International Journal of Computer Science & Applications
Vol. 3 Issue 3, pp 12-26

Personalized Integration of Recommendation Methods for E-commerce

Przemyslaw Kazienko and Pawel Kolodziejski

Wrocla w University of Technology, Institute of Applied Informatics
Wybrzeze S. Wyspianskiego 27, 50-370 Wroclaw, Poland
kazienko@pwr.wroc.pl, super@hot.pl

Abstract

The hybrid system of personalized product recommendation in e-commerce, by integrating various methods, was presented in the paper. Each e-commerce user has assigned their own weights corresponding to particular methods. Due to the permanent and personalized adaptation of these weights, the system can adjust the influence of individual methods separat ely for each user. Testing the implementation and evaluation of recommendation efficiency were also described.

1 Introduction

Recommender systems are an important part of recent e-commerce. They enable the increase of sales by suggesting to users selected products on offer. The problem of how to choose the most suitable items, possibly with respect to the user's inclinations, is a challenging research problem that has been investigated for many years.

2 Recommendation Methods

Four fundamental approaches to recommendation can be mentioned: demographic filtering, collaborative and content-based recommendation, and simplified statistical approaches [15]. In demographic recommendation, users are classified based on their personal data, which they themselves provided during the registration pro cess [19]. Alternatively, this data can be extracted from the purchasing history, survey responses, etc. Each product is assigned to one or more classes with certain weights and the user is attracted to items from the class dosest to their profile. This is attribute based recommendation.

Collaborative recommendation is typically based on item ratings explicitly delivered by users. The system recommends products, which have been evaluated positively by another similar user or by a set of such users, whose ratings have the strongest correlation with the current user [10]. This is user-to-user correlation.

Content-based recommendation focuses on the similarity between products, usually taking into account their features like textual descriptions [14], hyperlinks, related ratings [24], or co-occurrence in the same purchased transactions or web user sessions [13]. Items that are the closest to the most recently processed (viewed), are recommended regardless of user preferences. This is item-to-item correlation. Association rules and sequential patterns are the most interesting techniques used in recommendation based on item-to-item correlation [7, 8, 18, 20, 32]. They are usually applied to data sets related to items such as purchases [2, 18], ratings of TV programs [26], navigation paths [8, 18, 22, 32] rather than directly to item attributes.

In the statistical approach, the user is shown products based on some statistical factors; usually popularity measures like averages or summary ratings (the best rated), and numbers of sold units (the best buy) [28].

A single recommendation method can offer either ephemeral or persistent personalization. The former is based only on a current session and can deliver a different list on every page of a website but be the same for all users. Persistent personalization uses the history of user's behaviour and generates a different product list for each user, but it works only with identified, logged in users [28].

There are some surveys about recommender systems published [1, 25, 27, 28, 29].

3 Problem Description

Most recommendation methods have significant limitations (Table 1). Collaborative and some content-based filtering methods hardly cope with new users and new products, for which there is no appropriate data (ratings or purchases). Yet another analogue weakness is the problem of sparseness. It could be difficult to estimate reliable correlations between a product and a user in the environment with large amounts of empty data. This may also result in a recommendation list that is too short [13].

Method	Data source	**User** dependent adaptation (context)	**Viewed item** dependent adaptation (context)	Solution of the **new item** problem	Solution of the **new user** problem	Solution of the **sparseness** problem
Statistical approach (the best buy, the best rated)	Purchases / ratings / visits	–	–	– –	+	+
Collaborative filtering	Ratings	+	–	– –	–	–
Association rules, sequential patterns	Purchases / basket placements / navigation sessions	–	+	– –	+	–/+
Content based	Item attributes / text content	–	+	+	+	+
Demographic recommendation	Item attributes & user attributes	+	–	+	+	+

Table. 1. Features of basic recommendation methods. "+" denotes the solved problem while "–" the ones unsolved

Methods dependent on the user such as collaborative and demographic filtering cannot be used in anonymous e-commerce sites due to identification necessity. Additionally, they require much onerous effort from the customers who are forced to input their personal data.

