mathcal{S}_p$, namely $\mathcal{F}_{ws,p}$, is a collection $\mathcal{F}_{ws,p} = \{f_{ws,p,1}, \ldots, f_{ws,p,l_{ws,p}}\}$ of $l_{ws,p}$ form fields. A unique Web service, *search*, appears within the page of fig. 2. In such a case, $\mathcal{S}_p = \{search\}$, and $\mathcal{F}_{search,p}$ consists of a sole input field.

Finally, pages in $\mathcal{W}$ allow to catch visitor browsing activities. Assume that $\mathcal{U} = \{u_1, \ldots, u_N\}$ corresponds to the overall user population of $\mathcal{W}$. A set of n_u browsing sessions, $D_u = \{s_{u,1}, \ldots, s_{u,j}, \ldots, s_{u,n_u}\}$, is associated with any visitor $u \in \mathcal{U}$, where the generic session $s_{u,j}$ is defined in terms of a sequence of accesses to pages in $\mathcal{W}$. As a consequence, $\mathcal{D} = \{D_{u_1}, \ldots, D_{u_N}\}$ is the data set with all the interactive sessions from the above N users. The following subsections provide an insight into the distinct phases of the devised personalization approach.

3.1 User Clustering and Behavioural Modelling

These are two off-line steps, that respectively address the problem of finding groups of users with similar navigational habits and learning a behavioural profile for the individuals within each such cluster. Both phases rely on a technique introduced in [3] and performed by the *E.M. Clustering* component. We discuss below how the mentioned technique can be seamlessly exploited within our formulation. The main idea is dividing the original set of N users into K sets, in such a way that the generic visitor $u \in \mathcal{U}$ has probability $p(c_u = k)$ of belonging to cluster k, with $1 \leq k \leq K$ and $\sum_{k=1}^{K} p(c_u = k) = 1$. Note that c_u is a random variable that takes into account the cluster membership of u. The behaviour of users within cluster k is characterized by a parameter set Θ_k, that depends

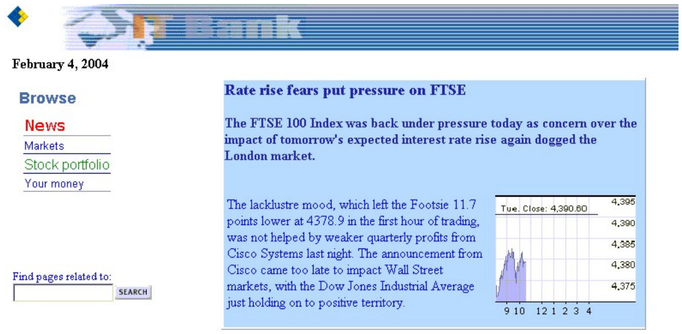

Fig. 2. A sample Web page from ITB@NK hyperspace.

on the specific probabilistic approach adopted to model the navigational activities in that cluster. $\Theta = \{\Theta_1, \dots, \Theta_K\}$ is the set of all cluster parameters. For simplicity, we assume that browsing sessions from individual visitors are independent: though such an hypothesis may not catch some rare behavioural patterns, it works reasonably well in practice. This assumption allows us to adopt a simple, first-order Markov model for the behaviour within individual clusters. Given $p, p_i, p_j \in \mathcal{W}$, for each k between 1 and K, it holds that $\Theta_k = \{\pi_k(p), T_k(p_i, p_j)\}$, where $\pi_k(p)$ is a vector of starting page probabilities and $T_k(p_i, p_j)$ is an $n \times n$ matrix of transition probabilities. From the Markovian definition of a chain of events (i.e. page accesses), the probability of a browsing session $s_{u,j}$ conditioned on cluster membership c_u can be written as follows

$$p(s_{u,j}|c_u = k, \Theta_k) = \pi_k(s_{u,j,1}) \prod_{l=1}^{L_{u,j}-1} T_k(s_{u,j,l+1}, s_{u,j,l}), \quad 1 \leq k \leq K$$

where $s_{u,j,l}$ and $L_{i,j}$ respectively denote the l-th page and the overall number of accesses within $s_{u,j}$. As a consequence, the probability of all navigation sessions D_u, conditioned on u's cluster membership becomes:

$$p(D_u|c_u = k, \Theta_k) = \prod_{j=1}^{n_u} p(s_{u,j}|c_u = k, \Theta_k)$$

Obviously, since there is no a priori knowledge of the cluster to which u belongs, the mixture model below must be exploited:

$$p(D_u|\Theta) = \sum_{k=1}^{K} p(D_u|c_u = k, \Theta_k) \cdot p(c_u = k, \Theta_k)$$

