

An adaptive system for the personalized access to news

Liliana Ardissono, Luca Console and Ilaria Torre
Dipartimento di Informatica, Università di Torino
Corso Svizzera 185 - 10149 Torino (Italy)
E-mail: {liliana,lconsole,ila.torre}@di.unito.it

Personalization is one of the keys for the success of web services. In this paper we present SeAN (Server for Adaptive News), an adaptive system for the personalized access to news servers on the WWW. The aims of the system are (i) to select the sections (topics) and news in the server that are most relevant for each user, (ii) to customize the detail level of each news item to the user's characteristics and (iii) to select the advertisements that are most appropriate for each page and user. In the paper we discuss the functionalities of the system and we present the choices we made in its design. In particular, we focus on the techniques we adopted for structuring the news archive, for creating and maintaining the user model and for generating the personalized hypertext for browsing the news server.

Keywords: User modeling, adaptive systems, news servers on the World Wide Web

1. Introduction

The possibility of providing customized services [45], i.e., services tailored to the needs and preferences of the individual user (customer) has been the focus of communication and marketing strategies for several years and the Web seems now to offer a unique opportunity for implementing, testing and improving these strategies. At the same time, the Web (and more recently the use of other media for personal communication) is giving new impulse to the research in the fields of user modeling and user-adapted interaction [25]. The personalization of the interaction with the user is, in fact, one of the main goals in the design of web services. This goal requires that the software agent providing the service has the capabilities of maintaining a *model of the user* containing data about her needs, interests and preferences (see [48, 36, 19, 34]) and of us-

ing strategies for adapting its behavior to each specific user (see [8, 10, 35]).

Adaptive systems have been designed and applied in different areas, such as intelligent tutoring [12, 15, 23, 28], access to information sources [40, 26, 16, 1, 18, 39, 49], electronic catalogues [42, 13, 38, 31, 20], health-care assistance [29, 7, 17], information filtering and recommender systems [33, 4, 43, 6, 9, 5, 27, 21]. These systems make use of different techniques as regards both user modeling and adaptation. For example, a model of a user can be generated after classifying her into stereotypical pre-defined classes (as in [44]) or using learning techniques after observing her behavior (as in [41, 46]) or looking for similarities between a new user and past users (as in collaborative filtering [32, 21]). Similarly, different forms of adaptation and adaptation strategies can be defined (see the overviews in [10, 11, 35]). For example, one may personalize the form or the content of the interaction or the navigation paths, and each choice implies other choices and strategies. The choice of such strategies depends on many factors such as the application task and domain, the goals of the system, the target users and the context (see the discussion in [47]).

The personalized provision of news is becoming an increasingly important application for adaptive systems. This is, in fact, an interesting opportunity for all companies operating on the web, not only for those in the communication field (such as newspapers, radio or television companies). Almost all portals provide access to news and also many companies provide specialized news services, usually related to their commercial fields or to the interests of their customers. The primary goal of these services is to attract web surfers and to gain their loyalty.

Tailoring information retrieval and information filtering is one of the keys for the success of these services. In fact, a personalized service enables a user to find easily the pieces of information that she needs or that are of interest for her; this is the added value that can lead her to visit a site frequently. Indeed, information retrieval is one of the main areas of application for

personalization techniques. Simple forms of personalization are implemented in many sites (based on preferences directly expressed by the users), while several more complex approaches have been proposed in the recent literature (e.g., [9, 30, 13, 24]).

In this paper we focus on the generation of adaptive hypermedia for accessing on-line electronic news servers, and we describe SeAN (Server for Adaptive News) a system that we designed for such a task. SeAN is a multi-agent system which can be accessed using any web browser; it aims at personalizing both the selection of topics that are of interest to the user and the detail level of the presentation of each news item. This combination of goals makes SeAN different from other approaches in the literature. This paper describes the system and the choices we made in its design. It is organized as follows. In Section 2 we discuss the goals and the functionalities of the system and the requirements for its design. We then focus on three main aspects: the organization of the news archive (Section 3), the definition of appropriate user modeling techniques (Section 4), the design of strategies for selecting what has to be presented to the user (Section 5). For each one of these aspects we motivate the choices that we made, comparing them with alternative ones. Section 6 outlines some initial experiments we made for testing the system while Section 7 provides conclusions.

2. Functionalities, requirements and architecture of SeAN

In SeAN we focused on three forms of personalization for supporting the access to an archive of news:

- The system should select only the topics (i.e., the sections of the archive) and news that are of interest for the user, leaving the others aside, as information which can be reached on demand.
- The system should adapt the detail level of the presentation of each news item to the user's interests and expertise.
- The advertisements that are inserted in the presentation should be appropriate for the user and for the context in which they are inserted (i.e., the topic of the news item presented in the page).

As a first requirement, we imposed that these forms of personalization should be provided to both first time and frequent users. As a consequence, SeAN must be able to generate an initial (possibly approximate) model of the user when she connects to the server for

the first time and fills in a registration form (including a small set of questions, concerning personal and demographic data). The system must then be able to revise the model of the user, making it progressively closer to her actual characteristics and following her possible changes of interest. In order to do that, the system must be able to track the user's behavior.

As a second requirement we decided that the system should never impose its choices on the user. The system's decisions should be always modifiable by the user, who should have the possibility of changing the set of sections and news selected by the system and their detail level. This is important for making the system more user-friendly and acceptable. Moreover, as we shall see in the following, the fact that the user can make changes is a powerful handle for supporting user modeling activities, allowing the system to get information about the user's actual characteristics and interests.

As a consequence, all the pages generated by SeAN contain buttons for modifying the choices made by the system. For example, Figure 1 shows the personalized Home Page containing the sections selected by the system, according to the user model: in the example, the selected sections are "Economia" ("Economy"), and "Politica" ("Politics"), while other sections, such as "Sport", are omitted. The name of each section is a link that opens the corresponding page. Using a menu and the button "Elimina" ("Delete"), the user can delete each one of the sections selected by the system; conversely, using a second menu and the button "Aggiungi" ("Add") the user can list the sections that were not considered by the system and add them to the page (see the bottom part of the screenshot in figure 1). Figure 2, on the other hand, shows a page containing the personalized presentation of a news item. Only some pieces of information associated with the news item are displayed; they can however be removed by the user, using the button "delete". The other pieces of information are shown only as a link that can be opened on demand.

The goals and the desired functionalities discussed above imposed some requirements on the design of SeAN, which is conceptually organized as a three-tier architecture, as shown by the scheme in Figure 3. The user can access the server using any web browser (first tier). The web browser interacts with a set of agents which are responsible for user-modeling and personalization activities (second tier) and which can access information maintained in a set of databases (third tier). The interaction with the user is managed by a "user in-



Fig. 1. Personalized Home Page

teraction agent” which is responsible for collecting the requests from the user and for sending web pages to the browser. User modeling is carried on by two agents: one is activated only when a user connects to the server for the first time and is responsible for creating an initial model, starting from the data provided by the user and using a library of stereotypes. The second is activated at the end of each session and is responsible for revising the user model, given the events summarizing the user’s behavior during the session and given a set of user modeling rules regarding such events. The models of the users are maintained in a database and thus the model of a specific user can be retrieved each time the user connects to the server.¹ Two other agents are in charge of extracting, respectively, from the news database and from the advertisement database, the specific pieces of information to be presented to a user, given her model. The pieces of information are then

¹The user models are permanently stored only after the user’s permission.

passed to the “interaction agent”, which generates the web pages and sends them to the user’s browser.

In the next sections we describe the system in more detail, focusing on three main groups of problems:

- The organization of the archive of news, which should support the personalized selection of topics and news and the presentation of each news item at different levels of detail.
- The user modeling agents, which should be able to capture different features (characteristics or dimensions) of the user in order to achieve the forms of personalization discussed above.
- The agents for the adaptive selection of news (and selection of the appropriate detail level) and for the generation of hypertextual pages.

3. Organization of content in the news server

In this section we analyze the choices we made in the design of the databases for news and advertisements ((1) in figure 3) .

