Personalized News Access

D. G. Kaklamanos, K. G. Margaritis

Parallel and Distributed Processing Laboratory, University of Macedonia, Greece
Email: {kaklaman, kmarg}@uom.gr

Abstract

PENA (Personalized News Access) is an adaptive system for the personalized access to news. The aims of the system are to collect news from predefined news sites, to select the sections and news in the server that are most relevant for each user and to present the selected news. In this paper are described the news collection process, the techniques adopted for structuring the news archive, the creation, maintenance and update of the user model and the generation of the personalized web pages. This is a preliminary work that is based on the system that is described in [1].

1 Introduction

Simple forms of personalization are implemented in many sites, based on preferences directly expressed by the users, while there are more complex approaches proposed in the literature, for example [2, 3]. In this paper, PENA is introduced, an adaptive web site for accessing news stories. PENA is a multi-agent system, which aims at personalizing both the selection of topics that are of interest to the user and the presentation of each news item.

This paper is organized as follows. In Section 2 we discuss the functionality requirements of the system. We then focus on three main aspects: the organization of content, described in Section 3, the adopted user modelling techniques, described in Section 4, and the news retrieval process, described in Section 5. Section 6 outlines the personalization strategy of PENA. Section 7 describes some testing experiments of the system and Section 8 gives an overview of the future goals.

2 Functionality, Requirements and Architecture of PENA.

In PENA we had two main functionality goals: Firstly, the system should be able to collect news from the web. Secondly, the system should present only the news that the user is interested in, and leave the others aside, as information which can be reached on demand.

As a first requirement, we imposed that these forms of personalization should be provided to both first time and frequent users. As a consequence, PENA must be able to generate an initial, possibly approximate, model of first time users.

Another important requirement is the ability of the system to revise the user model, in order to make it as close as possible to his actual characteristics and follow possible changes of interest.

Thirdly, the system should never impose its choices on the user. The user must be always able to modify system decisions. Moreover, being the user able to make changes supports the user modelling process, as the system can get information about the user's actual characteristics and interests. As a consequence, all the web pages generated by PENA contain buttons for modifying the system choices.

These functionality goals imposed some requirements on the design of PENA, which is organized as three tier architecture. 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-modelling and personalization activities, second tier, and which can access information maintained in a set of databases, third tier.

A user interaction agent collects user requests and returns web pages to the browser. User modelling is carried out by two agents: one is activated only when a user connects to the server for the first time and it creates the initial user model based on the data that are provided by the user in the registration form and a library of stereotypes. The second agent is activated at the start of each session and is responsible for revising the user model, based on the user's actions during his last session and on a set of user modelling rules. The user models are maintained in a database and thus can be retrieved each time the user connects to the site.

Two other agents are in charge of extracting, respectively, from the news database and from the user database, the specific pieces of information that are going to be presented to a user, given his model. All these pieces of information are then passed to the interaction agent, which generates the web pages and sends them to the user's browser.

3 Organization of Content.

The news database contains all the news stories that can be presented to the user, and is able to support the classification of news according to their topics. In PENA, we defined a hierarchy of sections, in which the news is classified. This hierarchy was decided based on the high level classification of news that is done to the web sites from where the news stories are downloaded. More specifically, we considered the high-level sections politics, economy, sport, technology, culture and world. As the news are retrieved from the web sites, the system, based on the web site news classification, stores them locally in the corresponding category.

There are many reasons that justify this choice. First of all, it worth's maintaining this high level classification since it is accurate. Second, the high level classification of news sections allows us to provide some form of personalization even when limited information about the user is available. This is very common for first time users. In this case, it is difficult to make a selection of topics more accurate than the one corresponding to high level sections in the hierarchy.

The news database consists of three Tables. The main table contains the article id, the source of the article and the high level category of the article. The other two tables contain the names of the categories and the name of the sources (news portal) of the articles. It must be mentioned that the retrieval part of PENA is responsible for populating this database.

4 User Modelling.

The user model plays a central role in the personalization process of PENA, because accurate personalization requires the modelling of different user features. For example, the selection of the sections to be displayed is related to the user's interests. In PENA, the user model is divided into a set of dimensions which represent the different level of user interest for a news section. In this case we considered four separate dimensions: high, medium, low, null.

