Personalisation of information based on the concept of relevance using a user model

Amos A. David[1] adavid@loria.fr

David Bueno Vallejo[1,2] bueno@lcc.uma.es

[1] Laboratoire lorrain de recherche en informatique et ses applications (LORIA), Campus scientifique. BP 239.54506 Vandoeuvre Les Nancy Cedex (France)

[2] Dpto. Lenguajes y Ciencias de la Computación. Universidad de Málaga, Complejo Tecnológico, Campus de Teatinos, 29071 Málaga (Spain)

Abstract

In information retrieval systems (IRS), the user's interests and preferences are obtained in most cases implicitly through the user's interaction with the system. In our studies, we represent each user by a user model, which is composed of four parameters: the user's identity, his behaviour, the history of his past activities and his feedback. For each application of an IRS, we request the user to provide explicitly his main objective and provide reasons for accepting or refusing a solution given by the system in order to obtain a knowledge of the user's interests and preferences. Our proposals are implemented in a prototype METIORE for accessing publications of our research laboratory.

Keywords: user modeling, information retrieval, relevance degree, relevance feedback and personalised response

1. Introduction

Personalisation of information has attracted a lot of interest in the research fields that deal with the access of information. In information filtering for example, many approaches are based on the use of user profile where the hypothesis is that the user's need is relatively stable. In information retrieval systems (IRS), various types of stereotypes are used to represent categories of users, based on their interest and preferences. The user's interest and preferences are obtained in most cases implicitly through the user's interaction with the system.

The objective of our research is, like the above preoccupations, to provide the most relevant information to the user by focusing on two parameters in the user model: the *user's objective* and the *degree of relevance of a solution* as *explicitly* given by the user. This approach applies to IRS in general. In our user model, the parameter *objective* represents the user's main aim of using the IRS. In the existing systems, the most important task is to discover the user's objective through his requests and his evaluation of the proposed solutions. One approach is to ask the user to present it in natural language. This user's main objective may not be well defined initially but it constitutes an initial reference for the system. The user may redefine it. Some reasons given for not asking the user to provide this initial objective is that the user does not necessarily know it or that it may be discouraging particularly to casual users. However, through our observation of the dialog between users and librarians, we discovered that in no way does the request from the user to provide his objective constitute a handicap. On the contrary, it facilitates and accelerates the assistance of the librarian.

The second parameter, *relevance,* is present in most proposed user models but it represents different things for different applications and it is generally based on the degree of acceptance of a solution by the user. We have focused our study on the meaning of *degree of acceptance*. In fact the acceptance or the non-acceptance of a solution may be due to many reasons. We have chosen to represent the degree of relevance of a solution by the reasons given explicitly by the user for the acceptance or the non-acceptance of the solution. We request the user to specify explicitly his reasons for its acceptance or its refusal. It is also generally believed that two many questions may discourage some users. But we have noticed that these types of information that we request from the user do not constitute any handicap in the real world. On the contrary, it helps accelerate the discovery of what the user is searching for.

We present in the following sections a full detail of how these parameters are exploited in the prototype we have developed. Our prototype METIORE for managing the publications of our research laboratory LORIA, France contains about 10 000 references. The users are mainly the researchers of the laboratory and external users so that we may have a more representative sample of users for carrying out an efficient evaluation of the system.

2. The user model

The reasons for personalising the system's response relate to the various types of problem a user may have in the situation of information retrieval. We present below three important problems faced by the users: the problem of transforming his objective into the system's requests; the exhaustively of the system's solution in response to the user's request; the user's specific preferences.

Formulation of requests – Each IRS has its set of rules for formulating acceptable requests. The user must learn how to express his objective using the rules of the system. In some cases, the user obtains irrelevant response because his request was wrongly formulated. An example of problem is the use of boolean algebra for combining constraints to be satisfied by a request. The use of AND may be interpreted as the " and" in naturally language for " adding" objects while in IRS systems, it is used for the intersection of two sets. The user must, in this case, think in " sets" .