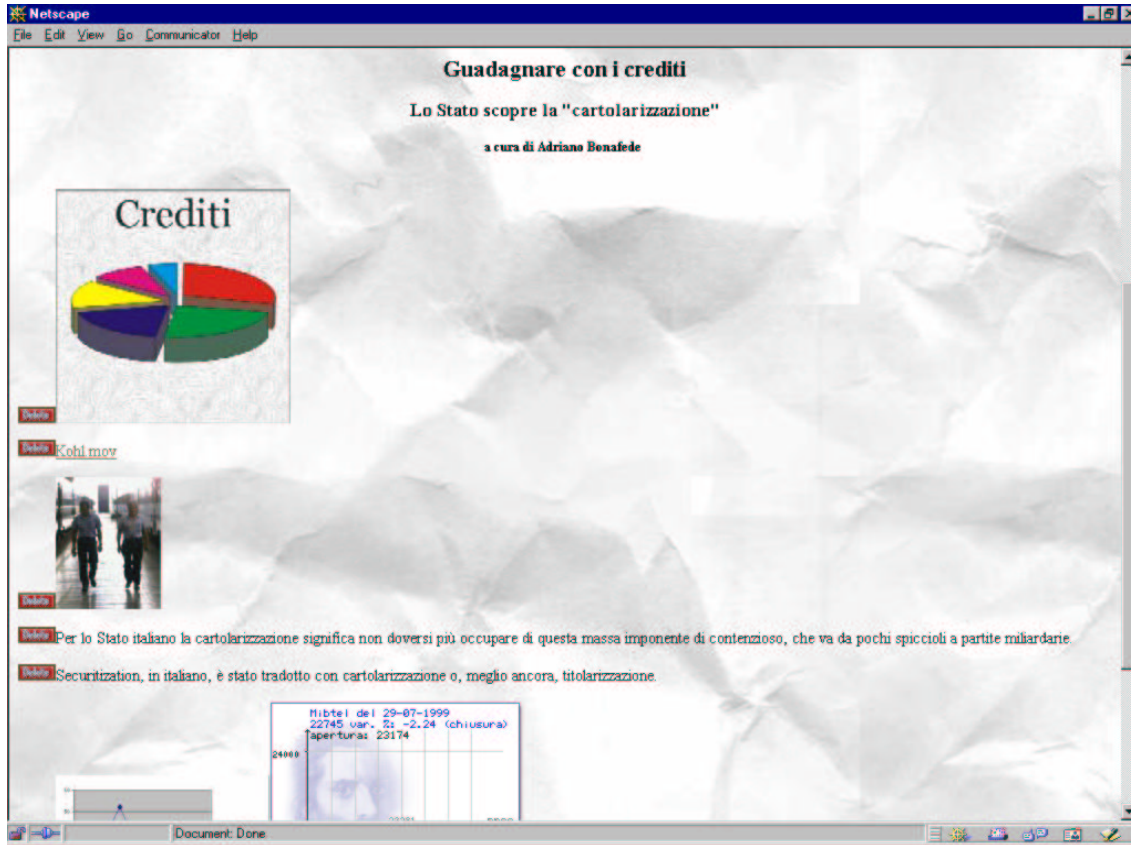


Fig. 2. Personalized News item

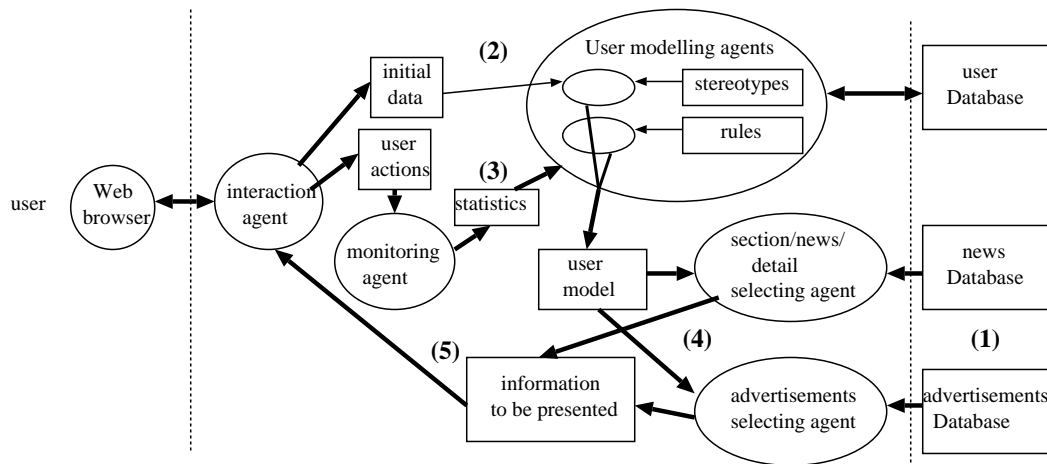


Fig. 3. Architecture of the system.

3.1. News Database

This database contains all the news that can be presented to the user. In particular, given the goals concerning personalization discussed in the previous section, the database must support the possibility of (i) classifying news according to their topics and (ii) generating different presentations (varying the detail level)

cerning personalization discussed in the previous section, the database must support the possibility of (i) classifying news according to their topics and (ii) generating different presentations (varying the detail level)

of each news item.

The first problem can be tackled in different ways. For example, a first option is to classify news according to an a-priori hierarchy of topics. This is the choice made in most of the horizontal portals and in (traditional and electronic) newspapers, where the hierarchy has a correspondence in the structure of the editorial board. As a consequence, this is also the choice of some recent editorial systems which aim at supporting the work of editorial offices. A second option is to generate the classification dynamically, using indices extracted from each news item. This is the technique adopted in most of the approaches to information retrieval. The two options are not incompatible and could be used together (as in many search engines on the web). The former can provide a high-level classification of news into a hierarchy of sections; the latter could provide a finer grained classification within each specific section.

In SeAN we decided to focus on the first option, which is common also to other systems, such as [6]. We defined a hierarchy of sections and subsections and news are classified according to this hierarchy. More specifically, we considered high-level sections such as politics (which has subsections such as international and national politics; the latter includes further subsections concerning different local communities), economy (with a structure similar to that of politics), sport (with subsections for specific sports), technology, culture and so forth.

There is a number of reasons for this choice. First of all, our system is aimed at providing a personalized electronic version of classical newspapers, integrated with the editorial software systems used in newspapers. This makes the organization into a hierarchy of sections a natural choice. Second, the hierarchical classification including high level topics allows us to provide some form of personalization even when limited information about the user is available (as it is often the case with first time users). In this case, in fact, it is difficult to make selection of topics at a finer grain than the one corresponding to high level sections in the hierarchy. Finally, as we shall see, the main focus of our work is the personalization of the detail level of the presentation. This is the aspect to which we paid most of our attention and for which we propose an innovative approach.

Most of the approaches in the literature assume that the repository of news is not structured, i.e., that each news item is a text file (or a set of text files). This makes the administration of the repository very sim-

ple and, moreover, makes the approaches very flexible since the techniques adopted for personalization (personalized filtering) can be applied to any repository of text files. However, in this way it is difficult to define strategies for personalizing the presentation of a document. Basically, the choice is binary: either a document is retrieved and shown to the user (possibly with some labels specifying the estimated interest level for the user) or it is not. On the other hand, we aim at structuring news in such a way that different detail levels for the presentation of a news item can be obtained dynamically depending on the content of the user model.

Clearly, there are different ways of modifying the detail level for presenting a news item. One expensive (and usually not practical) approach is to store different versions of the news item. Alternatively, one could insert annotations in the text of the news item (using some markup languages) and then produce different versions by interpreting the annotations. A more complex approach is to use techniques for automatically generating summaries from detailed news; see e.g., [22]. This approach requires the availability of a lot of knowledge on the topic of news and can be adopted only if news refer to specialized domains.

The use of annotations seems to be an interesting option and indeed our approach moves from this idea and takes it one step further. News are viewed as complex composite entities, having several attributes that define their components. For example, we considered attributes such as: title and subtitle; author(s); abstract; text; graphics summarizing the content of the text; photos, video, audio clips; commentaries, interviews, agency reports; raw data and/or detailed charts/graphics, and so forth. Some of the attributes are optional or can be multi-valued (e.g., more than one photo or clip may be associated with a news item); moreover, the same object (e.g., a photo or a clip) may be associated with more than one news item.

Each news item is thus regarded as a chunk of information, concerning e.g., an event or a situation to be conveyed to the user. For such a reason, pieces of information concerning the same news item (e.g., an event) may be available on different days, possibly with different attributes. Therefore, the database is an historical one, storing information concerning several days.

For example, let us consider a news item concerning the financial status of the company ACME (this item belongs to the section “economy”). Besides the title and a text, one may have other attributes such as graphs describing the financial status of ACME, an interview with the CEO of the company, interviews with finan-

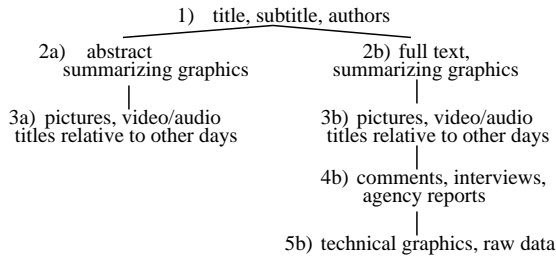


Fig. 4. Detail levels in the presentation of news

cial experts (some of the interviews may possibly have the form of a video or audio clip), agency reports concerning the company. Moreover, we can regard “Financial Status of Company ACME” as a news item for which different pieces of information may be available on different days: for example, one may have a report on one day and interviews on the day after. There may be also other news concerning ACME in the news server. For example, if ACME is a hi-tech company, one could have a news item in the technology section concerning the launch of a new product. The interview to the CEO could be associated also with this news item.