Statistical and collaborative filtering as well as association rules method are not able to recommend new items to the user, this is their substantial weakness. This comes from the usually long-term lack of enough data about sales, ratings or web user sessions related to these new items. This is not the case for other, content-based methods that make use of item attributes, which are provided with the insertion of the new item.

Susceptibility of association rules to the problem of sparseness can be solved by the adjustment of the appropriate values of parameters such as minimum support and minimum confidence or by the density increase by means of the introduction of indirect association rules [13, 18].

There are two kinds of pages in regular e-commerce sites: product pages, relevant to items from the e-commerce offer and non-product (*normal)* pages that possess static content: the latest company news, product reviews, some practical advise, etc. The latter are usually not related to particular products. [13, 14]. Methods based only on item-to-item correlation, sensitive to "viewed item", hardly cope with recommendations on normal pages, because they are unable to select related products.

The remedy for these and other shortcomings, many hybrid systems were proposed [3, 4, 6, 10, 11, 15, 30, 31].

4 Personalized and Adaptive Integration of Recommendation Methods

4.1 The Concept

The main concept is to overcome the shortcomings of a single recommendation method and to deliver full personalization, which could offer every user different product lists which change during navigation. It simultaneously depends on watched products (content-based, ephemeral personalization), history of user's behaviour (e.g. ratings) and the user's likes and dislikes (persistent personalization) as well as effectiveness of previous recommendations for the given user (adapted personalization). To achieve full personalization, the system combines association rules for ephemeral content-based personalization as well as collaborative and demographic filtering for persistent one. Consequently, a complete hybrid recommender system is obtained that integrates many independent recommendation methods in a personalized and adaptive way. It exploits weights that are dynamically recalculated according to the effectiveness of the recommendation. It means that the more effective the particular method is, the bigger the weight it will have. This results in a bigger influence on the final recommendation list. Another unique feature of the concept is its personalization capability. Every user has its own personal set of weights corresponding to the method's usefulness for this individual. The system also uses its knowledge gained from previous users to better suit its new ones. At the first launch, with no user accounts, weights of all methods are set to system base values determined by constant initial parameters. After some users join the system, these system base weights are recalculated. First, the average value for each method is estimated from weight sets of all users. Next, these values are normalized so that their sum is the same as at the beginning. Every new user starts from system base weights as initial values. Once a user gets his own weights set, only his personal behavior has an influence on it. The problem of initialization and synchronization of knowledge update in recommender systems was considered in [12, 16]. The first preliminary version of our concept was published in [17].

The work starts with the user's interaction (Fig. 1). The context of interaction (the requested web page URL and the user identifier UID) determines which conditions have been fulfilled and, in consequence, which methods are allowed to present their recommendation lists. The context is also utilized by some methods e.g. collaborative filtering uses UID whereas association rules discovered from purchases require URL. The system is capable of integrating any number of methods, although in the implementation, only five have been used. If a user is logged in, the system exploits collaborative and demographic recommendations – only in this case the system has the appropriate source data. Otherwise, two simple statistical methods are used: "the best rated" and "the best buy". Collaborative filtering makes use of ratings inserted previously by registered users whereas demographic recommendation is based on matching personal data: likes and dislikes, pre-owned products, annual expenses on certain category, etc. To improve recommendation quality association rules were introduced. They reflect cases in which a given product was purchased together with the set of another one frequently enough that this set might be recommended on the web page describing the given product. This recommendation technique is a kind of content-based method, which generates a different but static list on every product page. Its biggest disadvantage is that it can be used only on product pages and not on other ones, e.g. news pages or so-called white pages [14]. The system assumption is that one product page corresponds to exactly one product from the e-commerce offer. Other kinds of relationships were studied in [14]. Note that all other considered recommendation methods are insensitive to the type of the requested page.

