Minimal Knowledge Anonymous User Profiling for Personalized Services

ALFREDO MILANI, CHIARA MORICI, RADOSLAW NIEWIADOMSKI
Department of Mathematics and Computer Science
University of Perugia
Via Vanvitelli 1, 06123 Perugia
ITALY

Abstract: - The paper presents a solution to the problem of application of user profiles for anonymous internet users. The basic assumption is that only minimal knowledge about the user is given, i.e. information such as user session, user tracing and clickstream analysis is not available. This situation is of great interest because it characterizes most internet users, such as user of search engine. In the typical case the user is described only by IP address, date/time of connection and keywords. The proposed architecture is based on the use of predefined profiles and the computation of fuzzy similarities in order to match the user observed with appropriate target profiles. The proposed model for user profiling in presence of minimal knowledge has many applications like generation of banners for online advertising, dynamical web pages for public services etc. The notion of fuzzy similarity presented here is based on the theoretical framework of the Łukasiewicz structure; it guaranties the correctness of the approach.
A prototype implementation of a banner engine is finally presented and discussed.

Key-Words: - User profiling, Fuzzy similarity, Soft computing, Search engine, Words matching, E-commerce

1 Introduction

It is commonly known that the most efficient solution to capture user's attention is to collect in advance as much information as possible about him. In the case of huge number of portals this problem seems to be "quite easy" to solve, because this kind of services usually requires some mechanism (like login procedure) to identify users. Systems based on authorization have the opportunity to collect systematically information about user either by using questionnaires or by tracing his choices (like in clickstream analysis). In the literature [5] is also described a method based on server log file analysis for session reconstruction in case when neither authentication procedure nor cookies are used. One of the most widely used technologies for building personalized service is collaborative filtering. This requires to gather data about (anonymous) users transactions in order to find some of them who have similar behavior to the given one. In [1], i.e., it is presented a system which is based on multidimensional ranking of the content which is then the basis for user classification. Furthermore [3] presents robust analysis of user profiles definition based on web log file and clickstream data.

Unfortunately there are many internet interactions which don't offer the possibility of solid observation of users behavior. To this kind of interactions belong all those which are based on two steps request-response pages schema. Anonymous user uses the service only once and after having received the rewarding response he usually goes ahead without coming back to start-point service. So only one HTTP-protocol request is given. In this kind of situation only some partial personalization could be possible because of sparse data. We have called this situation *minimal knowledge hypothesis*.

In the next section the framework for proposed solution and possible applications are presented. SECTION 3 recapitulates the theoretical background stated also in [4] and SECTION 4 is devoted to describe the decision algorithm based on fuzzy similarity. The SECTION 5 is dedicated to present the application for advertisement campaign. Finally future extensions in SECTION 6 are discussed.

2 User Profiling with Minimal Knowledge

In *minimal knowledge hypothesis* we have only current HTTP request. The general idea of the algorithm is to, basing on possessed information, try to match concrete user query to one of a set of profiles predefined for the given problem. It is made by measuring the level of similarity degree. User request and profiles for potential users of service have to be defined using similar criteria and data types. The difference is that while data from user request is usually "crisp" value, profiles are described using some sets, intervals or fuzzy sets.

2.1 Data from User Request

It is possible to obtain from HTTP request some useful information like the date and time of an action, preferable language of response or the IP number. We can also define (if needed) some variables which do not come directly from the data but depend on them like boolean variables "foreigner" or "netscape-user".

Most of HTTP requests contain also some user specific personalized data - something which (even partially) describe him. In fact, most of HTTP requests are queries or decisions made by a concrete person and which express his interests. This personalized part of the query often could be viewed as a set of key-words. They can be put directly by user itself (like in the case of search engine or train reservation service) or obtained in other way (web pages keywords).

All those with user key-words and eventually other variables which are based on values of HTTP's POST method parameters (like language or domain restriction in the case of search engines) will determine input data.

2.2 Data Representation in Profiles

To make an association of any user query to one of n profiles it is necessary to express the same properties which were singled out in user query in terms of intervals or sets. The method to present hardly comparable (non quantitative) information to make the matching then will be introduced. We present two examples, but many other variables which one can define for a particular problem can be described in similar way.