The main advantage of our approach is that different detail levels for the presentation of news can be obtained as different aggregations of their attributes. For example, at a low detail level one may include only the title and abstract; a more detailed level may include also the text (possibly replacing the abstract); an even more detailed level may include also graphs or commentaries or interviews. Moreover, as most of the attributes convey complementary information (e.g., pictures complement textual information), there is a certain flexibility in the combination of attributes which can be presented together.

In SeAN we defined a fixed number of alternative aggregations (called *presentation formats* or *presentation levels*) and we introduced a partial order between them, corresponding to the detail level provided by the aggregations. The formats and ordering are shown in Figure 4, where each node adds detail to that one provided by its ancestor. For example, level 1 can be made more detailed in two different ways, either by adding the abstract of a news item or the full text.

The main problem with our approach is that it requires more efforts from the manager of the news server. In fact, filling in the repository requires adding the news items to the database and thus decomposing information according to the attributes of news. This effort, however, is not very different from the work re-

quired in the editorial offices of newspapers and required by the software systems used in these offices.

The approach we propose is conceptually similar to the idea of annotating news but it introduces a more complex form of semantic decomposition of news, which involves also other pieces of information associated with a news item, such as pictures, clips, graphics but also commentaries, interviews, agency reports etc.. In this sense we believe that our approach is more powerful. Annotations, however, may support a finer variation of the detail level, allowing the definition of different textual presentations of the same news item. In principle our approach and the use of annotations could be integrated. Annotations could allow the definition of a whole range of variations for presenting the text, thus introducing many alternatives in the tree of Figure 4.

3.2. Advertisements Database

Personalizing the advertisements that are inserted in the pages is particularly important since advertising is the main (and in some cases the only) income of companies providing services on the web. The selection of the appropriate advertisement should depend on at least two aspects: (i) the user and (ii) the content of the page in which the advertisement is added.

For these reasons we introduced a second database which contains all the advertisements (usually some sort of banner) that can be inserted into the pages. Each advertisement has two attributes, which are related to the criteria mentioned above for the selection:

- (i) a target, i.e., the segment(s) of population it is directed to;
- (ii) a topic, in relation to the sections of the news server.

4. User Modeling

The user model plays a central role to achieve the personalization goals discussed in Section 2. These goals require that very different features (characteristics) of a user are modeled. For example, the selection of the sections to be displayed is related to the user’s interests and expertise in the topic of each section. The selection of the detail level for news depends also on the user’s cognitive characteristics and in particular on her receptivity, which represents the amount of information that she can read and acquire. The selection of the advertisements depends on the user’s life style and

priorities. This means that all these different characteristics (dimensions) of a user must be modeled in order to achieve the goals above. This can be done in different ways. The simplest one would be the creation of a knowledge base that captures all these dimensions together. This approach, however, is not easy and, in our view, is not the right way to tackle the problem.

In the following we propose an approach in which a user model is divided into a set of dimensions which correspond to different conceptual characteristics of a user and then we decompose the user modeling process along these dimensions. Indeed the application we are presenting in this paper is one of the case studies of a more general project which aims at defining a conceptual framework for user modeling [47]. In the framework we aim at creating an ontology for user modeling and at defining guidelines for deciding which dimensions are relevant, given specific adaptation and personalization goals and given a specific application task and domain. What we aim at is a modular, compositional approach to user modeling in which the different dimensions are managed separately. This has at least two main advantages. On the one hand, it makes user modeling simpler: the knowledge bases for creating and revising the models are simpler and easier to build and, in particular, the combinatorics that may arise by looking at all the dimensions together are avoided. Second, the different dimensions (and therefore the knowledge bases for dealing with them) may be re-used across different application tasks and domains. We shall return later to discussing these points.

Let us come back to the news server application. In this case we considered four separate dimensions in user modeling:

- **Interests:** This dimension captures the user’s interests in the topics corresponding to the sections in the database of news.
- **Expertise:** This dimension captures the expertise of the user on the topics corresponding to the sections in the database of news.
- **Cognitive characteristics:** This dimension defines parameters corresponding to the cognitive capabilities of the user which allow the system to determine the amount and detail of the information that she can catch and retain.
- **Life Style:** This dimension captures the user’s preferences/priorities corresponding to different life styles.

Separate knowledge bases are used to create and manage the part of the user model corresponding to each dimension.

We can turn now to discussing the approach we defined for user modeling more precisely. In our application we consider two separate phases. The first one regards the creation of an initial model for a user connecting to the server for the first time ((2) in figure 3). The second one regards the revision of the model, by tracking the user’s behavior to identify in a direct and more reliable way her interests/preferences/goals and, possibly, the changes about them during time ((3) in figure 3). We decided to keep the two phases separated, using different user-modeling techniques. More specifically, the creation of the initial model is based on the use of stereotypical knowledge, i.e., on the a-priori definition of classes of users (along each dimension). The second phase is based on user modeling rules that are activated by the events captured by tracking the user’s behavior while she is browsing the news.

4.1. Using stereotypes for initializing the user model

The creation of an initial model for a first-time user is based on the use of stereotypes [44], representing the features of classes of readers/customers. The data used in the classification are elicited from the user in a registration form containing a small set of questions, i.e.: age and gender; education level and field; type and field of job; whether her access to the news server is for work or not; her hobbies or priorities (how much she likes activities such as traveling, doing sport, shopping, etc.). In this way, filling in the form is not too demanding and the system is not too intrusive, even if the limited set of data does not allow the system to make a very precise classification of the user. The user may also decide to skip the form; in this case we consider a “default” user.

According to the conceptual decomposition of user modeling discussed above, we consider separate sets (families) of stereotypes for each one of the four dimensions that constitute the user model (see [2]). The stereotypes use the data provided by the user in the registration form as classificatory information and make predictions on different features of the user. While there is a partial overlap between the classificatory data used by different families of stereotypes, the predictions are not overlapping. Thus, the user can be classified independently in each family and the predictions can be merged to define the initial user model.

More precisely, the four families of stereotypes are structured as follows:

- **Interests.** Starting from classificatory data such as the age, gender, type and field of job, purpose of the access to the server and hobbies, these stereotypes make a first prediction on the user’s interest level in the topics corresponding to the sections of the news servers.
- **Domain expertise.** Starting from data such as the degree and type of education and field of job, these stereotypes make predictions on the user’s expertise on the topics corresponding to the sections of the news server.
- **Cognitive characteristics.** This group of stereotypes makes a prediction on the user’s receptivity, a parameter used to determine the amount and detail of the information that can be presented to her. These stereotypes use classificatory data such as the user’s age, education level, job and frequency of access to the Web.
- **Life styles.** These stereotypes classify users according to their psychographic features, which include socio-demographic data and priorities (we based our classification on [14], and updates concerning recent years). They do not make specific predictions: the relevant prediction is the class to which a user belongs. The classes in this family correspond to the targets that can be associated with the advertisements in the database of section 3.2.

Let us now analyze the structure of the stereotypes and the approach we adopted for classifying a user and consequently for generating the predictions that constitute her initial model. Two examples of stereotypes are reported in Figures 5 (a stereotype in the family *interest* describing financial professionals) and 6 (a stereotype in the family *life styles*).

In the stereotypes we explicitly distinguish two groups of slots: *classification* slots and *prediction* slots (as noticed above, stereotypes in the family *life styles* have classification slots only). The structure of the slots is the same:

- Each slot corresponds to a feature F_i of the user (e.g., age, gender, . . . , interest in economy, politics, . . . in Figure 5).
- For each feature F_i we then consider a set of linguistic values $\{v_{i1}, \dots, v_{ik}\}$.
- A numeric value $x_{ij} \in [0, 1]$ is associated with each linguistic value v_{ij} of F_i . This number can be regarded as the probability that $F_i = v_{ij}$, given that the user belongs to the stereotype (i.e., $p(F_i = v_{ij} | S_k)$).

This means that the numeric value x_{ij} measures the frequency of $F_i = v_{ij}$ for the individuals belonging to the stereotypical class and thus it is a measure of the compatibility of $F_i = v_{ij}$ with the stereotype. For example, the slot “age” in Figure 5 specifies that 10% of the financial professionals are between 20 and 25 years old, while 20% are between 26 and 35.

In the classification slots this value can be used to compute a degree of match between a user and a stereotype.

In the prediction slots, the probability x_{ij} can be regarded as a measure of the strength of the prediction. For example, in Figure 5 (stereotype belonging to the *interest* family), the first slot specifies that, for a user belonging to the class “Financial professional”, the probability that her interest in economy is high is 1; the second slot, on the other hand, specifies that the probability her interest in politics is high is only 0.7.

We can now turn to discussing how a user is classified and how the predictions can be generated.