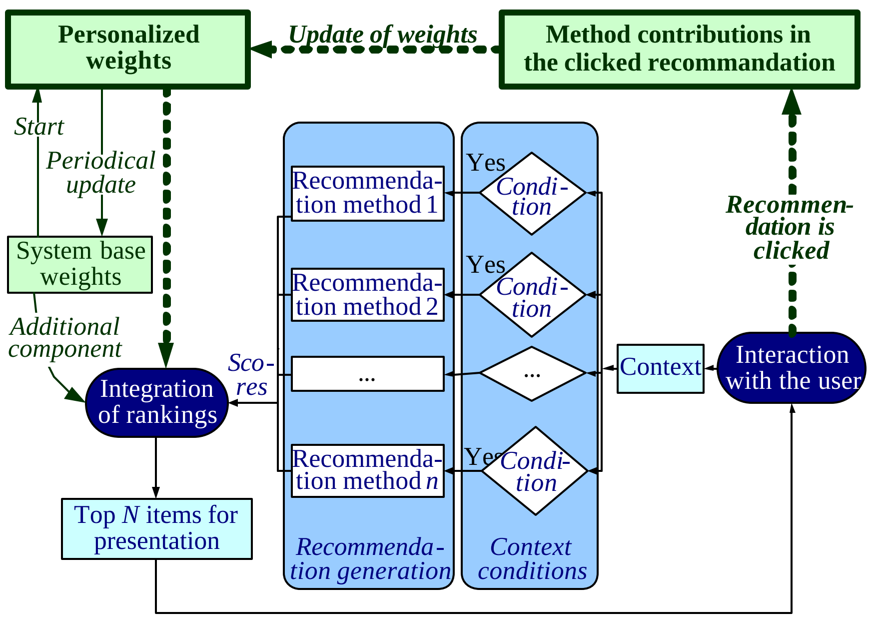

Figure 1. The concept of personalized integration of various recommendation methods. Dotted arrows correspond to the basic adaptation loop

Each method is independent from all others and it is provided only with the context data (URL, UID).

4.2 Personalized Score Integration

All methods relay, for further processing, their own list of recommended products with assigned appropriate scores for each. This method prerequisite is the positive value of every score. Having received these lists, the system integrates, normalizes and orders them using both the obtained scores and weights set that belongs to the given user:

$$f_{jkl} = \sum_{i=1}^{M} \frac{w_{ik} * s_{ijkl}}{max_{ikl}}, \quad s_{ijkl} = 0 \tag{1}$$

where: f_{jkl} – the final score of product j for user k in context (page) l; w_{ik} – the current weight of method i for user k; s_{ijkl} – the score of product j assigned by the recommendation method i for user k with respect to context l; M – the number of methods, max_{ikl} – maximum value of score s_{ijkl} among scores returned by the i-th method – the top one in the i-th ranking.

Factor $1/max_{ikl}$ is used to flatten different domains of methods to the range [0,1], i.e. the first item in the ranking of each method receives the value 1.

Note that all component recommendation methods are involved at recommendation process. In the opposite approach [9] only top ranked rules i.e. pairs *product1 -> product2* are used and a rule is always the result of a single method.

4.3 Personal Weight Adaptation

The top N candidates from the final recommendation list are presented to the user. Additionally, the system stores component scores for each of N items displayed to the user until the next user's request. This means that according to (1) several methods can have their contribution in a particular recommendation item. This contribution is the normalized score delivered by the method i.e. $\frac{s_{ijkl}}{max_{ikl}}$. If a user chooses one of recommendations linking to product j, the system checks what normalized score $\frac{s_{ijkl}}{max_{ikl}}$ had each i-th method in recommending this product and it adequately updates weights of all methods in the set of user k:

$$w'^{(1)}_{ik} = w^{(0)}_{i} + \frac{s_{ijkl}}{max_{ikl}}, \text{ after the first click on recommendation by the } k\text{-th user}$$

$$w'^{(n+1)}_{ik} = w^{(n)}_{ik} + \frac{s_{ijkl}}{max_{ikl}}, \text{ after the } (n+1)\text{-th click} \tag{2}$$

where: $w'^{(1)}_{ik}$, $w^{(n)}_{ik}$, $w'^{(n+1)}_{ik}$ – the weight of method i for user k after the first, n-th and n+1 user click on recommendation, respectively; $w^{(0)}_{i}$ – the initial system base weight for method i; s_{ijkl}– the score of the clicked, j-th product assigned by method i for user k; the recommendation is clicked in context l.

