International Journal of Computer Science & Applications
Vol. 3 Issue 3, pp 65 - 80

The Recommendation Mechanism in an Internet Information System with Time Impact Coefficient

Dariusz Król, Michal Szymanski, Bogdan Trawinski
Institute for Applied Informatics
Technical University of Wroclaw, Poland
dariusz.krol@pwr.wroc.pl, michaleq@wp.pl, trawinski@pwr.wroc.pl

Abstract:

In this paper we propose two generic mechanisms implemented in a cadastre internet information system. The first one is the list of last queries submitted by a given user and the second one is the list of page profiles recommended to a user. The idea of page recommendation is based on the concept of a page profile which represents a system option, type of retrieval mechanisms and search criteria. The calculation of rank values for page profiles is based on the usage frequency and the time impact coefficient. A recommended page is selected by a user from a list facilitates and accelerates his searches by moving him directly to the chosen option page with search form filled with the most expected criteria values. As an additional complementary mechanism the list of last submitted queries is available to each user.

Keywords: information filtering, recommendation systems, user assessment.

1 Introduction

Due to the growth of the Web systems both in the research and commercial area, the users impose new methods for predicting their needs. Systems that adapt its behaviour to their users are called user adaptive systems [5, 10]. One of the ways of helping the user is to adapt the interface elements. The other way is looking for similarities between a new user and past users [1].

Adaptation to the individual user is fulfilled by personalization [8]. Starting in the late 1990s personalization became very popular. First personalized services are based on static questionnaires that users fill out in order to make use of new capabilities. Nowadays we use machine learning techniques for adaptation [4]. The personalization process usually consists of (a) the collection and pre-processing of Web data, including content data, structure data, usage data and user profile data, (b) the analysis and discovery of correlations between such data, (c) the determination of the recommendation methods for hyperlinks [11], queries [3], products [9] and user interface [17].

The means to analyse the Web data listed in [12] include demographic filtering (DF), collaborative filtering (CF), content-based filtering (CBF), case-based reasoning (CBR), rule-based filtering (RBF), Web mining (WM) and hybrid approaches (HA). DF systems use the information stored in the profile that contains many different demographic attributes. CF systems collect visitor opinions on a set of objects, using ratings provided by the users to predict a particular user's interest in an item. The ratings may be implicit or explicit. The main idea is to compare the user model of an active user with the

previous user in order to cluster similar users. Many studies have shown that CF methods produce recommendations with usefully high accuracy. Improving the performance by optimizing feature weight using genetic algorithm is presented in [14]. CBR paradigm based on past experiences allows solution adaptation that leads to fine-grained tuning of historical solutions toward the new one. CBF systems track user's behaviour and take descriptions of the known content to learn the relationship between a user and new items. In RBF users are asked to answer a set of question and a result is customized for their needs. WM specifies three domains: Web content mining, Web structure mining and Web usage mining. The last domain is the most popular in the area of recommendation [19].

Web usage mining, also known as Web log mining, aims to discover user patterns from the data stored in server logs or browser logs while surfing the Web system. The mined knowledge can improve the design of Web pages, and develop adaptive usage scenarios more efficiently and effectively.

A hybrid approach to recommendations combines aspects of DF, CF, CBR, CBF and WM. Users evaluate the documents and provide feedback for the system and the system knows more.

The trend towards increasing profitability makes voice-browsing very promising approach to deliver proper content for devices such as mobile phones and PDA. Voice web service presented in [13] combines linguistic knowledge, Voice XML and Web ontology into a personalized recommendation system.

Traditional recommendation techniques identified in [12] such as non-personalized, attribute based, item-to-item correlation and people-to-people correlation have been applied for Web recommendations. The method proposed in the paper [11] based on direct and indirect association rules uses information about user behaviour to local Web pages. The rules discovered in this process estimate helpfulness of one page to make ranking lists of all visited pages. In papers [9, 20] the authors present a model which uses the visiting time and frequency of pages without considering the access order of page requests in use sessions. To capture the relationships between pages they extract information from log data.

Knowledge discovery usually is executed by periodically mining new contents of the log files and can be summarized in the following steps: pre-process logs to extract user sessions, summarize the session in terms of user profiles, and create context associations from user profiles. In such environment, fuzzy reasoning is also a good framework for the recommendation process. The approach proposed in [16] is fit for real-time recommendation. Similar approach is presented in [15] with a two-step recommendation system, which uses specific URL-predictor neural networks.

Recommendation techniques still address several problems and do not represent easy road to success [18]. They require more sophisticated methods of acquiring user needs. It is becoming harder to design a system suitable for all users and contexts. Open systems suffer from a drawback in which users surf via a proxy, and their identities are anonymous. Finally, it is often tricky to prove empirically ability of adaptation [8]. It is a very reasonable question to ask whether or not the recommendation will actually improve the system.

In this work we focus on Web usage mining by user profiling, content and log analysis to recommend queries profiles. It is organized as follows. In Section 2 we outline the proposed method of recommendation. Section 3 discusses the functionalities of the real estate cadastre system and the requirements of its design. Then we focus on the pages structure in the ISEG2000-INT system in Section 4. Section 5 and 6 introduce some theoretical and implementation details while Section 7 states the conclusions.

2 Outline of the recommendation method

2.1 Internet system and user model

The method is designed for an internet information system which provides for its users different retrieval mechanisms based on definite number of search criteria. The system is built in form of an option tree with search forms located at different levels of that tree (see Figure 1). The users of the system are the members of an organization e.g. a local government or a corporation and fulfil their everyday duties frequently use information obtained from the system. For some days they focus on specific topics and after completing one task they move to another one and change their topics of interest. The crucial assumption for the recommendation method presented in the article is that the users for some time perform searches around similar or the same topics. The main goal of the method is to prompt the users the page profiles calculated on the basis of usage frequency of queries and time when the searches were conducted. When the list of recommended page profiles is available in the main menu page, it can facilitate and accelerate user searches by moving him directly to the chosen option page with search form filled with the most expected criteria values.