The number of the system's response – In the case of many results in response to the user's request, what determines what and how the responses should be presented to the user? The most frequently used approach to this problem is to calculate the degree of relevance of the solutions to the user's request. The solutions are then ordered in decreasing order of the degree of relevance. However, in some cases, this may not be sufficient for personalisation. For example, if a user formulates the same request for an objective, three times in three weeks, and once per week. Merely reordering the results according to the degree of relevance no longer reflects a real personalisation of the system's result because the past experience with the user's need was not taken into consideration.

The relevance of the system's response – The notion of relevance of a response depends on various factors that may relate to the user or to the system. However, personalisation seems to depend more on the factors that relate to the user. For example, the user's level of knowledge may determine his evaluation of a response as relevant or not. Some other factors concern the user's frequency of using the system and the frequency at which some of his objectives reoccur.

With these few examples of problems, personalisation may be viewed in two angles: the personalisation of the system's interface and the personalisation of the system's response. As regards the personalisation of the system's interface, we have tried to make our prototype as simple as possible while integrating the various types of possible interaction. We have focused our study on the personalisation of the system's responses.

In order to personalise the system's responses, there are two types of adaptation that we take into consideration: the adaptation of the user's request and the adaptation of the system's solution.

Adaptations of the user's request - The main techniques we use for adapting the user's request are regular expressions, vectorial research and the use of thesaurus. Regular expressions are used for word truncation, substitution and replacement. Vectorial research is used for calculating the degree of relevance of a request and the description of an object. The objects in our prototype METIORE are documents. But the objects may be other things in other applications. We use thesaurus for representing the domain knowledge. This allows us to reformulate the user's request by substituting some of the terms of the request using related terms in the thesaurus.

Adaptation of the system's response – The results obtained as a result of the transformation of the user's request is further filtered using the information available in the user model. As we present in the following section, we have an individual model for representing a single user as well as a global model for representing the users in general. In a case where the user is using the system for the first time, the global model is used. But in a case where the user has an individual model, his individual model is used.

3. What to represent in the user model?

Our user model contains four principal parameters: the user's identity, his behaviour, the history of his activities, and his feedback.

3.1 The user's identity

This parameter is needed for identifying the individual user model. It should be possible to differentiate each user. A major problem with the use of this parameter is the privacy of each user. In our prototype, a user has no access to another user's individual model. In the generic user model, users are anonymous. This identity may be obtained automatically in operating systems like Unix but in other systems, the user must provide his identity. A user has only one identity.

3.2 The user's behaviour

This parameter represents the user's behaviour during the various process of information research. In our prototypes, the user's types of behaviour are obtained automatically through his interaction with the system. The types of behaviour that we represent for each user are the following:

1. The kind of response (*general*, accepted by everybody, or *individual*, accepted only by him)
2. Maximum number of answers evaluated
3. The types of solutions accepted (solutions derived from only the terms used in his request or from related terms using a thesaurus)
4. The types of feedback provided for each answer

The types of feedback that we observe for each user in the fourth type of behaviour will be presented latter in the paper. These types concern the reasons why the user accepts or refuses a response proposed by the system. The first type of behaviour helps us determine what type of response the system should favour. The second type of behaviour helps us determine the average number of response that should be presented to the user. The third type of behaviour helps us determine what type of strategy to adopt for reformulating the user's request. A consistency of this type of behaviour indicates also the user's level of knowledge. For example, if the user refuses systematically the solutions derived as a result of transformation, we may consider that the user knows exactly what he wants.

3.3 The user's history of activities

We consider this parameter as one of the particularities of our user model. All our inferences are based on the facts collected on each user from their past activities using the system. This approach is similar to the concept of Case Based Reasoning (CBR) in artificial intelligence [9]. CBR methodologies are proposed for managing voluminous cases, indexing cases to facilitate the retrieval of similar cases, use cases for deriving models, and integrating new cases after evaluating solutions used for past cases that are proposed to the current case.