Finally, under the reasonable conjecture that navigation data from distinct users are independent, the likelihood of $\mathcal{D}$ is given by $p(\mathcal{D}|\Theta) = \prod_{j=1}^{N} p(D_{u_j}|\Theta)$. To this point, the problem of clustering and profiling Web users can be simply formalized as that of computing a suitable estimate of the set Θ given $\mathcal{D}$, in such a way to maximize either the maximum likelihood (ML) or the maximum a-posteriori (MAP) likelihood of the data. Formally, it is a question of computing Θ^* such that $\Theta^* = \arg\max_\Theta\{p(D|\Theta)\}$ or $\Theta_{MAP} = \arg\max_\Theta\{p(D|\Theta)p(\Theta)\}$ respectively in the ML or MAP case. The Expectation Maximization algorithm in [3] allows to compute Θ^* from an initial estimate Θ by iterating a sequence of two steps, known as Expectation and Maximization, until at least a local maximum of the ML (or MAP) function is reached. Precisely, for each user $u \in \mathcal{U}$ and each cluster index k (with $1 \leq k \leq K$), the Expectation step computes the class probabilities $p(c_u = k|D_u, \Theta)$, on the basis of the current estimate of the parameters Θ. The Maximization step exploits the previous class probabilities for improving the current estimates of $\pi_k(s)$, $T_k(s_i, s_j)$ for each possible value of k. Three main benefits of the technique in [3] are mentioned next. It addresses the case of a distinct number of browsing sessions for each user: individuals with more sessions have an higher impact on the estimation of cluster transition matrices. On the contrary, most of previous studies in the literature, such as [7], typically assume that each session comes from a different user, thus affecting profile reliability. Moreover, user sessions are treated as sequences of page accesses. This allows to model navigation paths with variable length: [14] investigates the advantages of exploiting sequences rather than vectors for representing navigation sessions. Finally, no practical distinction between clustering and profiling really exists: both phases are performed in a single computational task. On the contrary, conventional approaches such as [7] exploit distinct methods to find and characterize session clusters.

3.2 Adaptive Content Delivery

This functionality is triggered at the time of a request for a Web page p. Here, the aim is at detecting and suggesting a small subset of Web pages in $\mathcal{W}$, whose contents are presumably of interest to the user. Page recommendation takes into account both visitors' location within the hyperspace $\mathcal{W}$ (topics in p should be of some interest to u, having been required) and their explicit preferences. The generic visitor $u \in \mathcal{U}$ is associated with a preference vector pv_u, obtained from her/his questionnaire, whose entries measure u's interest into the corresponding topics within SD. The *Preference Clustering* component finds a partition $\mathcal{C} = \{C_1, \ldots, C_L\}$ of L sets of homogeneous visitor preferences, according to a technique described in [10], that also synthesizes any such cluster C_l into a representative P_l. For each $l \in [1, L]$, representatives are such that $|P_l| = |SD|$ and $P_l[i]$ represents the average interest of all users within C_l in topic $SD[i]$. In the following we denote by P_l^u the representative of the interest cluster to which user u belongs. The *DataWarehouse Content Manager* merges the preference vector pv_u with P_l^u. Such a process of *collaborative refining* is conceived to improve the estimates of user tastes, when poor knowledge (i.e. sparse data) about their actual preferences is available. The resulting preference vector pv'_u is defined as:

$$pv'_u = \begin{cases} pv_u[i] & \text{if } pv_u[i] \neq 0 \\ \alpha_u \cdot P_l^u[i] & \text{otherwise} \end{cases} \quad 1 \leq i \leq |SD|,\ 0 < \alpha_u < 1$$