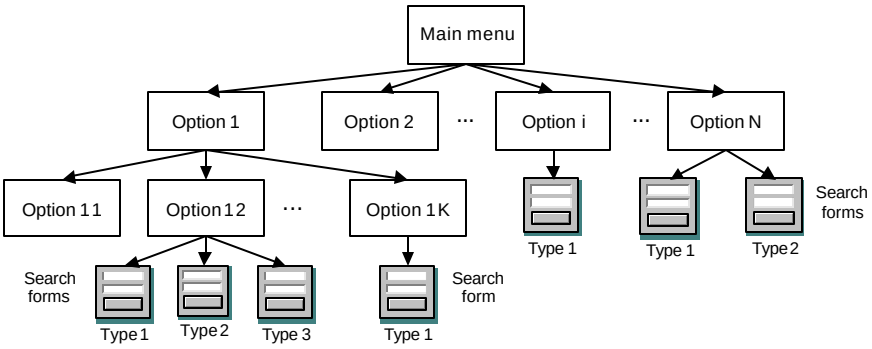

Figure 1 General model of an internet information system

2.2 Description of the method

The idea of page recommendation is based on the concept of a page profile, which concerns the pages with search forms. The page profile is characterized by the option of the system containing a retrieval mechanism, search criteria and finally by the type of that mechanism (e.g. simplified or extended). A formal model of page profiles can be presented as follows.

For representing page profiles we use a finite set O of system options and a finite set C of search criteria, where $C = \{C_1, C_2, \ldots, C_N\}$ and a finite set T of retrieval mechanism types. The page profile is defined as a tuple $P = \langle O, C_1, C_2, .., C_N, T \rangle$,
where

- $O = \{o_1, o_2, \ldots, o_{No}\}$ – a set of system options, where N_o is the number of options,
- $C_i = \{c_{i1}, c_{i2}, \ldots, c_{iNci}\}$ – a set of values of the i-th search criterion, where N_{ci} is the number of values of that criterion,
- $T = \{t_1, t_2, \ldots, t_{Nt}\}$ – a set of types of retrieval mechanisms, where N_t is the number of types.

So the number of page profiles is equal to $N_p = N_o * N_{c1} * N_{c2} * \ldots * N_{cN} * N_t$.

All criteria occurring in all search forms are taken into account, because some of them may be found in several or even in all forms.

The mechanism analyses previous queries input by a given user into the system and takes into account the frequency and time of each query element usage, it calculates the rank value of each page profile. The rank value of page profiles is automatically calculated for each user separately. In order to lower the contribution of query elements used earlier, the factor $1-d/D_u$ called time impact coefficient has been introduced (see Figure 2), where

- u is an identifier of the user,
- d is the number of expired days since the moment of calculation,
- D_u is the number of days taken into account for the user u. The value of D_u should be determined by each user as the component of his set of preferences.

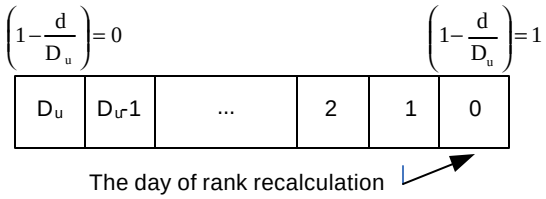

Figure 2 Time impact coefficient definition

Rank value $r_u(p_j)$ of a page profile p_j used by user u can be calculated in the following way

$$r_u(p_j)=\sum_{d=0}^{Du-1} f_{ud}(p_j)*(1-\frac{d}{D_u}) \qquad \textbf{(1)}$$

where $f_{ud}(p_j)$ is the usage frequency of a page profile p_j by user u and d days back before the day of rank calculation. Rank values of page profiles can be modified by rank values of options, selected set of search criteria and types of retrieval mechanisms computed in a very similar way but for all uses, irrespectively of any page profile. Modifying factors seem to be useful when rank values of page profiles are equal or small, then the most frequently used options or search criteria can play greater role.

Rank value $r_u(o_j)$ of an option o_j used by user u within the period of D_u days can be determined using following formula

$$r_u(o_j)=\sum_{d=0}^{Du-1} f_{ud}(o_j)*(1-\frac{d}{D_u}) \qquad \textbf{(2)}$$

where $f_{ud}(o_j)$ is the usage frequency of option o_j by user u and d days back before the day of rank calculation.

Rank value $r_u(c_{ij})$ of the j-th value of i-th criterion used by user u can be considered as follows

$$r_u(c_{ij})=\sum_{d=0}^{Du-1} f_{ud}(c_{ij})*(1-\frac{d}{D_u}) \qquad \textbf{(3)}$$

where $f_{ud}(c_{ij})$ is the usage frequency of of the j-th value of i-th criterion by user u and d days back before the day of rank calculation.

Rank value $r_u(t_j)$ of retrieval mechanism type t_j used by user u can be obtained in the following way

$$r_u(t_j)=\sum_{d=0}^{Du-1} f_{ud}(t_j)*(1-\frac{d}{D_u}) \qquad \textbf{(4)}$$

where $f_{ud}(t_j)$ is the usage frequency of the type t_j by user u and d days back before the day of rank calculation.