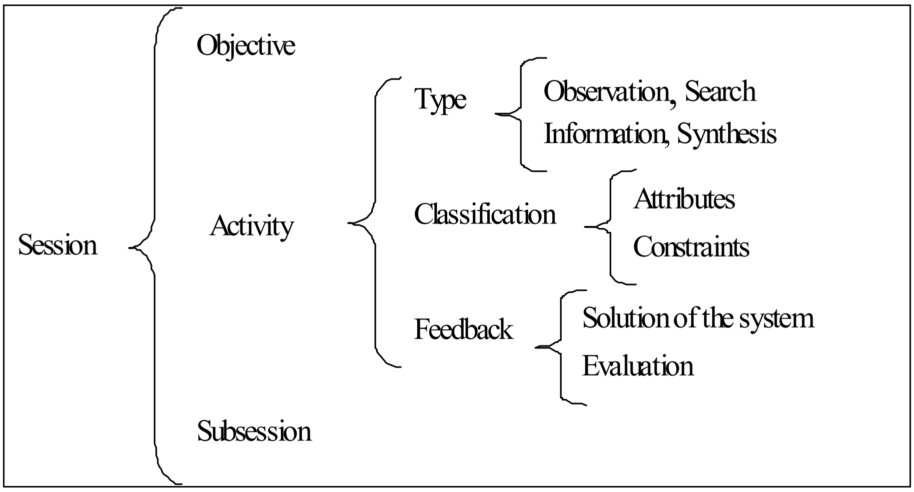

Fig.1 Structure of the history of activities

We have chosen to represent the user's activities based on *sessions*. This allows an efficient representation of the user's activities. As presented in fig.1, a session is composed of an *objective*, a set of research *activities* and as set of *subsessions.* A subsession is a session that is linked with the main session or to another subsession. For a session, the user must express explicitly his main information need in form of objective. In most existing systems, the most important task is to discover the user's objective through his requests and his evaluation of the proposed solutions [3]. One approach is to ask the user to present it explicitly in natural language. This main objective may not be well defined initially but it constitutes an initial reference for the system. The user may redefine this main objective to form subsessions. Some reasons given for not asking the user to provide this initial objective is that the user does not necessarily know it or that it may be discouraging particularly for casual users. However, through our observation of the dialog between users and librarians, we discovered that in no way does the request from the user to provide his objective constitute a handicap. On the contrary, it facilitates and accelerates the assistance of the librarian.

The user can employ a set of predefined types of interaction for expressing his information need. Each of this type of interaction is considered as activity. The set of interaction is provided by the method that we defined, called classification with constraints. In this method, "classification" indicates one or two attributes to be used for a cross analysis of the information base (IB). "Constraints" is a set of constraints that must be satisfied by objects of the IB in order to be included in the system's response. The constraints are expressed using any attribute of the objects of the IB. For example:

Classification: *author, country*
Constraints: *(language = english; year >= 1990; keyword = "user modeling")*
In this example, the user wants to obtain the frequency of co-occurrence of authors and countries. He however limits his interest on publications in English that are published as from 1990 on the subject "user modeling".

This method provides a way of formulating the classical request using constraints. It also provides a way of global analysis of the IB. A classical request will not provide the distribution of authors over the countries, which is provided by this method.

The activities of the user is classified the activities into four categories. The evolution of the user within each category allows us to monitor his behaviour and his level of knowledge in the domain of application. The four categories are defined as follows:

Observation - The user employs only one attribute for classification and without constraints. We interpret this type of activity as the user trying to discover the content of the IB.
Information - The user employs only one constraint. We interpret this as the user trying to acquire the vocabulary within the domain of application of the IB.
Research - The user employs several constraints without classification. We interpret this as a user having knowledge in the domain of application. He is applying his knowledge for solving his problem.
Synthesis - The user employs one or two attributes for classification, and with constraints. We interpret this as the user wanting to obtain a global analysis of the IB. We suppose that the user has important knowledge of the domain of application.

The interpretations we give to these types of activity represent our first step toward understanding how the user's activity corresponds to his level of knowledge in the domain of application. These categories of activity also provide us a means of adapting the responses of the system to the user's behaviour. A user that no activity that corresponds to the category "research" or "synthesis" may be considered as somebody with little knowledge in the domain of application. This information is used to determine the best way of transforming his request, such as using the thesaurus. For example, for the above user, we may decide to substitute a term used in the constraints with more specific ones rather than using more abstract terms in the domain of application.