where α_u is a damping factor for the relevance of those interests that are likely to be shared by u, being borrowed from other individuals with partly overlapping tastes. Intuitively, for each user $u \in \mathcal{U}$, constants α_u represent a degree of correlation between pv_u and P_l^u. So far, u's general preferences have been addressed. Those related to her/his browsing activities are simply represented by the term vector v_p, that is associated to the required page p. Given an interest threshold τ, the *Content Search* component dynamically yields $\mathcal{R} = \{p' \in \mathcal{W} | \alpha_1 \cdot sim(v_{p'}, v_p) + \alpha_2 \cdot sim(v_{p'}, pv'_u) > \tau\}$, a set of recommendations whose top-N pages are embedded within p (generally, values of N range from 2 to 4). Heterogeneity in recommendation topics helps users accomplish a number of distinct tasks within an interactive browsing session. In the above definition, constants α_1 and α_2 determine the degree of recommendation adherence respectively to contents in p and u's general preferences. Note that constants α_1, α_2 are such that $0 \leq \alpha_1, \alpha_2 \leq 1$ and $\alpha_1 + \alpha_2 = 1$. Function *sim* is the cosine similarity [5]. Finally, α_1, α_2 and τ are empirically evaluated.

3.3 Adaptive Navigation Support

The *Adaptation Algorithm* component performs link annotation [2] to indicate appropriate browsing paths to the user. It dynamically changes some visual cues of links such as their color, font-type and font size to reflect their relevance to the visitor. If k represents the index of the visitor cluster to which the generic user u belongs with highest probability and p denotes the Web page required by u, the annotation of the m_p navigation alternatives in $\mathcal{L}_p$ requires detecting a suitable value assignment to the index x of the term $a^x_{p,j}$, for each j between 1 and m_p. Let us indicate with p' the page pointed by the generic link $l_{p,j}$ and with τ^n and τ^m two probability thresholds. $l_{p,j}$ acquires a low relevance to u if $\pi_k(p, p') < \tau^n$, that is if u has a low probability of accessing p' from p: in such cases, $l_{p,j}$ is annotated with $a^n_{p,j}$. Conversely, annotations $a^m_{p,j}$ and $a^h_{p,j}$ are a consequence of higher transition probabilities: a moderate (resp. high) relevance is attributed to $l_{p,j}$ if it holds that $\tau^n < \pi_k(p, p') \leq \tau^m$ (resp. $\pi_k(p, p') > \tau^m$). The advantage of link annotation is that it does not alter the layout order of links and avoid the issues arising with incorrect mental maps [2].

3.4 Adaptive Form Filling

This functionality dynamically fills in the form fields of a Web service with the values that the user typically inputs. The *Service Descriptor* keeps trace of the services in $\mathcal{W}$ that are subjected to form filling. In particular, for each such service ws_p within a Web page $p \in \mathcal{W}$, a sequence of indexes, i_1,i_2,…,i_h (with $1 \leq i_1 < i_2 < \ldots < i_h \leq l_{ws,p}$), is leveraged to identify a meaningful subset of $\mathcal{F}_{ws,p}$, namely $\mathcal{F}'_{ws,p} = \{f_{ws,p,i_1}, \ldots, f_{ws,p,i_h}\}$, that contains the form fields of ws_p to be filled in. Let y_{u,ws_p,i_l} represent the input from a user $u \in \mathcal{U}$ to the field $f_{ws,p,i_l} \in \mathcal{F}'_{ws,p}$. An interaction of u with ws_p can be modelled as an itemset $is_{u,ws_p} = \{y_{u,ws_p,i_1}, \ldots, y_{u,ws_p,i_h}\}$. The *Association Miner* performs the Apriori algorithm [1] for mining c-itemsets (i.e. itemsets with cardinality c) from the interactions of an individual. Formally, the problem of identifying a set of input values, that closest resembles the typical interaction of u with ws_p, reduces to the computation of a c-itemset is^*_{u,ws_p} for suitable values of its support and cardinality. Support guarantees a certain degree of input regularity. Cardinality indicates a minimum number of fields in $\mathcal{F}'_{ws,p}$, that is^*_{u,ws_p} must cover.