Total rank value of a page profile assumes that primarily calculated rank value is then modified by rank values of other elements according to user preferences. So it can be calculated as follows

$$R_u(p_j) = r_u(p_j) + w_{uo} * r_u(o_j) + w_{ut} * r_u(t_j) + w_{uc1} * r_u(c_{1j}) + \ldots + w_{uck} * r_u(c_{kj}) \quad \textbf{(5)}$$

where

- w_{uo} – significance weight for options determined by u-th user,
- w_{ut} – significance weight for retrieval mechanism types determined by u-th user,
- w_{uc1}, w_{uc2}, ..,w_{uck} – significance weights for C_1, C_2, C_k criteria respectively determined by u-th user and k is the number of criteria chosen by the user to be used as rank value modifiers.

Each user can set his preferences by determining the number of days D_u for calculation of rank values and the weights of modifying rank values of other elements. So if the weight assigned to a given element equals zero, it means that the user does not want to use that element to modify his page profiles.

2.3 Idea of the implementation

When a user runs a query at any option page with a query form, the features of that query i.e. option, type of retrieval mechanisms and the values of criteria used are immediately saved to the log of submitted queries. Total rank values are calculated for each page profile, for each user separately, every day using data saved in the log of submitted queries and preferences set by individual users. This can be performed in delayed mode, e.g. at night, when almost nobody uses the system. The result is saved in the log of recommended pages (see Figure 3).

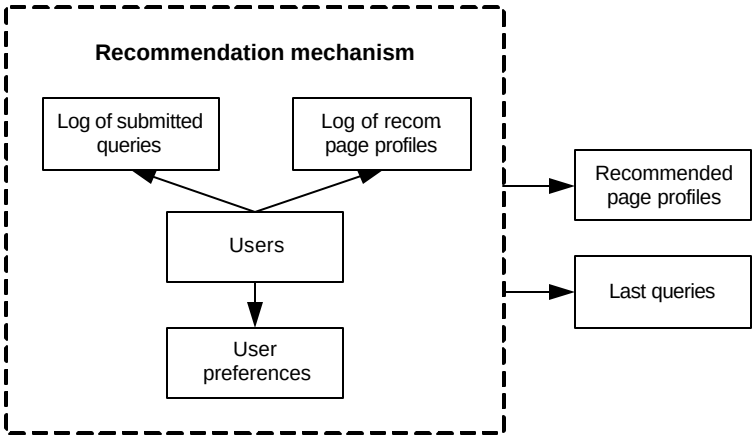

Figure 3 Schema of the implementation of the method

Resulting lists of page profiles with the highest scores are presented to each user on the main menu page and on all option pages with search forms. Having chosen an item from the list the user control is moved to the option page with search form determined by the profile. The criteria fields are filled with values according to the profile selected. Due to this mechanism the user is relieved of an arduous navigation through the option tree and from laborious filling of criteria fields in the case when he intends to carry out search which is the same or similar to searches previously performed. Then the user can immediately run the query prompted by the system or modify that query. Additional mechanism is list of last queries submitted by individual users available in the same pages as the list of recommended profiles. This could be very useful especially when a

user performed many similar searches during one day and they were not taken into account yet by the recommendation mechanism.

3 The real estate cadastre system

The real estate cadastre system is designed to maintain the register of all parcels, buildings and apartments as well as their owners and users at a given territory. The maintenance of real estate cadastre registers in Poland is dispersed. There are above 400 information centres located by administrative district governments as well as by the municipalities of bigger towns which exploit different cadastre systems. The ISEG2000-INT system presented in the paper is an internet information system designed for the retrieval of real estate cadastre data and is complementary to the main system in which cadastre database is updated. The system has been deployed in about 50 intranets and extranets in local governments throughout Poland while the main EGB2000 cadastre system is used by above 100 centres.

The ISEG2000-INT system has been implemented using PHP script language and accommodated for cooperation with Apache or IIS Web servers. It assures communication with MS SQL Server and MySQL database management systems. Using the system mainly rests on formulating queries, browsing the list of retrieved objects, choosing the objects to reports and generating reports in PDF format. As shown in Figure 4 two main search criteria are municipality and the section which reflects the main spatial division of data in the cadastre system.

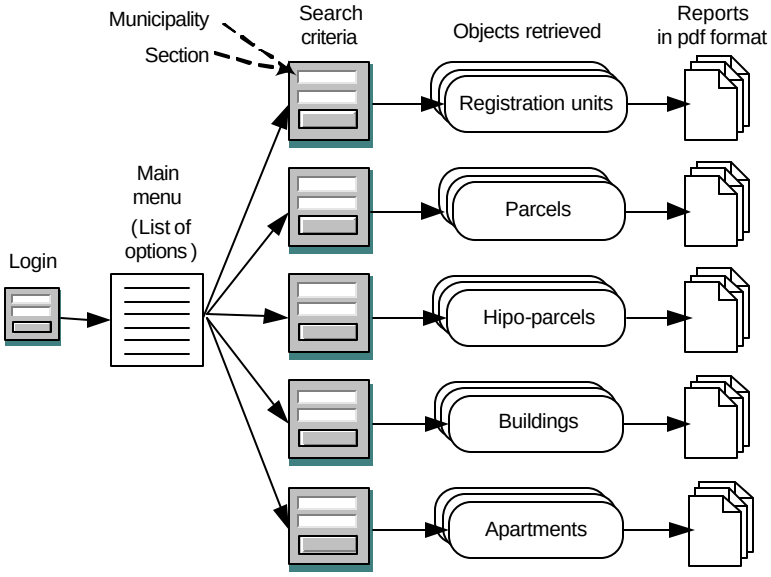

Figure 4 Generalized schema of the ISEG2000-INT cadastre information system

At present the data in cadastre systems have not been completed yet. However, descriptive data of land premises are fully completed. The information centres have been still gathering the data of buildings and apartments. Numeric plans of real estate are being created or complemented too. Numbers of objects contained in databases of four selected information centres are given in Table 1.