Lets suppose that user request is described by query's date/time and IP number. Then also profiles have to contain appropriate information, but it has no sense to enumerate explicitly all possible time values of connection for given profile. Instead date/time constraints will be specified in the profiles by some fuzzy sets. In details date/time profile constraints are described in dual way: the day of the week is fixed when the time is represented by a set of trapezoidal time intervals associated with a day of the week. Those trapezoidal membership functions can be defined as:

$$tr(b,c,x)\begin{cases} x-b-\frac{1}{c-b} & a \le x \le b \\ \frac{1}{c-b} & b < x < c \\ c+\frac{1}{c-b}-x & c \le x \le d \\ 0 & x < a \quad or \quad x > d \end{cases}$$

$b,\ c \in [0,\ 24)$ and $a = b$ - $\frac{1}{c-b}$, $d = c + \frac{1}{c-b}$.

From the IP number it is possible to obtain the name of the country from which the connection is made. Using some statistical sources we can find a lot of particular and useful information which depends on location data. The decision about what properties are important is taken when defining profiles and it depends on the problem domain. That information is not directly inserted into profiles, but profiles are described by a set of countries.

2.3 Applications

Once, having the tool which will be able to make even partial profiling of anonymous users without creation user session but relying only on *minimal knowledge hypothesis*, it would be difficult to enumerate all possible application of it. Every web site at the moment of first interaction with the user could be on that list. Starting from the end the final effect is, generally speaking, the return information adjusted to the recipient - anything it will be - personalized portal site (both the subject and the esthetics), appropriately selected advertisement information, piece of music or even other internet user (in the case of interactive services). The most obvious application of it can be profiled advertisement or information campaigns for services like search engines. In this case key-words are given explicit. In more general situation key-words can derived from web page key-words (appropriate tag) which the user has arrived from. However this solution is based on some doubtful assumptions like that he was attracted by last visited web site. More interesting and effective solution could be obtained by creation a set of web pages belonging to one or a set of portals. Then every link to new information on any page of it could

be associated with some key-words chosen in static or dynamic way. The target page could be personalized depending on those key-words and other HTTP derived data. One can imagine a chat service in which the user given first some short information (like favorite film, place, job etc.) about himself, one from many chat-rooms is chosen for him then.

3 Fuzzy Similarity in Łukasiewicz Structure

We briefly recall the basic concepts of fuzzy similarity pointing out its relationship with Łukasiewicz structure. The detailed analysis of it can be found in [4]. The use of fuzzy similarity [6] is quite natural in order to evaluate and compare user profiles since it is a many-valued generalization of the classical notion of equivalence relation. More over fuzzy similarities and pseudo-metrics are dual concepts, as shown in [2].

Łukasiewicz structure is the only multi-valued structure in which the mean of many fuzzy similarities is still a fuzzy similarity. This property guarantees the correctness of the proposed algorithm for user profile comparison.

As proved in [4] fuzzy similarities can be used to compare pairs of objects. The various properties of objects can be expressed through membership functions f_i valued between [0,1]. The idea is to define *maximal fuzzy similarity* by computing the membership functions $f_i(x_1)$, $f_i(x_2)$ for comparing the similarity of objects x_1 and x_2 on each property i and then combining the similarity values for all properties.

Definition 1. Maximal fuzzy similarity. Since Łukasiewicz structure is chosen for membership of objects, we can define the maximal fuzzy similarity as follows:

$$S < x_1, x_2 >= \frac{1}{n} \sum_{i=1}^{n} (f_i(x_1) \leftrightarrow f_i(x_2)).$$

where x_1, $x_2 \in X$, f_i membership functions, $i \in \{1,...,n\}$ and $\leftrightarrow$ *(double residuum)* is the Łukasiewicz structure equivalence relation defined by:

$$a \leftrightarrow b = 1 - max\{a, b\} + min\{a, b\} = 1 - \mid a - b \mid$$

Non-zero weights can be associated to the different properties in order to express their different contribution to the similarity of the objects.

Definition 2. Weighted fuzzy similarity. A weighted fuzzy similarity can be defined by taking a weighted average on the single properties comparisons:

$$S(x_1, x_2) = \frac{\sum_{i=1}^{n} w_i (f_i(x_1) \leftrightarrow f_i(x_2))}{\sum_{i=1}^{n} w_i}$$

Note that this is still resulting a fuzzy similarity.