4.1.1. Classification of users

The classification is performed separately for each family of stereotypes. In each family, the goal is to determine a degree of match $M(S_k)$ between the user and each stereotype S_k in the family, thus ranking the stereotypes in each family. This corresponds to computing the probability of the stereotype, given the values of the classificatory features provided by the user in the registration form. For each feature F_i , the corresponding slot provides the probability x_{ij} that $F_i = v_{ij}$, given the stereotype S_k . In order to compute the degree of match $M(S_k)$ we assume that the features are independent; this is reasonable since our goal is to rank the stereotypes in the same family and these stereotypes contain the same set of classificatory slots (features). Moreover, as a stereotype is the result of the conjunction of all the features, we compute the degree of match of the user with a stereotype as the product of the contributions of all the individual slots. In summary, we have:

$$M(S_k) = p(S_k | F_1 = v_{1j_1}, \dots, F_r = v_{rj_r}) = p(S_k | F_1 = v_{1j_1}) * \dots * p(S_k | F_r = v_{rj_r})$$

These conjuncts can be easily obtained using Bayes’ rule, i.e.:

$$p(S_k | F_i = v_{ij_i}) = \frac{p(F_i = v_{ij_i} | S_k) * p(S_k)}{p(F_i = v_{ij_i})}$$

starting from the values $x_{ij_i} = p(F_i = v_{ij_i} | S_k)$ and an a-priori probability distribution $p(S_k)$ of the stereo-

FINANCIAL PROFESSIONAL:**profile:**

age: <20: 0; 20-25: 0.1; 26-35: 0.2; 36-45: 0.3; 46-65: 0.3; >65: 0.1

gender: Male: 0.8; Female: 0.2

job: manager: 0.57; self-trader: 0.3; self-employed: 0.05; ...; student: 0.01

job field: {financial, banking insurance}: 0.8; {politics, law, civil services}: 0.14; ...

reason of connection: work: 0.9; personal: 0.1

hobbies - theatre: a lot: 0.1; some: 0.3; a little: 0.4; not at all: 0.2;

hobbies - watching sports: a lot: 0.4; some: 0.3; a little: 0.2; not at all: 0.1;

...

predictions on interests:

economy: high: 1; medium: 0; low: 0; null: 0

politics: high: 0.7; medium: 0.3; low: 0; null: 0

sport: high: 0.2; medium: 0.4; low: 0.3; null: 0.1

culture: high: 0; medium: 0.2, low: 0.5; null: 0.3

technology: high: 0; medium: 0.3; low: 0.6; null: 0.1

...

Fig. 5. An example of a stereotype in the family *interests*.

ADULT SUPERIOR COMMITTED:**profile:**

age: <26: 0; 26-35: 0.1; 36-45: 0.4; 46-65: 0.4; >65: 0.1

gender: Male: 0.5; Female: 0.5

education: primary school: 0; secondary school: 0.2; university: 0.8

education field: economy: 0.1; {politics, law, sociology}: 0.5; {medicine, biology}: 0.05, scientific: 0.05, human sciences: 0.3...

hobbies - theatre: a lot: 0.3; some: 0.5; a little: 0.2; not at all: 0;

hobbies - traveling: a lot: 0.5; some: 0.4; a little: 0.1; not at all: 0;

hobbies - body care: a lot: 0; some: 0.3; a little: 0.5; not at all: 0.2;

hobbies - shopping: a lot: 0; some: 0.2; a little: 0.8; not at all: 0;

hobbies - doing sport: a lot: 0; some: 0.1; a little: 0.3; not at all: 0.6;

...

Fig. 6. An example of a stereotype in the family *life styles*.

types in each family, estimated in the population of readers². A final set of degrees of match for the stereotypes in a family is obtained after a normalization of the values $M(S_k)$ computed using the equation above. For example, let us consider the following (partial) description of a user:

$$age = 46-65$$

gender = Male
job = manager
job field = financial
reason of connection = work
hobbies: theatre = some,
 ...

Moreover, let us assume that the a-priori probability of the stereotype is

$$p(\text{financial professional}) = 0.2$$

We have the following computation for the degree of match with the stereotype “Financial Professional” of Figure 5:

²The a-priori probability distribution of the values v_{ij_i} (i.e., $p(F_i = v_{ij_i})$) is not needed because the stereotypes in a family contain the same set of classificatory slots and thus this can be regarded as a normalizing factor.

$$\begin{aligned}
M(\text{financial professional}) &= \\
&= p(\text{financial professional} \mid \text{age} = 46 - 65, \\
&\quad \text{gender} = \text{Male}, \dots) = \\
&= p(\text{financial professional} \mid \text{age} = 46 - 65) * \\
&\quad p(\text{financial professional} \mid \text{gender} = \text{Male}) * \\
&\quad \dots = \\
&= (p(\text{age} = 46 - 65 \mid \text{financial professional}) \\
&\quad * p(\text{financial professional})) * \\
&\quad (p(\text{gender} = \text{Male} \mid \text{financial professional}) \\
&\quad * p(\text{financial professional})) * \dots = \\
&= (0.3 * 0.2) * (0.8 * 0.2) \dots = \\
&= 0.00000015
\end{aligned}$$

Let us assume that there is only another stereotype in the family concerning the interests:

$$M(\text{young reader}) = 0.00000005$$

The normalization applied to the two stereotypes of this family produces the following result:

$$\begin{aligned}
M(\text{financial professional}) &= 0.75 \\
M(\text{young reader}) &= 0.25
\end{aligned}$$

4.1.2. Prediction of user characteristics

Given the degree of match $M(S_k)$ between a stereotype and the user's data, we can compute the strength of the predictions generated by the stereotype. For each predictive feature F'_i the value x'_{ij} associated with each linguistic value v'_{ij} corresponds to the probability that $F'_i = v'_{ij}$, given that the user belongs to the stereotype ($p(F'_i = v'_{ij} \mid S_k)$).

The prediction for a specific user is obtained by combining this value and the degree of match $M(S_k)$ between the user and the stereotype, computed using the classification slots:

$$p(F'_i = v'_{ij}) = p(F'_i = v'_{ij} \mid S_k) * M(S_k)$$

Considering again the example above, we can then generate the following predictions as regards the interests of this user, using the stereotype "Financial Professional":

$$\begin{aligned}
p(\text{interest in economy} = \text{high}) &= \\
&= p(\text{interest in economy} = \text{high} \mid \\
&\quad \text{financial professional}) * \\
&\quad M(\text{financial professional}) = \\
&= 1 * 0.75 = 0.75 \\
p(\text{interest in economy} = \text{medium}) &= \\
&= p(\text{interest in economy} = \text{medium} \\
&\quad \mid \text{financial professional}) * \\
&\quad M(\text{financial professional}) = \\
&= 0 * 0.75 = 0 \\
\dots
\end{aligned}$$

that is:

$$\begin{aligned}
p(\text{interest in economy} = \text{high}) &= 0.75 \\
p(\text{interest in economy} = \text{medium}) &= 0 \\
p(\text{interest in economy} = \text{low}) &= 0 \\
p(\text{interest in economy} = \text{null}) &= 0 \\
p(\text{interest in politics} = \text{high}) &= 0.525 \\
p(\text{interest in politics} = \text{medium}) &= 0.225 \\
p(\text{interest in politics} = \text{low}) &= 0 \\
p(\text{interest in politics} = \text{null}) &= 0 \\
\dots
\end{aligned}$$

Merging the stereotypical predictions. Since the user may partially match more than one stereotype in a family, the corresponding predictions have to be merged. We assume that the contributions by different stereotypes are independent and we use an additive formula to combine them. If, for a specific user, we have that:

$$\begin{aligned}
p(F'_i = v'_{ij}) &= X \text{ (using a stereotype } A) \text{ and} \\
p(F'_i = v'_{ij}) &= Y \text{ (using a stereotype } B),
\end{aligned}$$

then the combined prediction is:³

$$p(F'_i = v'_{ij}) = X + (1 - X) * Y$$

The final predictions are computed by normalizing the values obtained for the different linguistic values of each feature F'_i .

For example, from the predictions above we have that:

$$p(\text{interest in economy} = \text{high}) = 0.75$$

from the stereotype *financial professional*. If we assume that using the stereotype *young reader* we have

$$p(\text{interest in economy} = \text{high}) = 0.1$$

Then the merging of the two predictions produces the following result:

$$\begin{aligned}
p(\text{interest in economy} = \text{high}) &= \\
&0.75 + (1 - 0.75 * 0.1) = 0.775
\end{aligned}$$

In this way we can compute an initial model for a new user. This model provides, for each predictive feature F'_i , a probability distribution of the linguistic values v'_{ij} , computed for that specific user.

It is worth noting, in conclusion, that the kind of information provided by the registration form and then used for classifying the user is very general. Thus, for example, we can only make predictions on the user's interest/expertise in high-level sections (such as sport, politics, etc.) and we cannot be more specific about the subsections. The assumption we make is that the

³This formula is associative; therefore, it generalizes to the case of multiple data to be combined.

interest and expertise in all subsections of a section S are initialized with the predictions computed for S , i.e., they are exactly the same. We shall return to this issue in the next section.

4.2. Dynamic revision of the user model

The approach based on stereotypes discussed in the previous section supports the creation of an initial model for a user. However, this is not sufficient to achieve our goals. First of all, the predictions made by the stereotypes cannot be very precise due to the limited set of data provided by the user in the registration form. Second, we are interested in providing a personalization for each individual user and thus we want to achieve predictions that are more accurate (individualized) than those provided by the stereotypes. Finally, and most important, the interests/priorities/goals of a user may change during time and we are interested in tracking these changes and modifying the user model accordingly.