Centre no	1	2	3	4
Registration units	27 925	6 593	7 432	7 011
Parcels	38 685	11 200	9 987	9 687
Hypo-parcels	44 727	16 409	12 945	14 753
Buildings	146	8	0	18
Apartments	69	0	0	102
Registered users	28	21	16	9
Active users	25	11	7	6

Table 1 Number of objects in cadastre database in selected centres

The access to the system is limited. Each user should be registered in the system and the rights should be assigned to the data from a given territory. The users of the system are the workers of local governments who utilize data to prepare administrative decisions, to inform real estate owners and to prepare reports for management boards of local governments. Usage statistics of the most frequently requested pages (1, 11, 13, 15, 41) in four information centres during months from November 2004 to February 2005 are presented in Figure 5. It is expected that the usage of the system will increase significantly according to the greater number of buildings and apartments registered.

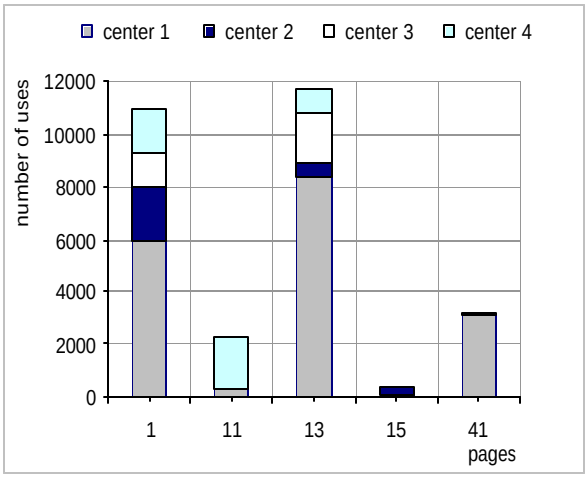

Figure 5 Usage statistics of selected pages in selected information centres

4 The structure of pages in the system

The structure of pages in the ISEG2000-INT system is presented in Figure 6, where R denotes pages with retrieved objects. In turn Table 2 contains the list of the options of the main menu with option codes, which are used to represent recommended pages.

Option no	Name of the option	Option code
1	Land registration unit search	LRU
2	Parcel search	PAR
3	Parcel search from a list	PLI
4	Building registration unit search	BRU
5	Building search	BUI
6	Apartment registration unit search	ARU
7	Apartment search	APA

8	Hypo-parcel search	HYP
9	Price and value of premises search	PRI
10	System usage monitoring	MON
11	Statistics	STA

Table 2 Options of main menu

Building a recommendation mechanism in this case is a challenging problem for several reasons. We need to build a flexible query subsystem with the width range of topics which could interest the user. Since the user's past interests is of limited practical use the system should place less weight on history observations.

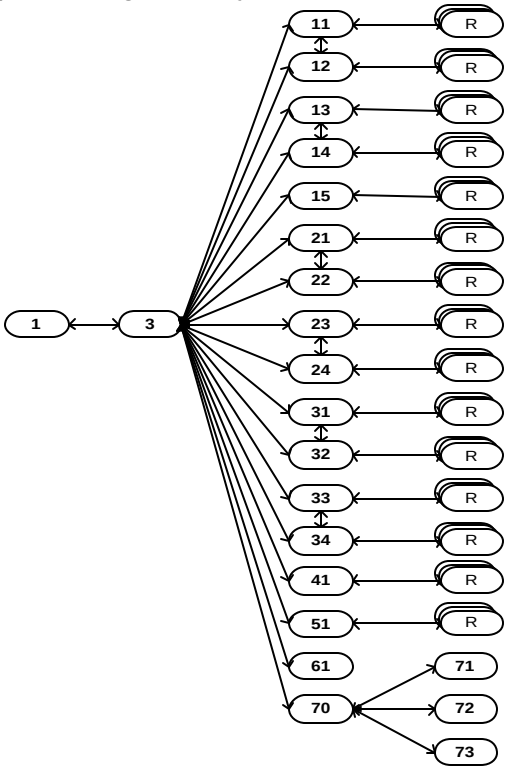

Figure 6 Structure of pages in the ISEG2000-INT system

Page No	Name of the option	Type of criteria
1	Login	
3	Main menu	
11	Land registration unit search	simplified
12	Land registration unit search	extended
13	Parcel search	simplified
14	Parcel search	extended
15	Parcel search based on a list	
21	Building registration unit search	simplified
22	Building registration unit search	extended
23	Building search	simplified
24	Building search	extended

31	Apartment registration unit search	simplified
32	Apartment registration unit search	extended
33	Apartment search	simplified
34	Apartment search	extended
41	Hypo-parcel search	
51	Price and value of premises search	
61	System usage monitoring	
70	Statistics – menu	
71	Statistics – list of parcels on a map	
72	Statistics – area of units in sections	
73	Statistics – area of sections	

Table 3 Pages with search criteria

5 The recommendation method used in the cadastre system

The idea of recommendation used in the cadastre system is based on the concept of a page profile introduced in Section 2. Only two search criteria i.e. a municipality and a section were selected. Both are presented on almost every page enumerated in Table 3. The page profile of the cadastre system comprises therefore the option of the main menu, the type of search criteria (simplified or extended) and finally a municipality and a section chosen during retrieval process. A formal model of page profiles can be presented as follows.

The page profile is defined as a quadruple <O, M, S, T>,
where

- $O = \{o_1, o_2, \ldots, o_{No}\}$ – is a set of system options, where N_o is the number of options,
- $M = \{m_1, m_2, \ldots, m_{Nm}\}$ – is a set of municipalities registered in the system, where N_m is the number of municipalities,
- $S = \{s_1, s_2, \ldots, s_{Ns}\}$ – is a set of sections registered in the system, where N_s is the number of sections,
- $T = \{t_1, t_2, \ldots, t_{Nt}\}$ – a set of types of retrieval mechanisms, where N_t is the number of types.