4 Algorithm for User Profile Matching

The goal of the algorithm is to determine the most appropriate profile for a given user. It is made by evaluating similarities between the observed data and a set of possible profiles. The set of possible matching methods contain banal case of comparison between to boolean variables, matching crisp value with fuzzy value and finally two non-quantitative values (like two different words). Direct definition for similarity relation is possible for variables which have only few possible values like boolean variables by enumerating all possible values for cartesian product. More interesting is the case in which there are many or even infinite possible values for the variable.

Basing on some typical examples we present now the computation model for establishing the level of similarity between them. Lets consider three different data types which usually describe user profiles: keywords, connection date/time, IP/user location. The similarity of other variables distinguished from user query and profile can be established in similar way. Finally the similarities values previously computed are finally composed to obtain the final result; the composition is still a fuzzy similarity [4].

Ontologies. The most important information for profiling algorithm are usually user key-words. Neither single words nor their concatenation they are not quantitative values - they can't be directly compared without using any ontology. Ontologies are necessary in order to classify and compare all those keywords according to their semantics. We suppose that the classification structure is a tree in which all keywords (leaves) are ordered in some categories and subcategories (nodes). Some keywords could be repeated as different leaves representing in this way two meanings of the keyword. The classification structure has a root node, which corresponds to a set of all possible classification paths. The classification tree represents a hierarchy in

which more generic categories are closer to the root. We also assume that any keyword has at least a classification path and for every pair of keywords is possible to find a common path in the classification tree.

4.1 Evaluating Keywords Similarity

The similarity between the set of *observed* keywords and the set of *target* keywords is based on evaluating similarity between pairs of keywords. We assume that a set $F=f_1,...,f_n$ of classification functions are available, each function f_j is such that given a keyword K_i, $f_j(K_i)=v_i$, returns v_i, the path from node K_i to the root in the classification tree represented by f_j.
A path is an ordered sequence of nodes $v_i=(n_0,...,n_k)$ where n_0 is the root and n_k is K_i, a path in the classification tree represents a set of categories/subcategories which define a particular meaning of a keyword. By extension $F(K_i)=v_i$ returns the set of classification paths for keyword K_i.
In addition we define:
- L the longest path from any leaf to root in the classification tree represented by F,
- l_{v_i} denotes the length k of path $v_i=(n_0,...,n_k)$,
- $l_{v_{ij}}$ denotes the length of the common path (i.e. the number of common arcs).

Moreover we admit that every keyword K_i and every v_i of K_i can have associated some weights w_{K_i} and w_{v_i} which express the importance of given keyword in the definition of profile and "the importance" of the certain meaning of keyword. Those values can depend for example from user origin (in case in which some key-words have two different meanings in two different languages). Only for the legibility and without losing the completeness of the solution those weights are omitted in formulas presented below.

Path similarity. The path similarity between v_i, v_j is defined as:

$$S_p(v_i, v_j) = \frac{1}{2L}(2L - d(v_i, v_j))$$

where $d(v_i,v_j)$ can be seen as a "dissimilarity":

$$d(v_i, v_j) = (l_{v_i} - l_{v_{ij}}) + (lv_j - l_{v_{ij}}) = lv_i + lv_j - 2l_{v_{ij}}$$

$S_p(v_i,v_j)$ is a similarity since we can prove that $d(v_i,v_j)$ is a pseudo-metric.

Keywords Pair Similarity. Since every keyword is classified by a set of classification paths, the similarity $S_K(K_i,K_j)$ between two keywords K_i and K_j is defined as the maximum of $S_p(v_i,v_j)$ over each $v_i \in F(K_i)$ and $v_j \in F(K_j)$:

$$S_K(K_i, K_j) = max S_p(v_i, v_j)$$

Keywords Set Similarity. Let U be the set of keywords *observed* in the user query $U=(K_1,..,K_m)$ and every profile P_i is described by a *target* set of keywords $W_i=(K_1,..,K_n)$. Then a straightforward solution for determining the best matching set W_i for U is to consider the mean value of $S_K(K_i,K_j)$ over every pair of keywords of $K_i \in W_i$ and $K_j \in U$.
The adoption of a mean value reflects the intention that all keywords contribute to represent the *meaning* of a profile, on the other hand redundant or similar keywords do not contribute to increase the mean.