3.4 The user's feedback

In all the types of activities, we request the user to evaluate the solutions proposed by the system. This enables us to associate the solutions and the user's evaluation with the user's main objective. The user's feedback is also necessary in order to understand why a solution is accepted or refused. We have focused our studies on the reasons why a solution may be accepted or refused. The acceptance or the refusal of a solution may be due to: (1) the solution is accepted because it corresponds to the user's objective; (2) the solution is rejected because the user already knows the solution; (3) the solution is rejected because the solution does not correspond to the user's objective and this reason is confirmed; (4) the solution is rejected because the solution does not correspond to the user's objective but this reason is not confirmed; (5) the solution is rejected but the user does not know why. Reasons (3) and (4) may arise in the cases where the system reformulates the user's request. Other reasons may be given in applications where sentiments may constitute reasons for the acceptance or the refusal of a solution. The following reasons may be given : (6) the user *hates* the solution ; (7) the user *does not like* the solution ; (8) the user *likes* the solution ; (9) the user *loves* the solution. The above types of feedback are activated each time that the user analyses a response.

Another type of observation we make on the user's behaviour is the frequency of a particular objective or request. Notice that requests (classification with constraints) are used for expressing objectives. If we detect that the user repeats the same request many times for a particular objective, the system initiates a dialogue to understand why the request is repeated despite the past solutions. In response to this demand by the system, the user may give the answers in the table below. The associated reaction or response of the system to the answers is also given in the table. The system's demand is as simple as "*why do you repeat the same request many times for this same objective?"*

User answer	**System action**
The solutions given didn't	The system explains the transformations on the user's request if there

correspond to my request	is any
I need more answers	The system uses the thesaurus to enlarge the user's request.
To remember	The system reorders the past solutions classified, by the types of reasons given for the acceptance or the refusal of the past solutions
Others	In the context of co-operative information research, an expert may intervene to assist the user.

Table 1: Types of interaction activated when the user has problems

4. Using the our model

We presented in the preceding section what we represent in our user model and how we acquire the needed information for this representation. We present in this section the main procedure of how the user model is exploited (cf. fig. 2). We differentiate the casual users from the regular users. We use a generic model to represent all the activities of all users and a specific model for representing the activities of a particular user.

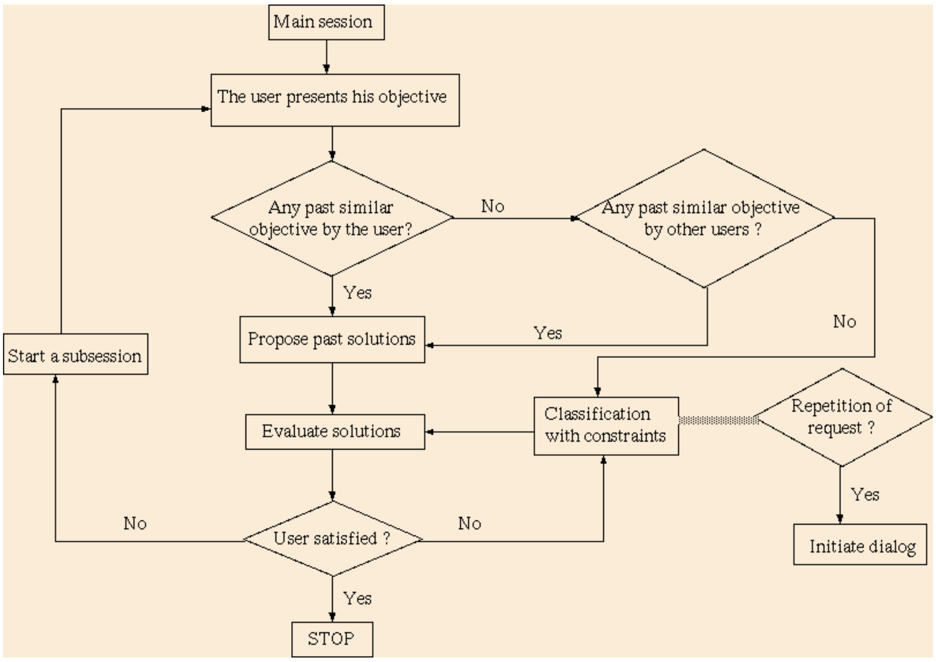

Fig. 2 How the user model is exploited

4.1 Casual users

Casual users do not use the system often. This means that the system does not have reliable information on the user for making inferences on his behaviour or preferences. Our approach for this type of user is similar to approach used in stereotype [7][8]. We start by using the generic model for the interpretation of his requests. When he becomes a regular user, we change to the specific model. What we use in the generic model is the following quadruple:

(Objective, Solution, Evaluation, Frequency)

Solution contains the response that the system has given to a user for *Objective*. E*valuation* represents what kind of feedback that the user gave for the solution in a range between, 1 and 5, each value representing the reason of acceptance or refusal of the solution. *Frequency* shows the percentage at which the type of evaluation is given for *Solution* in response to *Objective*.