The time impact coefficient i.e. the factor $1 - d/D_u$ has been also taken into account to reflect the following principle: the earlier used the element the lower its rank value.

5.1 Rank value of a page profile

Rank value $r_u(p_j)$ of a page profile p_j used by user u is calculated in the following way:

$$r_u(p_j) = \sum_{d=0}^{Du-1} f_{ud}(p_j) * (1 - \frac{d}{D_u}) \qquad \textbf{(6)}$$

where $f_{ud}(p_j)$ is the usage frequency of page profile p_j by user u and d days back before the day of rank calculation.

In order to reveal the nature of this element the query history log has been analysed for one of the most active user in a chosen information centre for the period of two months from November 2005 to December 2005. This user formulated 9810 queries with simplified criteria, used 6 options and 254 sections during this period. Figure 7 shows how the rank values of three pages change in function of time.

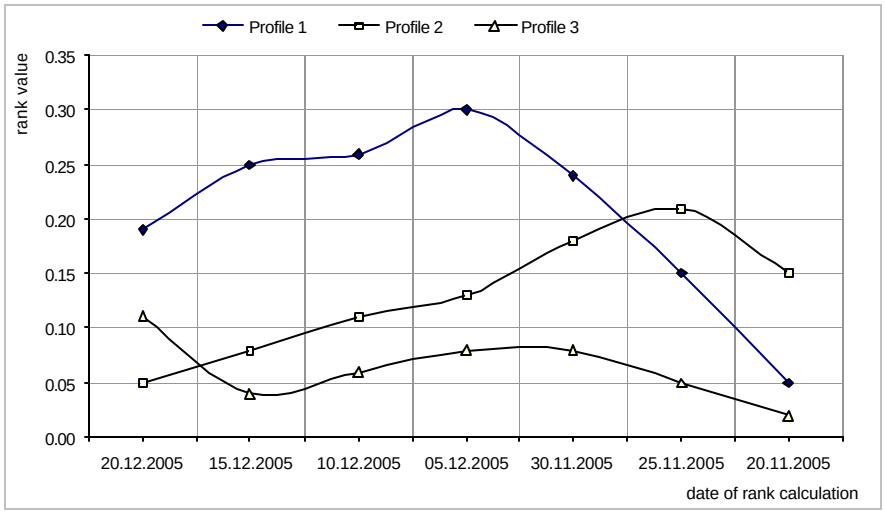

Figure 7 Rank values of page profiles for a selected user

5.2 Rank value of an option

Rank value $r_u(o_j)$ of an option o_j used by user u is shown as follows

$$r_u(o_j) = \sum_{d=0}^{Du-1} f_{ud}(o_j) * (1 - \frac{d}{D_u}) \qquad \textbf{(7)}$$

where $f_{ud}(o_j)$ is the usage frequency of option o_j by user u and d days back before the day of rank calculation.

Similarly, in order to reveal the nature of this element analogous analysis for the same user and the same period has been carried out. Figure 8 shows how rank values of three options change in function of time. It should be noted that the graph in this figure was plotted using data taken from different information centre and for different user than graphs presented in other figures.

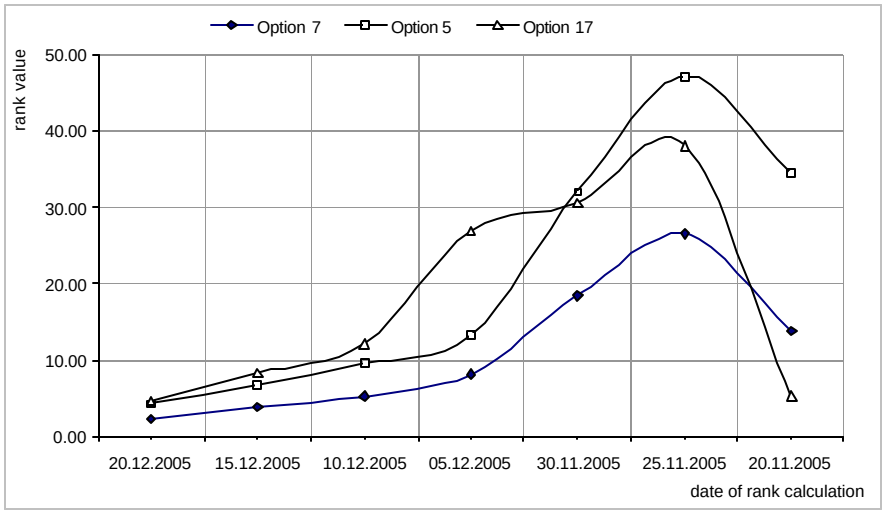

Figure 8 Rank values of options for a selected user

5.3 Rank value of a municipality

Rank value $r_u(m_j)$ of a municipality m_j used by user u is presented in the following way

$$r_u(m_j) = \sum_{d=0}^{Du-1} f_{ud}(m_j) * (1 - \frac{d}{D_u}) \quad \textbf{(8)}$$

where $f_{ud}(m_j)$ is the usage frequency of municipality m_j by user u and d days back before the day of rank calculation. In the cadastre system tested the rank value of a municipality was not calculated, because the system covers only one registration unit and a municipality is always the same.

5.4 Rank value of a section

Rank value $r_u(s_j)$ of a section s_j used by user u is obtained in the following way

$$r_u(s_j) = \sum_{d=0}^{Du-1} f_{ud}(s_j) * (1 - \frac{d}{D_u}) \quad \textbf{(9)}$$

where $f_{ud}(s_j)$ is the usage frequency of section s_j by user u and d days back before the day of rank calculation.