4.2 Evaluating Connection Time Similarity

Lets consider now the case of matching based on date/time of connection which derive from HTTP protocol. The similarity is made here between crisp value from the user query and fuzzy value of a profile.
The date/time constraints associate with a profile P_i are represented by a set of trapezoidal time intervals.
Given the user local date/time x_u and given, for every profile P_i, the date/time intervals in the profiles $[b_{ik}, c_{ik}]$ *(where* $1 \leq k \leq m$, m *- number of intervals of* P_i*)* the *date/time similarity* degree is computed. First the greatest value of $tr(b_{ik},c_{ik},x_u)$ is computed (tr_{x_u}) for the user observed time x_u along all date/time intervals in all profiles:

$$tr_{x_u} = \max_{ik} tr(b_{ik}, c_{ik}, x_u),$$

Let b and c define the interval such that $tr(b,c,x_u)=tr_{x_u}$. The second step consists in computing the similarity degree between the membership of x_u for each time interval $[b_{ik}, c_{ik}]$ of each profile P_i and the best membership of x_u in all intervals, i.e tr_{x_u}. This similarity degree is computed by *maximal fuzzy similarity* as in DEF.1 with $n=1$:

$$S_T(tr(b_{ik},c_{ik},x_u),tr_{x_u})=tr(b_{ik},c_{ik},x_u) \leftrightarrow tr(b,c,x_u)$$

4.3 Evaluating user location

The information about country of user location

can be easily obtained from the IP address. At the same time profiles are described by a set of countries. "Comparing countries" is possible only by establishing first some quantitative criterion, so additional information and properties are associated to each country, properties like annual income, population, religion etc. The similarity between countries is based on the similarity of the quantitative properties associated with them. In order to make a consistent comparison among values of properties in the different countries an ordering is induced by appropriate fuzzy sets; n fuzzy sets μ_j, associated to the n relevant properties for countries, are defined for each country

$$\mu_j(C) = \frac{p_j(C) - \underline{p}_j}{\overline{p}_j - \underline{p}_j},$$

where $\mathrm{p}_j(\mathrm{C})$ returns the value of property j for country C, *j=1...n*, n number of properties, and:

$$\underline{p}_j = \min_C p_j(C), \quad \overline{p}_j = \max_C p_j(C).$$

The memberships $\mu_j(C)$ of the different properties are then compared and weighted using *weighted fuzzy similarity* (DEF. 2) thus obtaining the global degree of similarity between countries for profile P_i:

$$S_C(C_1, C_2) = \frac{1}{\sum_{j=1}^{n} w_{ij}} \sum_{j=1}^{n} w_{ij}(\mu_j(C_1) \leftrightarrow \mu_j(C_2))$$

where C_1, C_2 denotes the countries to compare and n is the number of considered properties. Weights w_{ij} are defined for every profile in order to point out the relevance of a property in the context of a profile.

4.4 Combining Similarities

Once having n similarity values independently evaluated for different types of data finally it become possible to combine them in order to find the target profile which best matches the current user.
Let m_{ij} be the value of similarity j for observed data and a profile P_i, then a decision function can be defined:

$$up_i = \frac{1}{\sum_{j=1}^{n} w'_{ij}} (\sum_{j=1}^{n} w'_{ij} m_{ij})$$

Again w'_{ij} are weights defined for every profile, they allow to express the relevance of the type of observed data for determining a certain profile.

Finally, the profile most similar to the user observed data can be determined by considering maximum value *(up)* of up_i.

Generic profile. The last improvement to the mechanism considered so far is the use of an generic profile if system finds difficulty to choose the profile for a user. Not recognized user (or not recognized enough) will correspond to the situation in which *up* value is very small. ($up < upvs$, where *upvs* is a fixed constant). It means that user's data doesn't match any particular profile definition. The generic profile accounts for this kind of situation ensuring the completeness of the algorithm.