This can be done only by adopting some techniques for tracking the user's behavior and for dynamically revising the user model in accordance to such a behavior. In order to do that:

- We isolated a set of events (corresponding to actions performed by the user during her browsing in the news server) that are monitored by the system. We keep track of these events in a database; this allows us to generate statistics about the user's behavior.
- We introduced a set of *dynamic user modeling rules*, to regularly revise the user model. The rules are based on the monitored events and on the resulting statistics.

In the following, these two aspects will be discussed in more detail.

Events monitored by the system

As we noticed in Section 2, a user connecting to SeAN has always the possibility of modifying the choices made by the system. In particular, the user can always explore the whole space of alternatives, suppressing sections or news selected by the system, or asking to explore sections or news which were not selected by the system (see Figure 1). Moreover, the user can change the detail level in the presentation of each specific news item (see Figure 2).

SeAN monitors all the actions performed by the user during a session, tracking her behavior. In particular, the following events are monitored (and stored into the user's database):

- The sections and news that the user explores and the amount of time devoted by the user to each section;
- The sections (and news) that the system selected and the user suppressed;
- The sections or news that were not selected by the system and that the user asked to display;
- The details about a news item presented by the system and suppressed by the user;
- The user's requests for more detail (on a specific news item) than that selected by the system;
- The selection of specific advertisements.

These events provide important feedback from the user to the system, which can compute statistics about the user's behavior. Indeed, what is important to revise the user model is not a single event, but the whole history of events during time. For example, the fact that the user changes the detail level of the presentation for a single news item is not very relevant. In contrast, the fact that during a session and when looking at a specific section, she modifies the detail level of most of the news is an important piece of information to be taken into account. The following are examples of the kinds of statistics that are computed by our system:

- The percentage of subsections that the user opens while looking at a specific section.
- The percentage of news that the user looks at in each specific section.
- For each section, the percentage of subsections or news not displayed by the system that the user asks to display and, similarly, the percentage of subsections or news displayed by the system and that the user suppresses.
- For each section, the percentage of times that, while looking at a news item in that section, the user modifies the detail level of the presentation, by expanding or suppressing the attributes of the news. More specifically, given the definition of the presentation levels in terms of aggregations of attributes, the system monitors the percentage of times the user selects each possible presentation level.
- The number of banners (advertisements) on which the user clicks and the percentage of times the user clicks on banners of each category.

These statistics are used by a set of *user modeling rules* that are activated at specific times to revise the user models.

User modeling rules

As in the case of stereotypes, also for these rules we adopt a compositional approach. The rules are partitioned into groups and each group refers to one of the dimensions we isolated in user models. Indeed, looking at the events and statistics above, it should be clear that, for each dimension, only a subset of the events is relevant. For example:

- the events concerning the advertisements are relevant only for the *life style* of the user;
- the events concerning the number of subsections or news in a section S that the user explores or adds (removes) are relevant for the user's *interest* and *expertise* in that section;
- the events concerning the changes of detail level when looking at the news in a section S are relevant for the user's *expertise* in that section;
- the average number of news items explored by the user over all the sections and the average number of changes to the detail level are relevant for the user's *receptivity* (cognitive characteristics).

Each rule set is activated independently of the others and is used to revise a specific portion of the user model.

The rules are organized as compositions of conditions and consequents and have the following format: the antecedents are formed by logical conditions on the statistics about the user's behavior and the consequents specify changes to the probability distributions over some features of the user model.

For example, let us consider the rules in the *interests* family. One group of rules for this family takes into account the changes in the number of subsections/news made by the user. The following is an example of a (composite) rule:

Rule R021

Let X be the percentage of news added by the user (w.r.t. the maximum number of news)

in a section S ;

if $X \leq 5\%$ **then** no change to the user model

if $5\% < X \leq 35\%$ **then**

interest in S : *high* = +0.5, *medium* = +0.5

if $35\% < X \leq 70\%$ **then**

interest in S : *high* = +0.6, *medium* = +0.4

if $70\% < X \leq 90\%$ **then**

interest in S : *high* = +0.7, *medium* = +0.3

if $X > 90\%$ **then** *interest* in S : *high* = +1

In other words, the consequent of the rule specifies that, if during a session, the user adds subsections/news in S , the prediction on the user's interest in S has to be changed; we shall return to that in the next paragraph.

It is worth noting that the statistics about the user's behavior concern both sections and subsections. In this way rules like the one above can generate specific predictions about the user's interest in each subsection, while the predictions generated by the stereotypes are the same for a section and all of its subsections (see the comments at the end of section 4.1.2).

As another example the following is a (composite) rule affecting the user's receptivity:

Rule R037

Let X be the percentage of news items removed by the user (w.r.t. the number of news items presented by the system);

if $X \leq 10\%$ **then** no change to the user model

if $10\% < X \leq 20\%$ **then**

receptivity: *null* = +0.5, *low* = +0.5

if $20\% < X \leq 40\%$ **then**

receptivity: *null* = +0.6, *low* = +0.4

if $40\% < X \leq 60\%$ **then**

receptivity: *null* = +0.7, *low* = +0.3

if $60\% < X \leq 80\%$ **then**

receptivity: *null* = +0.8, *low* = +0.2

if $X > 80\%$ **then** *receptivity*: *null* = +1

Rule activation and revision of the user model.

Two aspects still have to be defined:

1. When the rules are activated;
2. How the user model is revised as a result of the activation of a set of rules.

Different choices can be made as regards the first problem. On the one hand, the changes cannot be too frequent during a session, otherwise the changes in the user model and the consequent changes in the presentation of sections and news may confuse the user. In particular, changes should not influence in any way the sections and news that the user has already browsed during a session, otherwise the user may get completely confused about the news server. On the other hand, the update cannot be too slow, otherwise the user may get the impression that the system is not adapting at all to her behavior. We considered two options: (i) activating the rules each time the user moves from one section to another section and (ii) activating the rules at the end of a session⁴.

⁴Notice that, in other systems, such as [26], the personalization decisions are further clarified by negotiating with the user the system's proposals to the presentation changes and letting her accept or refuse such proposals. We recognize that this type of negotiation

In the current prototype we decided to implement the second option: therefore, the user's behavior is monitored during a session and the user modeling rules for the revision of the user model are applied when the user closes the session.

Our approach has several advantages concerning both the efficiency of the system and the coherency of the system's behavior.

- The statistics that can be made by looking at the behavior within a whole session are more reliable and more significant than those regarding shorter browsing periods, which might also lead to useless oscillations in the user model.
- Carrying on the analysis of the user's actions and the revision of the user model at the end of the user session enables the system to work "off-line"; therefore, the user does not perceive any delays due to the performance of the revision task.
- As discussed in [37], the predictability of a user interface is an extremely important property and has to be guaranteed to avoid that users get confused while interacting with the system.

Let us turn now to discussing how the model is revised using the rules. Once a rule is activated, the probability distributions in the user model are first combined with the predictions expressed in the consequent of the rules, and then normalized obtaining the revised distributions.

For example, let us suppose that the user model contains the following probability distribution for the interest in a section s_i :

$$\begin{aligned} p(\text{interest in } s_i = \text{null}) &= 0.1 \\ p(\text{interest in } s_i = \text{low}) &= 0.2 \\ p(\text{interest in } s_i = \text{medium}) &= 0.6 \\ p(\text{interest in } s_i = \text{high}) &= 0.1 \end{aligned}$$

and that, during a session, the user modifies the presentation of s_i adding 75% of the news that were not displayed by the system. At the end of the session the rule $R021$ above can be fired, leading to the following modification of the probability distribution of the user's interest in s_i :

can be very helpful, on the one hand, to make the system's behavior understandable to the user and, on the other hand, to collect relevant feedback from the user. However, we have not yet addressed this type of functionality in our system: in fact, to manage this type of interaction in a fully acceptable way, a trade-off must be resolved between explaining the system's behavior and boring the user with frequent questions, or with questions which she can hardly understand.

$$\begin{aligned} p(\text{interest in } s_i = \text{null}) &= 0.1 \\ p(\text{interest in } s_i = \text{low}) &= 0.2 \\ p(\text{interest in } s_i = \text{medium}) &= 0.6 + 0.3 = 0.9 \\ p(\text{interest in } s_i = \text{high}) &= 0.1 + 0.7 = 0.8 \end{aligned}$$

which, after the normalization becomes:

$$\begin{aligned} p(\text{interest in } s_i = \text{null}) &= 0.05 \\ p(\text{interest in } s_i = \text{low}) &= 0.1 \\ p(\text{interest in } s_i = \text{medium}) &= 0.45 \\ p(\text{interest in } s_i = \text{high}) &= 0.4 \end{aligned}$$

This new distribution will be used when the user connects again to the news server. Similarly for all the other rules that are activated.