In order to reveal the nature of this element analogous analysis for the same user and the same period has been carried out. Figure 9 shows how the rank values of three sections change in function of time.

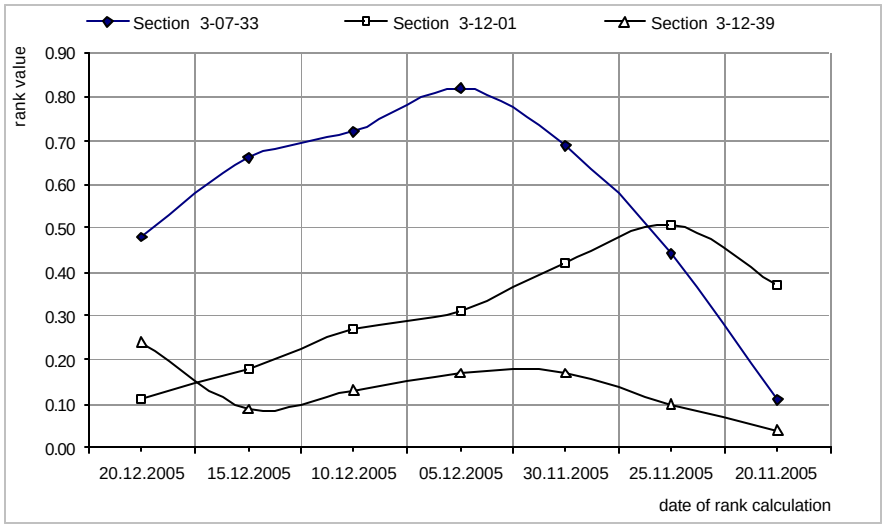

Figure 9 Rank values of sections for a selected user

5.5 Rank value of a criterion type

Rank value $r_u(t_j)$ of a criterion t_j used by user u can be calculated in the following way

$$r_u(t_j) = \sum_{d=0}^{Du-1} f_{ud}(t_j) * (1 - \frac{d}{D_u}) \quad \textbf{(10)}$$

where $f_{ud}(t_j)$ is the usage frequency of criterion type t_j by user u and d days back before the day of rank calculation.

5.6 Total rank value of a page profile

Total rank value of a page profile assumes that the rank value calculated primarily is modified by rank values of other elements according to user preferences. So it can be calculated as follows

$$R_u(p_j) = r_u(p_j) + w_{uo} * r_u(o_j) + w_{um} * r_u(m_j) + w_{us} * r_u(s_j) + w_{ut} * r_u(t_j) \quad \textbf{(11)}$$

where

- W_{uo} – significance weight of an option determined by u-th user, (0.05 used),
- W_{um} – significance weight of a municipality determined by u-th user, (0 used),
- W_{us} – significance weight of a section determined by u-th user, (0.25 used),
- W_{ut} – significance weight of a criterion type determined by u-th user (0.4 used).

Similarly, in order to reveal the nature of this element Figure 10 shows how the total rank values of three profiles change in function of time.

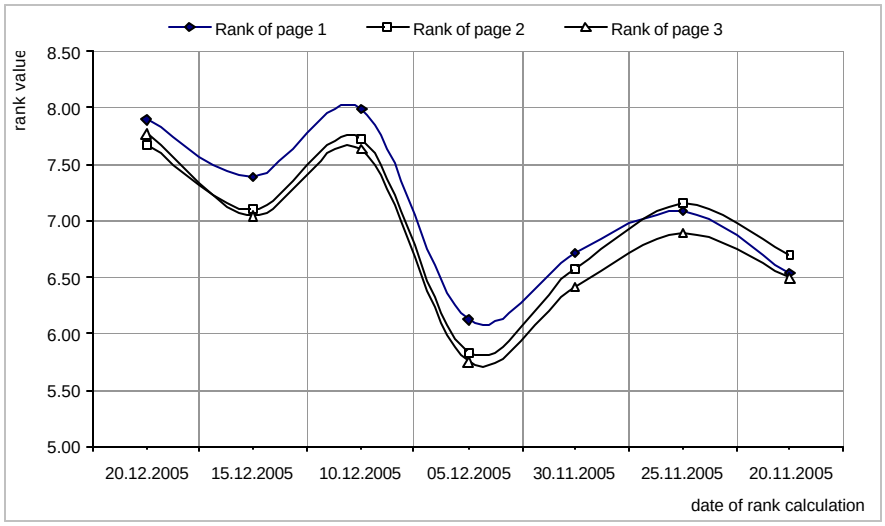

Figure 10 Total rank values of page profiles for a selected user

6 The implementation of the method

Two mechanisms have been designed and implemented. The first one is the list of last queries submitted by a given user and the second one is the list of page profiles recommended to a user. In order to implement the recommendation mechanisms logs of submitted queries have been created. Users have the possibility to determine their preferences by assigning significance weights to the elements of page profiles, by pointing the number of days when the usage frequency is taken into account and by stating the number of items in the lists of recommended pages and last queries.

The structure of main objects implemented is shown in Figure 11. The query history comprises all queries submitted by users to the system including information of the option and the type of search mechanism used. On the basis of the query history the list of last submitted queries can be created as well as the rank values of page profiles can be calculated.