We can turn now to discuss our user modelling approach more precisely. In our application, there are two separate phases. The first one regards the creation of an initial model for a user connecting to the server for the first time. The second one revises the user model, by tracking the user's actions, in order to identify his interests and, possibly, the changes about them during time. The creation of the initial model is based on the use of stereotypical knowledge. The second phase is based on user modelling rules that are activated based on the analysis of user's actions during the browsing the news.

4.1 Using Stereotypes for Initializing the user model.

The creation of an initial model for first-time users is based on the use of stereotypes [4], which represent the features of classes of readers. The data used in the classification are collected from the user in a registration form, which contains a small set of questions.

At this point, it must be mentioned that there are seven predefined stereotypes in the database. These stereotypes are twenty four fields that contain probabilities and are grouped in six logic groups. For each one of the six interests, which are presented in the registration form, the user is asked to choose one of the four possible interests, high, medium, low, null. Each one of the twenty four probabilities corresponds to each one of those possible user selections.

The stereotypes consist of two groups of slots: classification slots and prediction slots. The structure of the slots is the same: Each slot corresponds to a feature F_i of the user. For each feature F_i we have a set of linguistic values $(v_{i1}, , v_{ik})$. A numeric value x_{ij} E [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. 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.

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. 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, we assume that the features are independent. 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.

At this point, there are seven probabilities. The maximum probability is selected. In this way we decide about the stereotype which is closer to the user. All the probabilities of the selected stereotype are chosen and stored in the database, as the initial user profile.

It is worth noting that the 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 interests in high-level sections, such as sport, politics.

4.2 Dynamic revision of the user model.

Since the interests a user may change, we are interested in both tracking any changes and modifying the user model accordingly. This can be achieved only by adopting some techniques for tracking the user's behaviour and for dynamically revising the user model in accordance to user behaviour.

In order to do that: a set of events was isolated, corresponding to actions performed by the user during his browsing session. A set of dynamic user modelling rules for the regular revision of the user model was introduced. 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.

4.3 Events monitored by the system.

PENA monitors and stores into the database the following events: Firstly, the news stories that the user reads and the amount of time devoted for reading by the user to each news story. Secondly, the sections that the system selected and the user maintained. Finally, the sections that were not selected by the system and that the user asked to display.

These events provide important feedback about the user actions to the system, in order to compute statistics about the user's behaviour. The reliable revision of the user model, evaluates as more important the whole history of events during time than a single event.

The computed statistics are: The percentage of stories that the user reads in each specific section. The percentage of news, the user does not read. For each section, the percentage of news not displayed by the system that the user asks to display and the percentage of news displayed by the system that the user suppresses. Finally, the amount of time the user spends in each news story he reads.

These statistics are used by a set of user modelling rules in order to revise the user models.

4.4 User modeling rules.

At this point two things must be mentioned about the rules. Firstly, each rule is activated independently of the others and is used to revise a specific portion of the user model. Secondly, 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 behaviour and the consequents specify changes to the probability distributions over some features of the user model.

The system contains three rules. The main goal of the first rule modifies the probabilities of the user model in a way that reassures the selection of the news category. Based on the fact that a news category is selected as relevant for a particular user if the sum of the low and null probabilities from this category is less than forty nine percent, the probabilities of a news category that was added is formulated as follows: the high probability of the news category is set to sixty percent, the medium probability is set to forty percent and the low and null probabilities of the news categories are set to zero.

The third rule is the opposite of the first one, as it deals with the removal of an entire news category from those that are presented to the user. Based again on the fact that a news category is selected as relevant for a particular user if the sum of the low and null probabilities from this category is less than forty nine percent, the probabilities of a news category that was removed is formulated as follows: the null probability of the news category is set to sixty percent, the low probability is set to forty percent and the high and medium probabilities of the news category are set to zero.

Finally, the second rule deals with the view of an article from the user. The main idea of this rule is to either reinforce the existing user profile, by increasing the high probability ten percent, if the user spent more than ten seconds in the article, or decrease the existing profile by decreasing the null probability of the news category by ten percent.

4.5 Rule Activation and Revision of the User Model.

Two aspects still have to be defined: When the rules are activated and how the user model is revised as a result of the rules activation.