In this way we take into account both the history (represented by the probability distributions in the model) and the new information coming from the monitored events (represented by the changes proposed by the rules), trying to avoid abrupt changes in the user model.

5. Agents for selecting news and for generating the presentation

In this section we discuss how the personalized news hypertext is generated, starting from the information in the user model ((4) in figure 3). This task is performed by two different agents that will be presented in the next sections:

- An agent that personalizes the *content* of the presentation: given the pieces of information in the user model (i.e., information about the user interest, expertise, receptivity and life style), it decides which sections and news have to be presented, the appropriate detail level for each section and news item and the advertisements that have to be included in each page.
- An agent that generates the hypertextual pages. This agent could also be responsible for personalizing the form of the presentation; however, in the current prototype we decided to adopt a simple and standard hypertextual format, focusing on the personalization of the content. This means that the layout of the hypertext is the same for all users but it would be easy to adopt forms of personalization also on this aspect.

5.1. Personalizing the content

The agent in charge of personalizing the content of the presentation makes decisions about the sections (and subsections) to be presented, the news items to be displayed in each section with the appropriate detail level and the advertisements to be added to each page.

This decision process relies on a knowledge base formed by various sets of rules and on a heuristic scoring approach. The rules take into account the different pieces of information contained in the user model.

Taking into account interest and expertise

A first set of rules is used to evaluate, for each section and for each possible presentation format (detail level, as defined in Figure 4), the probability that the user wants to read the news of the section at that level. In order to make this computation we use a matrix that specifies the probability of each detail level for a generic section S , given all possible combinations $\langle X, Y \rangle$ of linguistic values of the interest and expertise in the subject of S :

$$p(\text{level} = i \text{ for section } S \mid \text{interest in } S = X, \text{ expertise in } S = Y)$$

This means that there are 16 entries for each detail level i , one for each combination of linguistic values for interest and expertise. For example:

$$\begin{aligned} p(\text{level} = 4 \text{ for } S \mid \text{interest in } S = \text{high}, \text{ expertise in } S = \text{high}) &= 0.2 \\ p(\text{level} = 4 \text{ for } S \mid \text{interest in } S = \text{high}, \text{ expertise in } S = \text{medium}) &= 0.4 \\ p(\text{level} = 4 \text{ for } S \mid \text{interest in } S = \text{medium}, \text{ expertise in } S = \text{high}) &= 0.4 \\ p(\text{level} = 4 \text{ for } S \mid \text{interest in } S = \text{medium}, \text{ expertise in } S = \text{medium}) &= 0.7 \\ \dots \end{aligned}$$

The last, for example, specifies that the probability of level 4, given that the user has a medium interest and expertise, is 0.7. Notice that the rules include a level 0, that corresponds to disregarding the section.

The rules are applied, for each section s_i , to the probability distribution for the interest and expertise in s_i , stored in the user model. In particular, the application of the rules to the sections of the news server enables the computation of one probability value for each detail level for each section.

The computation for a section s_i is performed as follows: for each level k , we consider the 16 entries in the matrix and we compute 16 values for $p(\text{level} = k \text{ for section } s_i)$, given the probability distributions for the user's interest and expertise. The 16 values are added and then a normalization is performed considering all levels.

For example, if the user model predicts for section s_i that:

$$\begin{aligned} p(\text{interest in } s_i = \text{high}) &= 0.2 \\ p(\text{interest in } s_i = \text{medium}) &= 0.7 \\ p(\text{expertise in } s_i = \text{high}) &= 0.3 \\ p(\text{expertise in } s_i = \text{medium}) &= 0.6 \\ \dots \end{aligned}$$

then, we have:

$$\begin{aligned} p(\text{level} = 4 \text{ for section } s_i) &= \\ & (p(\text{level} = 4 \text{ for section } S \mid \text{interest in } S = \text{high}, \text{ expertise in } S = \text{high}) \\ & \quad * p(\text{interest in } s_i = \text{high}) \\ & \quad * p(\text{expertise in } s_i = \text{high})) + \\ & (p(\text{level} = 4 \text{ for section } S \mid \text{interest in } S = \text{medium}, \text{ expertise in } S = \text{high}) \\ & \quad * p(\text{interest in } s_i = \text{medium}) \\ & \quad * p(\text{expertise in } s_i = \text{high})) + \\ & (p(\text{level} = 4 \text{ for section } S \mid \text{interest in } S = \text{high}, \text{ expertise in } S = \text{medium}) \\ & \quad * p(\text{interest in } s_i = \text{high}) \\ & \quad * p(\text{expertise in } s_i = \text{medium})) + \\ & (p(\text{level} = 4 \text{ for section } S \mid \text{interest in } S = \text{medium}, \text{ expertise in } S = \text{medium}) \\ & \quad * p(\text{interest in } s_i = \text{medium}) \\ & \quad * p(\text{expertise in } s_i = \text{medium})) + \\ & \dots = \\ & = 0.2 * 0.2 * 0.3 + 0.4 * 0.7 * 0.3 + \\ & \quad + 0.4 * 0.2 * 0.6 + 0.7 * 0.7 * 0.6 + \dots \end{aligned}$$

which has to be normalized after performing the same computation for each level k .

Taking into account the receptivity

In a second step we use information about the user's receptivity to decide which sections should be presented and the appropriate detail level for each one of them. This is done in two sub-steps:

a) *Selecting the sections to be presented.* This decision is made by considering the scores computed by the

rules taking into account interest and expertise (see the step above). All the sections for which the level 0 has the highest score or for which the cumulative scores of levels 0 and 1 is over a threshold (0.7) are excluded from the set P of sections to be presented. Those for which the level 5 has the highest score or such that the cumulative scores of levels 4 and 5 is over a threshold (0.7) are included in P . The remaining sections R are ranked according to the distribution of the scores.

b) Defining the number of sections and selecting the detail level of news in each section. As said, the basis for this computation is the user's receptivity: first of all, given the user's receptivity, the system decides how many sections (and subsections) have to be presented. This corresponds to deciding the percentage of sections in R that have to be added to P . Second, for each section S in P , the system considers the detail level L with the highest score and evaluates whether it is compatible with the user's receptivity. If it is, then this is the detail level for S ; otherwise, we have to consider two cases:

- L is too detailed with respect to the user's receptivity. This means that we do not expect that the user could receive all the pieces of information corresponding to the level of detail L , although the user is interested in such a topic and has some expertise on it. Thus, the system searches for a lower level representing a good compromise between the scores computed by the first group of rules and the user's receptivity.
- L is low with respect to the user's receptivity. This means that the user could receive more information. However, L has been decided given the user's interest and expertise and thus we keep it as the level of the presentation. This means that, although the user could receive more information, presumably she is not very interested (or expert) in the topic and thus it is not reasonable to provide more information.

If the selected level is 2 or 3, a decision between 2a and 2b (3a and 3b) has to be made. The choice between the abstract and the full article is based on the user's interest and expertise on the topic of the section under examination, and on her receptivity: the full article is presented only when, for all these features, the probabilities of the values "low" and "null" are close to zero. Moreover, in case there are alternatives (e.g., between a picture or a video clip), the choice is based on the user's past preferences.

Selecting the advertisements

The selection of the advertisements to be included in each page depends on: (i) the topic of the page, that is the section to which it belongs, and (ii) the classification of the user according to the *Life Style*, which specifies a degree of compatibility between the user and each stereotype in that family. The selection is performed as follows: first we consider the life style classes whose compatibility is over a threshold; then we select advertisements that are compatible with the topic of the page and whose target includes these classes, with frequency proportional to the compatibility values of the classes.

5.2. Hypertextual format for the presentation

In this section we discuss the strategies that we adopt to organize the pieces of information, selected using the techniques discussed in the previous section, into a set of hypertextual pages ((5) in figure 3). The pages are generated dynamically according to a standard format. In particular, we chose a simple hypertextual structure which is very easy to navigate and allows the user to modify the system's choices, both as regards the sections and news selected by the system and as regards their detail level.

The hypertext is structured according to the hierarchy of sections and subsections in the news database. This means that we generate one page for each section and such page contains links for moving in the hierarchy and for displaying the news items in the corresponding section.

All the pages are generated dynamically, including the home page (Figure 1), which contains a list of the high-level sections that are considered relevant for the specific user (for each section the selected sub-sections are listed as well). In the bottom part of the page there are two menus: one allows the user to delete sections from the home page (the sections she is not interested in); the other allows the user to add sections that were not selected by the system but are interesting for the user. A banner with advertisements is located at the top of the page.