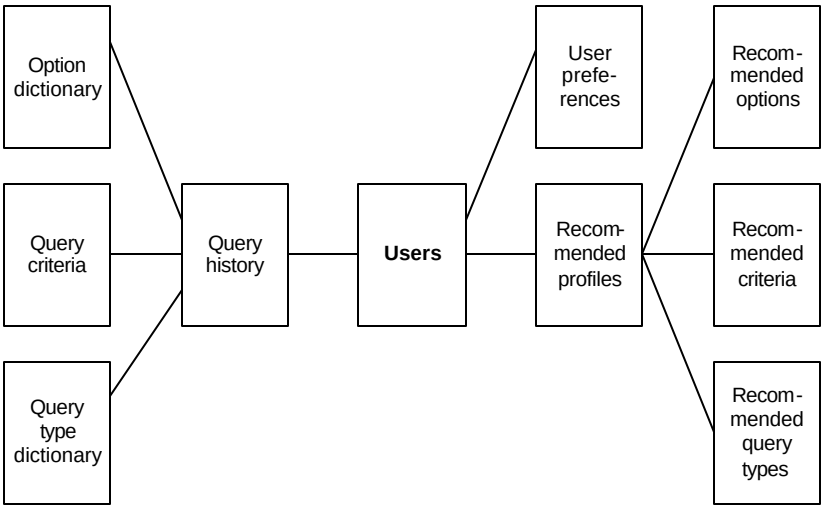

Figure 11 Structure of the objects in the recommendation method

6.1 The list of last queries submitted

The list of last queries submitted by a user is updated online. The list is provided for the user in a combo box available on each page with search criteria as well as on the page with main menu. Queries on the list are described by data of submission, option code, type of criterion, municipality code, section code and the number of object retrieved. An example of the list of last queries ordered descending by data is shown in Table 4. The item at the list consists of following elements:

- number of a query submitted by a given user, e.g. 2419
- data of query run, e.g. 2005-12-20
- option code as enumerated in Table 2, e.g. PAR
- type of criteria: S for simplified and E for extended
- name of a section, e.g. 307-34; in other systems names are not codes but short names of settlements or villages.

The name of a municipality has been omitted, because the system tested covers only one registration unit and the name of municipality is always the same. When a given item element was not used in a query the mask of xxxxx is inserted instead of a code of a section in order to point out this fact.

Queries submitted on 2005-12-20	Queries submitted on 2005-11-20
2418 : 2005-12-20 : PAR : S : xxxxxxxx	1165 : 2005-11-20 : PAR : S : 3-13-06
2418 : 2005-12-20 : PAR : S : 3-07-34	1164 : 2005-11-20 : PAR : S : 3-15-03
2417 : 2005-12-20 : PAR : S : xxxxxxxx	1163 : 2005-11-20 : PAR : S : 3-11-21
2416 : 2005-12-20 : PAR : S : xxxxxxxx	1162 : 2005-11-20 : PAR : S : 3-15-15
2415 : 2005-12-20 : PAR : S : 3-07-34	1161 : 2005-11-20 : PAR : S : 3-13-23

Table 4 The list of last queries submitted by a given user

When a user chooses a query from the list, the system moves the control to a page with a search form pointed by the option and the type of criteria and fills all the fields with the values put previously by the user.

6.2 The list of page profiles recommended

The recommendation mechanism analyses previous activity of a user and calculates and assigns a rank value to each page profile used by him. Similarly, the list of recommended page profiles is provided for the user in a combo box available on each page with search criteria as well as on the page with main menu. Page profiles on the list are characterized by an option code, type of criterion, municipality code and section code. An example of the list of recommended page profiles ordered descending by rank value is shown in Table 5. The item of the list consists of following elements:

- option code as enumerated in Table 2, e.g. PAR
- type of criteria: S for simplified and E for extended
- code of a section, e.g. 2419
- name of a section, e.g. 307-34; in other systems names are not codes but short names of settlements or villages.

Analogously to the list of last submitted queries the name of a municipality has been omitted and also when given item does not occur in a profile, the mask of xxxxx is inserted instead of a code or a name of a section.

Profiles recommended on 2005-12-20	Profiles recommended on 2005-11-20
PAR : S : xxxx : xxxxxxx	PAR : S : xxxx : xxxxxxx
PAR : S : 0733 : 3-07-33	PAR : S : 1231 : 3-12-31
PAR : S : 0732 : 3-07-32	PAR : S : 1330 : 3-13-30
PAR : S : 1115 : 3-11-15	PAR : S : 1201 : 3-12-01

Table 5 The list of page profiles with search criteria recommended to a user

When a user chooses a page profile from the list, the system moves the control to a page with a search form pointed by the option and the type of criteria and fills all the fields with the values contained in the profile.

7 Conclusions

The recommendation method has been designed for an internet information system which provides for its users different retrieval mechanisms based on definite number of search criteria. It has been assumed that the users for some time perform searches concerning similar or the same topics and then accomplishing next task they focus for some days on other topic. Each query submitted to the system is saved in a special log which is used to calculate rank values of page profiles separately for each user. The page profiles, which represent a system option, type of retrieval mechanisms and search criteria, are given values according to their frequency of usage and the time impact coefficient. A recommended page profile selected by a user from a list facilitates and accelerates his searches by moving him directly to the chosen option page with search form filled with the most expected criteria values. As an additional complementary mechanism the list of last submitted queries is available to each user.

The recommendation mechanism has been implemented and provided to the users of the ISEG2000-INT cadastre information system. The list of last queries is updated in online mode whereas the list of recommended pages is created for individual user every night.