5 User Profiling for Advertising on Search Engine

A typical application of our *minimal knowledge hypothesis* can be an advertisement campaign for search engine. Typical search engine interaction is based on two steps request-response pages schema, and user would not have any reason to accept other longer ways to receive response (for example through login procedure). Therefore, in the case of search engines we have to admit that in the worst situation we have not much information about the user than any the information contained in the query he has submitted.
Lets the user query for matching algorithm be based only on three variables:
· user's request string constructed from words and some operators,
· query's date and time,
· IP number, i.e. user location/country.
We use as bridge between advertiser and user of search engine a set of profiles. They have to be defined using similar data types as user request and should correspond to some typical patterns of persons who could be interested on banner's topics. It is important to point out that there is unlimited number of user queries which can be constructed by using any words and unique names from all world languages.

5.1 Ontology

In order to have an ontology for keywords classification, one can use search engine itself; because those services are able to categorize every user query. We have used the project DMOZ *Open Directory Project* [7] which is the largest categories directory and search engine edited by human. An ontology is freely accessible

both in RDF format and as on-line search engine. Primarily it's designed as a web directory but we propose to use it as hierarchical structure which classifies words by assigning them available classification paths. Unfortunately it's not ideal for our purpose, because used classification not always consider some important semantic aspects. It is considered in future extensions the composition of various ontologies to obtain more suitable results.

5.2 Choosing banners

New element is a set of advertisement information to be associated with profiles. We assume that every time the user makes query, one of n banners has to appear on response page. Every of them is described by a set of keywords and a set of weights w_{ij} - where w_{ij} express accuracy of banner B_j for profile P_i.
Now we can define the problem as finding the profile which fits in the best way a user request and after matching the right banner with chosen profile. The first part can be resolved using algorithm presented in [4.1] - [4.3] with $n=3$ in maximum weighted similarity formula (DEF. 2). Once knowing the profile, the choice of the banner is made by looking for a maximum value of some formula along banner weights w_{ij}. Moreover, the looked for value should also consider banners deadline. The frequency of banner showing would be a function of advertisement expiry date. At the beginning weights are predefined. After that they have to be modified depending on user (as a profile representative) interest of advertising information (feedback effect). Moreover the banner choice is parameterized by some random value and "randomization" decreases with the number of computation for B_j. In this way we avoid situation in which user receives every time the same banner and so the correctness of the algorithm is assured.

Step zero matching. Moreover, in the *step zero* algorithm tries to make direct matching between keywords of banners and request avoiding middle-tier of profiles. It would correspond to the situation in which user's keywords "are very close to" the description of banner. For this purpose is used the same ontology and the algorithm check all possible pairs of keywords (user's keyword, keywords of banner) looking for some with distance between each other less or equal to d.

6 Conclusion

The proposed algorithm is a part of more robust solution which is planed to realize. It is considered to expand the system by the use of hierarchical relation between profiles. Profiles with different level of accuracy can be considered diversely. The system can choose the profile more detailed if the decision can be made without much risk. We have just mentioned this problem by creating generic profile for not recognized user.
It is planed also to compose different ontologies and use query languages with it. We wish also at least semi-automated way to be adopted for defining profiles. By the moment both the number of the profiles and weights are set manually. Establishing profiles can involve some data-mining technics. It is also planed to extend the use of the solution presented in this paper to other problem domains.

References:

[1] R. Burke: *Semantic ratings and heuristic similarity for collaborative filtering*, AAAI Workshop on Knowledge-Based Electronic Markets, AAAI, 2000.
[2] P. Luukka, K. Saastamoinen, V. Könönen, E. Turunen: *A classifier based on maximal fuzzy similarity in generalised Łukasiewicz structure*, FUZZ-IEEE 2001, Melbourne, Australia.
[3] M.J. Martin-Bautista, D.H. Kraft, M.A. Vila, J. Chen, J.Cruz: *User profiles and fuzzy logic for web retrieval issues*, Soft Computing, Vol. 6, No. 5, August 2002.
[4] C. Morici, R. Niewiadomski: *A framework for a personalized advertising on Web based on maximal fuzzy similarity in Łukasiewicz structure*, to appear.
[5] M. Spiliopoulou, B. Mobasher, B. Berent, M. Nakagawa: *A Framework for the Evaluation of Session Reconstruction Heuristics in Web Usage Analysis*, INFORMS Journal on Computings 15.
[6] E. Turunen: *Mathematics behind Fuzzy Logic*, Advances in Soft Computing, Physica-Verlag, Heidelberg, 1999.
[7] Web resources for Open Directory Project: *http://dmoz.org/about.html*.