4 Conclusions and Future Developments

The main contribution of our work is the introduction of a novel personalization engine that combines two major functionalities, namely adaptive content delivery and navigation support, with adaptive form filling, a feature for enhancing user interaction with Web services. The modularity and flexibility of the devised framework make the overall personalization process parametric w.r.t. the adaptation methods: the above functionalities can be independently implemented by choosing an ad-hoc suite of data mining techniques, that best catch the peculiarities of a given environment. From this perspective, ITB@NK is a meaningful customization conceived for the context of Internet Banking. Such a customization is formally investigated. Experiments are being performed to measure the effectiveness of the ITB@NK adaptation process in a somehow objective manner, that accounts for subjective perception of personalization. Future developments include three main lines of research. First, the automatic detection of user interests. Here, the idea mainly consists in exploiting the term stems within recently accessed Web pages as a model of visitors' current interests and requirements. Second, the enhancement of navigation support, to the purpose of applying annotation and sorting respectively to contextual and non-contextual links [2]. Dynamic link generation may be exploited as a means for shortening navigation paths through the hyperspace. Third, an experimental evaluation of the clustering approach in [14], that employs textual and usage proximity to catch similarities even between pairs of Web pages with heterogeneous contents, such as text and images. This may bring to a more reliable characterization of the browsing behaviour within session clusters w.r.t. to the technique in [3].

5 Acknowledgments

This work has been developed within the framework of the project ITB@NK, which is carried out by Carisiel, in collaboration with ICAR-CNR and DEIS-UNICAL, and is partially funded by a MIUR grant. The authors wish to express their gratitude to Dr. Giuseppe Manco, Dr. Francesco Natale and to all the members of the ITB@NK project for their precious cooperation.

References

1. R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. of Int. Conf. on Very Large Data Bases*, pages 487–499, 1994.

2. P. Brusilovsky. Adaptive hypermedia. *User Modeling and User Adapted Interaction*, 11(1-2):87–100, 2001.
3. I. V. Cadez, S. Gaffney, and P. Smyth. A general probabilistic framework for clustering individuals and objects. In *Proc. of the ACM-SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 140–149, 2000.
4. S. Ceri, P. Fraternali, A. Bongio, M. Brambilla, S. Comai, and M. Matera. *Designing Data-Intensive Web Applications*. Morgan-Kaufmann, 2002.
5. S. Chakrabarti. *Mining the Web Discovering Knowledge from Hypertext Data*. Morgan-Kaufmann, 2003.
6. R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Knowledge and Information Systems*, 1(1):5–32, 1999.
7. B. Mobasher et al. Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization. *Data Mining and Knowledge Discovery*, 6(1):61–82, January 2002.
8. D. Pierrakos et al. Web usage mining as a tool for personalization: A survey. *User Modeling and User-Adapted Interaction*, 13(4):311 – 372, 2003.
9. J. Herlocker, J. Konstan, A. Borchers, and J. Riedl. An Algorithmic Framework for Performing Collaborative Filtering. In *Proc. of Conf. on Research and Development in Information Retrieval*, 1999.
10. Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304, 1998.
11. T. Joachims, D. Freitag, and T.M. Mitchell. Web Watcher: A Tour Guide for the World Wide Web. In *Proc. of Int. Joint Conf. on Artificial Intelligence, (IJCAI97)*, pages 770–777, 1997.
12. H. Lieberman. Letizia: An Agent that Assists Web Browsing. In *Proc. of Int. Joint Conf. on Artificial Intelligence (IJCAI 95)*, pages 924–929, 1995.
13. U. Manber, A. Patel, and J. Robison. Experience with personalization on Yahoo! *Communications of the ACM*, 43(8):35–39, 2000.
14. G. Manco, R. Ortale, and D. Saccà. Similarity-based clustering of web transactions. In *Proc. of ACM Symposium on Applied Computing*, pages 1212–1216, 2003.
15. B. Mobasher, T. Luo H. Dai, and M. Nakagawa. Improving the Effectiveness of Collaborative Filtering on Anonymous Web Usage Data. Technical Report 00-005, Computer Science Department, De Paul University, 2001.
16. D.S.W. Ngu and X. Wu. Sitehelper: A Localized Agent that Helps Incremental Exploration of the World Wide Web. In *Proc. of Int. World Wide Web Conference*, 1997.
17. M. Perkowitz and O. Etzioni. Adaptive Web Sites: Automatically Synthesizing Web Pages. In *Proc. of Nat'l Conf. on Artificial Intelligence*, pages 727–732, 1998.