Objective represents the user's main information need, and is given explicitly by the user in natural language. With a new objective, we calculate the **factor of similarity** with an existing objective. The method we use presently for calculating this factor is by indexing the objective with keywords. We use vectorial research technique to calculate the factor of similarity. This factor may be fixed either by the user or predefined in the system.

Another factor that we calculate for adapting the system's solution is what we call **the factor of tolerance,** using *Frequency*. This factor represents the percentage of times *Solution* is chosen for a particular type of evaluation and for a given factor of similarity for *Objective*. This factor may also be predefined in the system or given by the user. In our prototype, this factor is fixed at 60% for casual users while regular users have the possibility of choosing the percentage they want.

4.2 Regular users

Instead of using a generic model as we do for casual users, we use specific model for regular users. The element of the model that we user may be represented by the following quintuple:

(User, Objective, Solution, Evaluation, Frequency)

One major difference here is that every inference and interpretation is based on the past activities of the particular user. The factor of similarity and the factor of tolerance are calculated the same way as for casual users. For casual or regular users, we use the information acquired on them for determining whether to include a solution or not and how the solutions should be ordered.

4.3 Request analysis

As presented in section 3, the system also observes the frequency of request for a particular objective. We represent this observation by the following quintuple:

(User, Objective, Request, Frequency)

Frequency represents how many times the user employs the same request for the same objective which corresponds to what we call *factor of repetition*. According to this factor, we decide whether or not to initiate a dialog with the user as described in section 3.4.

5. Implementation in the prototype METIORE

Our proposals have been implemented in the prototype METIORE. The prototype is used for accessing information on the publications of our research laboratory LORIA, FRANCE. The prototype presently contains about 10 000 bibliographic references. The users are researchers from the laboratory and also external users like students, lecturers, etc.

6. Conclusions and future works

We think that our architecture is quite general and useful in many kinds of IRS. The architecture is independent of the system on which an application may be developed. We have used our proposals in another prototype, STREEMS, used for managing data on trees authorised by European union for reforestation. The European union sponsors the project within which this prototype was developed.

We continue our work on the study of the various parameters and factors in order to improve the personalisation of the systems responses. We continue with the experimentation of our prototype METIORE in order to validate and improve our proposals.

7. References

[1] Kay, J. " Vive la difference! Individualized interaction with users" , 1994

[2] Extensible Markup Language(XML). http://www.w3.org/XML/

[3] Spink,A., Ford, N., Wilson, T. " Modeling Users' Successive Searches in Digital Environments" , D-Lib Magazine 1998.

[4] Amos David, "Modelisation de l'utilisateur et recherche cooperative dans les systemes de recherche d'informations", ISKO '97, Lille(Francia)

[5] " Hanks, S., Grinden, L. " Interactive Assessment of User Preference Models" , UM97

[6] Mei-Mei Wu, Saracevic, T., Spink A.," Users and Intermediaries in Information Retrieval: What are they talking about" , UM97

[7] Alfred Kobsa " User Modeling: Recent work, Prospect and Hazards" , Report 1993

[8] Kay J., Wilson, T. " Lies damned lies and stereotypes: pragmatic approximation" , UM94

[9] Kolodner Janet, Case-Based Reasoning, Morgan Kaufmann Publishers, Canada, 1993

[10] Douglas W. Oard, " The state of the Art in Text filtering" , International journal of User modeling and user-adapted interaction, Vol 7, No 3, 1997

[11] Bhavani Raskutti, Anthony Beitz and Belinda Ward, " A feature-based approach to recomending selections based on pas preferences, International journal of User modeling and user-adapted interaction, Vol 7, No 3, 1997