Whenever the user opens (browses) a section S , the corresponding page is generated by the system. The page is divided into three parts (see Figure 7):

- The upper part contains banners with the selected advertisements.
- The middle part contains the navigation buttons. In particular, we have:

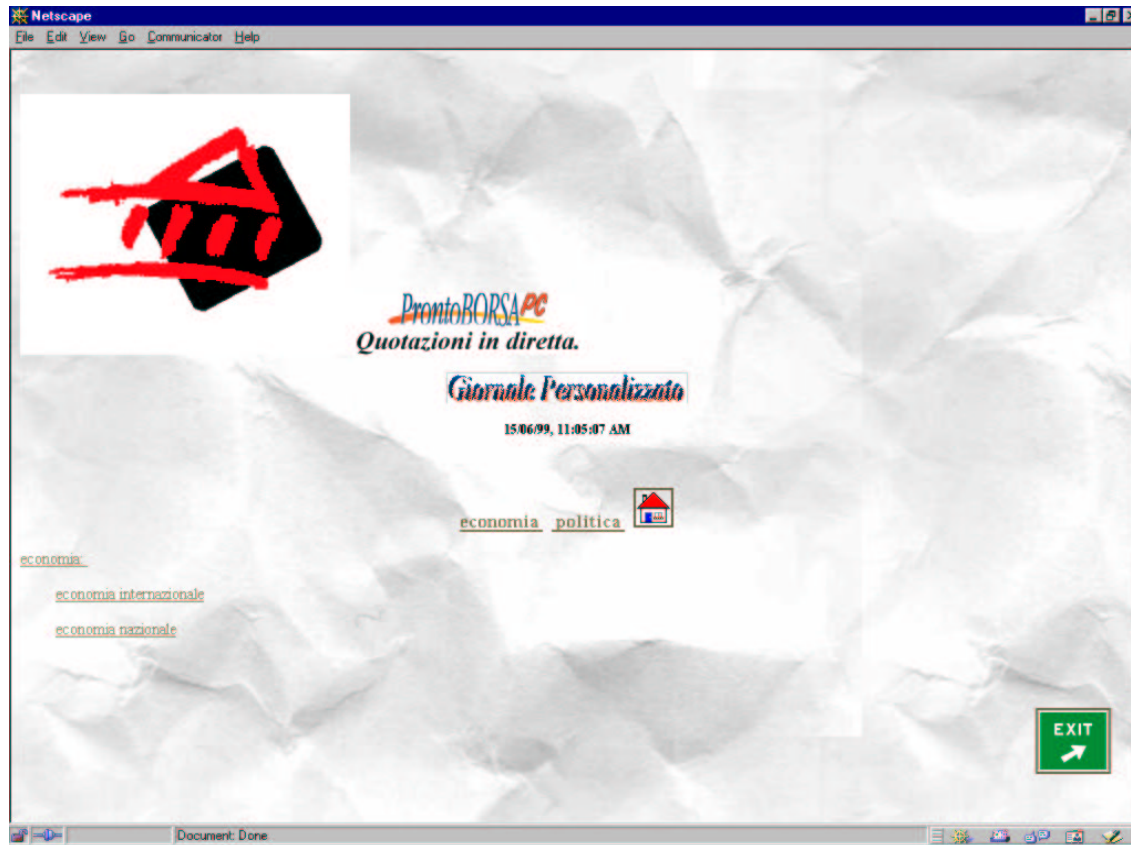


Fig. 7. Example of page corresponding to the section *economia*

- * A list of the highest-level sections in the news server (naturally only the ones that are considered of interest for the user, the sections “Economia” (“Economy”) and “Politica” (“Politics”) in Figure 7).
 - * The *home* icon that allows the user to start the navigation again from the home page of the server.
- The lower part of the page contains the name of the section S being displayed (“Economia” in Figure 7) and a list of its subsections (or news) that are considered relevant for the user (the subsections “Economia internazionale” (“International Economy”) and “Economia nazionale” (“National Economy”) in Figure 7. Each name of subsection is a link to the page corresponding to the subsection. A menu with the button “Elimina” (“Delete”) allows the user to delete subsections or news from the list; similarly another menu with the button “Aggiungi” (“Add”) allows the user to list other subsections (news) not selected by the

system (the menus and button are not visible in Figure 7 but are similar to those in the Home Page, see Figure 1).

Whenever the user clicks on the title of a news item, a new pop-up window is opened by the system for displaying the corresponding pieces of information (see the example in Figure 2). The news item is displayed at the selected detail level, i.e., some of its attributes are displayed while others are not. For each displayed attribute there is a delete button that allows the user to hide the corresponding pieces of information (the window is refreshed after this action). As regards the attributes that are not displayed (i.e., that were considered as too detailed for the user), only the name of the attribute is included in the page. The name of an attribute is actually a button that allows the user to display in the page the pieces of information corresponding to it (the window is refreshed whenever the user clicks on one of these buttons).

6. Experiments

In this section we briefly report on a first set of experiments that we made on our system. It is worth noting that the only aim of these experiments was to verify that the approach is feasible and can provide interesting forms of personalization. More intensive testing would be needed in order to refine the knowledge bases (stereotypes and rules) and to actually validate the system.

The test has been performed upon a small set of users (25), selected as representatives of different categories of readers. We presented to each user the complete list of sections and subsections in the server and we asked her to select those she was interested in and to express a measure of this interest (high, medium, low, null). We then asked each user to fill in the registration form and to use SeAN. As a result, SeAN classified the user based on the stereotypes, generated predictions on her interest and produced personalized pages with selected sections, news and detail levels. The result of this personalization has been compared with the preferences expressed a-priori by the user. We evaluated the comparison as follows:

- The selection of the system was judged compatible with the preferences of the user if (i) the system included all the sections that the user indicated as of high interest; (ii) the system did not include any section that the user indicated as of no interest and (iii) the detail level selected by the system was (in most of the cases) in accordance with the degree of interest indicated by the user (i.e., the detail level of a section indicated as of high interest was not lower than that of a section indicated as of medium interest and so on for sections of medium and low interest).
- The selection was judged completely incompatible if the system failed to include all sections indicated as of high interest and included at least a section of no interest.

In 16 of the 25 cases the system provided results compatible with the user's desiderata, while in 5 of the cases the system completely failed to satisfy the desiderata.

A couple of examples of test cases are reported in the following. Let us consider a male user, whose age is between 36 and 45 with a degree in economy, working as a manager and connecting to the server for work (the screenshots displayed in the paper correspond to this profile). In this case the system selected the sec-

tions "Economy" and "Politics" (and their subsections) and selected the highest detail level for presenting the news in these sections. On the other hand, in the case of a male user, whose age is between 14 and 19, studying in a technical school and connecting for personal reasons, the system displayed the sections "Sport" and "Technology" (with detail level 3, i.e., including the text and video/audio clips but leaving out interviews and technical data).

Although the test set was very small and in many senses naive, the results are quite encouraging on the feasibility and on the practical applicability of the approach. Further work has to be done in order to refine the knowledge bases in the prototype system.

7. Conclusions

In the paper we presented SeAN, an adaptive system for accessing on-line news servers. SeAN is designed as a set of cooperating agents, following a three-tier architecture, which has been inherited from SETA, a system for adaptive electronic commerce [3]). The system is implemented using Java; the communication between the agents is obtained using the RMI facility offered by Java. The three databases are implemented using MS Access and are accessed using JDBC (thus, in principle, they could be implemented using any relational DBMS).

SeAN tailors the presentation of news to the peculiarities of the user by exploiting a structured database of news where items are classified in a hierarchy of topics and are characterized by a set of attributes. These attributes can be shown or hidden, depending on the content of the user model. This is a significant difference with respect to other approaches to personalized information filtering, which do not assume information is structured and then have difficulties in the selection of contents, especially as regards the detail level. Filling in the database of news is additional work. However, this task is not very different from what has to be done in the software systems that are used in the editorial offices of some newspapers. These systems require that the author of a paper submits her work to a specific section and, in case there are photos or extra items, she must specify the paper these items are related to (this is used to define the layout of the pages).

It is worth noting, moreover, that there is a difference between the goals of personalization in our approach and those pursued in systems for the personalization of electronic catalogues [38] or archives, such

as virtual museums [22, 40] or information kiosks. In fact, we aim at defining the “right” detail level for presenting news on a given topic by aggregating different attributes of the news, while they generate personalized linguistic descriptions for presenting in different ways the items in the database.

The system we presented relies on a modular and compositional approach to user modeling. Indeed this work is part of a bigger project in which we aim at creating an ontology for user modeling and at defining guidelines for selecting the dimensions that are relevant, given an application task and domain and given specific goals for personalization [47]. This modular approach has, in our view, several advantages. First of all, creating stereotypes and user modeling rules is made simpler since these knowledge bases must focus only on one specific user modeling dimension at a time. Second, modularity allows the re-use of user modeling knowledge bases. In fact, there are dimensions that are common to multiple application tasks and domains and thus the knowledge bases referring to these dimensions can be easily re-used. For example, a dimension such as the “life style” is relevant in several tasks and domains and indeed the knowledge bases we used in the system described in the paper have been re-used in other systems for electronic commerce.