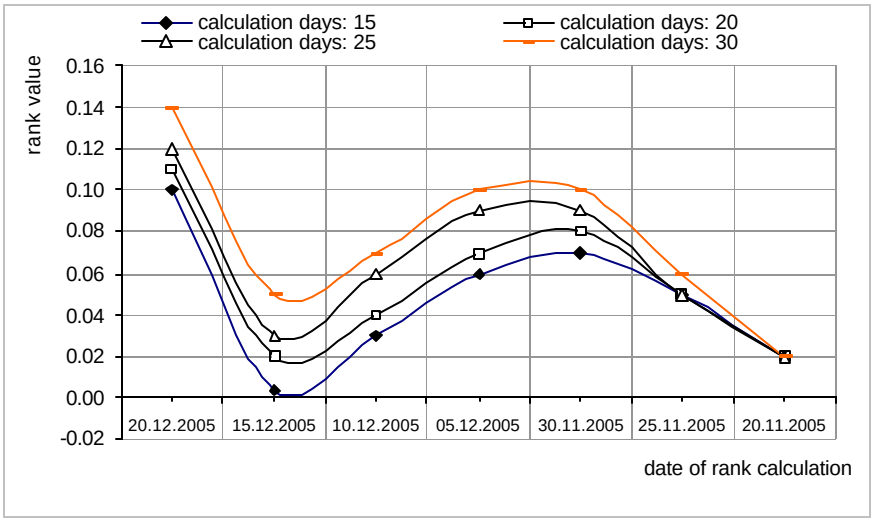

Figure 12 Total rank values of page profiles for a selected time period

The performance of the proposed recommendation method may be evaluated in comparison with the last queries method. Experimental results have shown that the recommendation algorithm with the time impact coefficient improves the efficiency of the usage, especially when the information needed becomes more diverse and the number of accessed pages gets larger. In addition, we have measured rank value with respect to different time periods. The results are given in Figure 12. When the period of calculation becomes to short, the rank value decreases. Therefore, we used 30 days period for all evaluations.

Future work will focus on observing how users will use the recommendation mechanism and what values will gain the ranks depending on different values of parameters. For example, we want to test whether a system with recommendation works better for experienced user or new users. It will be interesting to investigate the behaviour of system users using association rules and fuzzy logic methods.

Acknowledgements

We thank Dr. Ngoc Thanh Nguyen who gave us a lot of valuable advices. We also thank the Computer Association for Information BOGART Ltd for financial support and participation in the experiments. We thank anonymous reviewers for their helpful suggestions.

References

[1] L. Ardissono, L. Console, I. Torre: *An adaptive system for the personalized access to news*. AI Communications, vol. 14(3), 2001, pp. 129-147

[2] L. Ardissono, A. Goy, G. Petrone, M. Segnan, P. Torasso: *Intrigue: Personalized Recommendation of Tourist Attractions for Desktop and Handset Devices*. Applied Artificial Intelligence, vol. 17(8-9), 2003, pp. 687-714

[3] R. Baeza-Yates, C. Hurtado, M. Mendoza: *Query Recommendation Using Query Logs in Search Engines*. Lecture Notes in Computer Science vol. 3268, 2004, pp.588–597

[4]. D. Billsus, M. Pazzani: *User Modelling for Adaptive News Access*. User Modelling and User-Adapted Interaction, vol. 10, 2000, pp.147-180

[5] P. Brusilowsky: *Adaptive Hypermedia*. User Modelling and User-Adapted Interaction, vol. 11, 2001, pp. 87-110

[6] Z. Chedrawy, S. S. R. Abidi: *Intelligent Knowledge Sharing Strategy Featuring Item-Based Collaborative Filtering and Case Based Reasoning*. In Proc. International Conference on Intelligent Systems Design and Applications, 2005, pp.67-72

[7] D. N. Chin: *Empirical Evaluation of User Models and User-Adapted Systems*. User Modeling and User-Adapted Interaction, vol. 11, 2001, pp.181-194

[8] M. Eirinaki, M. Vazirgiannis: *Web Mining for Web Personalization*. ACM Transactions on Internet Technology vol. 3, no. 1, 2003, pp. 1–27

[9] S. Gunduz, T. Ozsu: *A User Behaviour Model for Web Page Navigation*. University of Waterloo, 2002

[10] A. Jameson: *Adaptive Interfaces and Agents*. In J. Jacko, A. Sears (Eds.), Human-Computer Interaction Handbook, Mahwah, NJ: Erlbaum, chapter 15, 2003, pp. 305-330

[11] P. Kazienko: *Multi-agent Web Recommendation Method Based on Indirect Association Rules*. Lecture Notes in Artificial Intelligence, vol. 3214, 2004, pp. 1157–1164

[12] P. Kazienko, M. Kiewra: *Personalized Recommendation of Web pages*. International Series on Advanced Intelligence, vol. 10, 2004, pp. 163–183

[13] H. T. Macedo, J. Robin: *Increasing Profitability: Voice-Based Browsing to Recommendation System Web Services*. Lecture Notes in Artificial Intelligence vol. 3528, 2005, pp.279–285

[14] S-H. Min, I. Han: *Optimizing Collaborative Filtering Recommender Systems*. Lecture Notes in Artificial Intelligence vol. 3528, 2005, pp.313–319

[15] O. Nasraoui, M. Pavuluri: *Complete this Puzzle: A Connectionist Approach to Accurate Web Recommendations based on a Committee of Predictors*. In Proc. WebKDD Workshop on Web Mining and Web Usage Analysis, Seattle, WA, 2004

[16] O. Nasraoui, C. Petenes: *Combining Web Usage Mining and Fuzzy Inference for Website Personalization*. In Proc. KDD Workshop on Web mining as a Premise to Effective and Intelligent Web Applications, Washington DC, 2003

[17] J. Sobecki: *Web-Based Systems User Interfaces Recommendation Using Hybrid Methods*. International Series on Advanced Intelligence, vol. 10, 2004, pp. 464–469

[18] M. Vozalis, K. G. Margaritis: *Applying SVD on Item-based Filtering*. In Proc. International Conference on Intelligent Systems Design and Applications, 2005, pp.67-72

[19] B. Zhou, S. Hui, A. Fong: *Efficient sequential access pattern mining for web recommendation*. International Journal of Knowledge-based and intelligent Engineering Systems, vol. 10, no 2, 2006, pp. 155-68

[20] T. Zhu: *Web Usage Mining for Internet Recommendation*. University of Alberta, 2001