In the current prototype, the rules are activated at the start of each session, based on the user actions that are stored in the database during his last session. Therefore, the user's behavior is monitored during a session. Any user changes are activated instantly, such as the addition of an entire section of news. At this point the user model does not change. It will change in the start of the next user session, when the user modeling rules for the revision of the user model are activated.

Our approach has several advantages concerning both the efficiency of the system and the coherency of its behavior. Firstly, the statistics derived from the user behavior analysis within a whole session are more reliable and more significant than those regarding shorter browsing periods. Secondly, since the analysis of the user's actions and the revision of the user model are carried out at the start of each user session, the system does delay due to the performance of the revision task.

The user model revision is based on 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. This new distribution is stored as the revised user model in the user's database.

5 News retrieval.

The main goal of the retrieval part of the personal news agent is to visit periodically some Hellenic news sites and e-newspapers, to retrieve new stories and store the plain text of the article in the news database of the system. The retrieval part contains five agents and is a thread. This feature allows us to arrange the time interval between two subsequent executions. The general idea for the retrieval agent is to be executed periodically and retrieve new articles.

6 Personalization of Content.

The personalisation task is performed by two different agents. The first agent personalizes the content of the presentation: given the pieces of information in the user model, it decides which sections and news have to be presented. The second agent generates dynamically the web pages and is also responsible for the personalization of the presentation. However, in the current prototype we decided to maintain a standard format for the presentation, as we focused on the personalization of the content. This means that all the users see the same layout of the web pages, but various forms of personalization can be adopted on this aspect.

All the pages are generated dynamically, including the home page, which contains a list of the high-level sections that are considered relevant for the specific user. After that, a list of all the articles of the relevant sections follows. Each node of this list contains a small portion of the article and a link to the full text article. In the case of a previously read article, there is a message that the specific article was read and a link that allows the user to read it again.

Whenever the user opens an article, the corresponding page is generated by the system. The page is divided into two parts: The upper part, which contains the links for getting back to the page with the list of all the articles and the lower part that contains the text of the article.

7 Experiments.

The testing was performed with a small set of users, who were selected as representatives of different categories of readers, as they differ in terms of education income and occupation. There were selected a high school student, a university student, a network administrator, a university professor, an accountant, a civil engineer, a doctor, a salesman, an economic manager and a PhD candidate.

They were presented with the complete news section list and were asked to express a measure of their interest: high, medium, low or null. After that, they were asked to register and use PENA. As a result, PENA classified the user based on the predefined stereotypes, generated predictions and produced personalized pages with news. The result of this personalization process has been compared with the preferences expressed a priori by the user.

The selection of the system was judged compatible with the preferences of the user, if the system included all the sections that the user indicated as of high interest and not included any section of no interest for the user. 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 7 out of the 10 cases the system provided satisfactory results, while in 3 cases the system completely failed to satisfy the desired 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, since the system achieved high accuracy 70%, in the prediction of interesting news sections for the users.

8 Implementation details.

PENA is designed as a set of cooperating agents and implemented in a three -tier architecture. The system is implemented using Java, especially Servlets and JSP. The databases are implemented using MySQL and are accessed using JDBC.

9 Conclusions and Future Work.

The described system is in an early development and experimental stage and is part of a project that aims at the creation of an adaptive news site. There are three main goals of our future work. Firstly, alternative ways should be investigated for the initial user model. Secondly, news model adaptation can be achieved, by employing a further classification of the news, inside their high level classification that exists now. Finally, recommendations enhancement can be done by proposing articles based both on the user model and news stories similarity.

References

[1] Ardissono, L., Console, L., and Torre, I. (2001) An adaptive System for the personalized access to news. AI Communications 14: 129-147.

[2] Billsus, D., Pazzani, M., (1999) A personal news agent that talks, learns and explains. Autonomous Agents 1999, pp. 268-275.

[3] Joachims, T., Freitag, G., Mitchell, T. (1997) Web-Wacher: a tour guide for the world wide web. IJCAI, 1997, pp 770-775.

[4] Rich, E, (1989) User Models in Dialog Systems, chapter Stereotypes and user modeling, Springer, Wien.