References

- [1] E. André and T. Rist. Presenting through performing: on the use of multiple lifelike characters in knowledge-based presentation systems. In *Proc. 2000 Int. Conf. on Intelligent User Interfaces (IUI'00)*, pages 1–8, New Orleans, Louisiana, 2000.
- [2] L. Ardissono and A. Goy. Tailoring the interaction with users in web stores. *User Modeling and User-Adapted Interaction*, 10(4):251–303, 2000.
- [3] L. Ardissono, A. Goy, R. Meo, G. Petrone, L. Console, L. Lesmo, C. Simone, and P. Torasso. A configurable system for the construction of adaptive virtual stores. *World Wide Web*, 2(3):143–159, 1999.
- [4] F. Asnicar and C. Tasso. ifWeb: a prototype of user model-based intelligent agent for document filtering and navigation in the World Wide Web. In *Proc. workshop "Adaptive Systems and User Modeling on the World Wide Web"*, pages 3–12, 1997.
- [5] Autonomy. Technology white paper. <http://www.autonomy.com>, 2000.
- [6] M. Balabanović. Exploring versus exploiting when learning user models for text recommendation. *User Modeling and User-Adapted Interaction*, 8:71–102, 1998.
- [7] D. Bental, A. Cawsey, J. Pearson, and R. Jones. Adapting web-based information to the needs of patients with cancer. In *Proc. International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH 2000)*, pages 73–85, Trento, Italy, 2000.
- [8] D. Benyon. Adaptive systems: a solution to usability problems. *User Modeling and User-Adapted Interaction*, 3:65–87, 1993.
- [9] D. Billsus and M. Pazzani. A personal news agent that talks, learns and explains. In *Proc. 3rd Int. Conf. on Autonomous Agents (Agents '99)*, pages 268–275, Seattle, WA, 1999.
- [10] P. Brusilovsky. Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction*, 6(2-3):87–129, 1996.
- [11] P. Brusilovsky. Adaptive hypermedia. *User Modeling and User-Adapted Interaction*, 11(1-2):87–110, 2001.
- [12] P. Brusilovsky, E. Schwartz, and G. Weber. ELM-ART: An intelligent tutoring system on World Wide Web. In *Proc. 3rd Int. Conf. on Intelligent Tutoring Systems (ITS-96)*, Montreal, 1996.
- [13] R.D. Burke, K.J. Hammond, and B.C. Young. The FindMe approach to assisted browsing. *IEEE Expert*, pages 32–39, 1997.
- [14] G. Calvi. *Indagine Sociale Italiana, rapporto Eurisko*. Franco Angeli, 1986.
- [15] L. Calvi and P. De Bra. Proficiency-adapted information browsing and filtering in hypermedia educational systems. *User Modeling and User-Adapted Interaction*, 7:257–277, 1997.
- [16] S. Carberry, J. Chu-Carroll, and S. Elzer. Constructing and utilizing a model of user preferences in collaborative consultation dialogues. *Computational Intelligence*, 15(3):185–217, 1999.
- [17] B.N. De Carolis, F. de Rosis, C. Andreoli, V. Cavallo, and M.L. De Cicco. The dynamic generation of hypertext presentations of medical guidelines. *New review of Hypermedia and Multimedia*, 4, 1999.
- [18] K. Cheverest, N. Davies, K. Mitchell, and P. Smith. Providing tailored (context-aware) information to city visitors. In *Proc. International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH 2000)*, pages 73–85, Trento, Italy, 2000.
- [19] D.N. Chin. Acquiring user models. *Artificial Intelligence Review*, 7:185–197, 1993.
- [20] L. Chittaro and R. Ranon. Adding adaptive features to virtual reality interfaces for e-commerce. In *Proc. International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH 2000)*, pages 86–97, Trento, Italy, 2000.
- [21] P. Cotter and B. Smyth. Personalisation technologies for the digital TV world. In *Proc. 14th Conf. ECAI*, pages 701–705, Berlin, 2000.
- [22] R. Dale, S.J. Green, M. Milosavljevic, and C. Paris. Dynamic document delivery: Generating natural language texts on demand. In *Proc. 9th Int. Conf. and Workshop on Database and Expert Systems Applications (DEXA'98)*, Vienna, 1998.
- [23] B.N. De Carolis. Introducing reactivity in adaptive hypertext generation. In *Proc. 13th Conf. ECAI*, Brighton, UK, 1998.
- [24] J. Domingue and P. Scott. KMi Planet: a Web based news server. In *Asia Pacific Computer Human Interaction Conference (APCHI'98)*, Kanagawa, Japan, 1998.
- [25] J. Fink and A. Kobsa. A review and analysis of commercial user modeling servers for personalization on the World Wide Web. *User Modeling and User-Adapted Interaction, Special Issue on Deployed User Modeling*, 10(2-3):209–249, 2000.
- [26] J. Fink, A. Kobsa, and A. Nill. Adaptable and adaptive information for all users, including disabled and elderly people. *New review of Hypermedia and Multimedia*, 4:163–188, 1998.

- [27] D.R. Greening. LikeMinds: Building consumer trust with accurate product recommendations. http://likeminds.com/technology/white_papers/websell/, 2000.
- [28] N. Henze and W. Nejdl. Adaptivity in the KBS hyperbook system. In *Proc. 2nd Workshop on adaptive systems and user modeling on the World Wide Web*, pages 67–74, Banff, Canada, 1999.
- [29] G. Hirst, C. DiMarco, E. Hovy, and K. Parsons. Authoring and generating health-education documents that are tailored to the needs of the individual patient. In *Proc. 6th Conf. on User Modeling*, pages 107–118, Chia Laguna, Italy, 1997.
- [30] T. Joachims, D. Freitag, and T. Mitchell. WebWatcher: a tour guide for the World Wide Web. In *Proc. 15th IJCAI*, pages 770–775, Nagoya, Japan, 1997.
- [31] T. Joerding. Intelligent multimedia presentations in the web: Fun without annoyance. In *Proc. of the 7th World Wide Web Conference (WWW7)*, Brisbane, Australia, 1998.
- [32] N. Karunanithi and J. Alspector. Feature-based and clique-based user models for movie selection. In *Proc. 5th Int. Conf. on User Modeling*, pages 29–34, Kailua-Kona, Hawaii, 1996.
- [33] J. Kay. Vive la difference! individualised interaction with users. In *Proc. 14th IJCAI*, pages 978–984, Montreal, 1995.
- [34] A. Kobsa. Generic user modeling systems. *User Modeling and User-Adapted Interaction, Ten Year Anniversary Issue*, 2000.
- [35] A. Kobsa, J. Koenemann, and W. Pohl. Personalized hypermedia presentation techniques for improving online customer relationships. *The Knowledge Engineering Review*, page to appear, 2001.
- [36] M.F. McTear. User modelling for adaptive computer systems: a survey of recent developments. *Artificial Intelligence Review*, 7:157–184, 1993.
- [37] C.A. Miller. Rules of etiquette, or how a mannerly AUI should comport itself to gain social acceptance and be perceived as gracious and well-behaved in polite society. In *Working Notes of the "Adaptive User Interfaces" Spring Symposium of AAAI (Technical Report SS-00-01)*, pages 80–81, Stanford, CA, 2000.
- [38] M. Milosavljevic and J. Oberlander. Dynamic hypertext catalogues: Helping users to help themselves. In *Proc. 9th ACM Conf. on Hypertext and Hypermedia*, 1998.
- [39] E. Not and M. Zancanaro. The macroNode approach: mediating between adaptive and dynamic hypermedia. In *Proc. International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH 2000)*, pages 166–178, Trento, Italy, 2000.
- [40] D. Petrelli, A. De Angeli, and G. Convertino. A user centered approach to user modelling. In *Proc. 7th Int. Conf. on User Modeling*, pages 255–264, Banff, Canada, 1999.
- [41] W. Pohl and A. Nick. Machine learning and knowledge representation in the LabouUr approach to user modeling. In *Proc. 7th Int. Conf. on User Modeling*, pages 179–188, Banff, Canada, 1999.
- [42] H. Popp and D. Lödel. Fuzzy techniques and user modeling in sales assistants. *User Modeling and User-Adapted Interaction*, 6:349–370, 1996.
- [43] P. Resnick and H.R. Varian, editors. *Special Issue on Recommender Systems*, volume 40. Communications of the ACM, 1997.
- [44] E. Rich. Stereotypes and user modeling. In A. Kobsa and W. Wahlster, editors, *User Models in Dialog Systems*, pages 35–51. Springer Verlag, Berlin, 1989.
- [45] D. Riecken, editor. *Special Issue on Personalization*, volume 43. Communications of the ACM, 2000.
- [46] I. Schwab, W. Pohl, and I. Koychev. Learning to recommend from positive evidence. In *Proc. 2000 Int. Conf. on Intelligent User Interfaces*, pages 241–247, New Orleans, Louisiana, 2000.
- [47] I. Torre. A modular approach for user modelling. In *Proc. International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH 2000)*, pages 414–420, Trento, Italy, 2000.
- [48] W. Wahlster and A. Kobsa. *User Models in Dialog Systems*. Springer Verlag, 1989.
- [49] R. Wilkinson, S. Lu, F. Paradis, C. Paris, S. Wan, and Mi Wu. Generating personal travel guides from discourse plans. In *Lecture Notes in Computer Science n. 1892: Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 392–85. Springer Verlag, Berlin, 2000.