where: $w'^{(1)}_{ik}$, $w^{(n)}_{ik}$, $w'^{(n+1)}_{ik}$ – the weight of method i for user k after the first, n-th and n+1 user click on recommendation, respectively; $w^{(0)}_{i}$ – the initial system base weight for method i; s_{ijk} – the score of the clicked, j-th product assigned by method i for user k.

Additionally, the following normalization mechanism is used to preserve the constant sum of weights for each user:

$$w_{ik}^{(n+1)} = w'^{(n+1)}_{ik} * \frac{\sum_{i=1}^{M} w_{ik}^{(n)}}{\sum_{i=1}^{M} w'^{(n+1)}_{ik}} \qquad (3)$$

where: $w_{ik}^{(n+1)}$ is the normalized weight of method i for user k after the n+1 user click. At the next click, $w_{ik}^{(n+1)}$ is used at calculation of $w'^{(n+2)}_{ik}$ with (2). Note that:

$$\sum_{i=1}^{M} w_{ik}^{(n)} = \sum_{i=1}^{M} w_{i}^{(0)}$$

Generally, the greater initial value $w_i^{(0)}$ is used, the smaller influence onto weight changes has the single user selection of recommendation. The initial values of $w_i^{(0)}$ can help to adjust the recommender framework to the specific application domain. In an environment with the relatively big number of clicks on recommendation and general large activity of users greater values of $w_i^{(0)}$ can be more adequate. Similarly, if a typical user clicks on recommendation rather occasionally, the system should exploit this rare feedback more extensively by using smaller values of $w_i^{(0)}$.

After the user's interaction and weight update, the described cycle repeats with the next user http request.

4.4 System Base Weight Adaptation

Users have their personal weights adapted but we can also take advantage of them for the new users. To achieve it, the system base weights are periodically recalculated using the experience of all active users, as follows:

$$w_i^{(0)} = \frac{1}{K} \sum_{k=1}^{K} w_{ik} \qquad (4)$$

where K is the number of all active users. Note that only active users are taken into consideration. A user becomes active after their first click on any recommendation.

4.5 Balanced Personalized Score Integration

The periodically updated system base weights can be used also in the recommendation process as the additional component (Fig. 1) by the following modification of (1):

$$f'_{jkl} = \sum_{i=1}^{M} \frac{\left(a\ w_{ik} + (1-a)\, w_i^{(0)}\right) s_{ijkl}}{max_{ikl}}, \quad s_{ijkl} = 0, \qquad (5)$$

where α is the personalization factor from the range [0,1].

The greater a is the more personalized is the system. This approach is particularly useful for users, who rarely visit the site, because they can take advantage of the experiences of others even if they have not had any of their own.

4.6 Working Modes

The system can operate in three working modes: initial, regular or update mode (Fig. 2). The initial mode consists of an initial base weights assignment. It is invoked at the start of the system as well as at new user creation. In the regular working mode, the system exploits personal weights at recommendation usage (1) or alternatively (5) and it updates these weights at each selection of a recommendation by the user according to (2) and (3). Periodically, the system runs the third update mode in which the general knowledge is modified, especially, system base weights are updated (4). Additionally, the component recommendation methods have to revise their base knowledge once in a while due to some new ratings (collaborative filtering), new users, a new, deleted or modified product on offer, or a significant set of new purchases (association rules), etc. Some of these methods share the same data source e.g. association rules, the best buy and demographic method all make use of historical purchases. Since the update processes are usually system resource and time consuming, they should be carefully scheduled and not launched too often. The problem of synchronization and initialization of knowledge among many engaged units (agents) was considered in [12, 16].

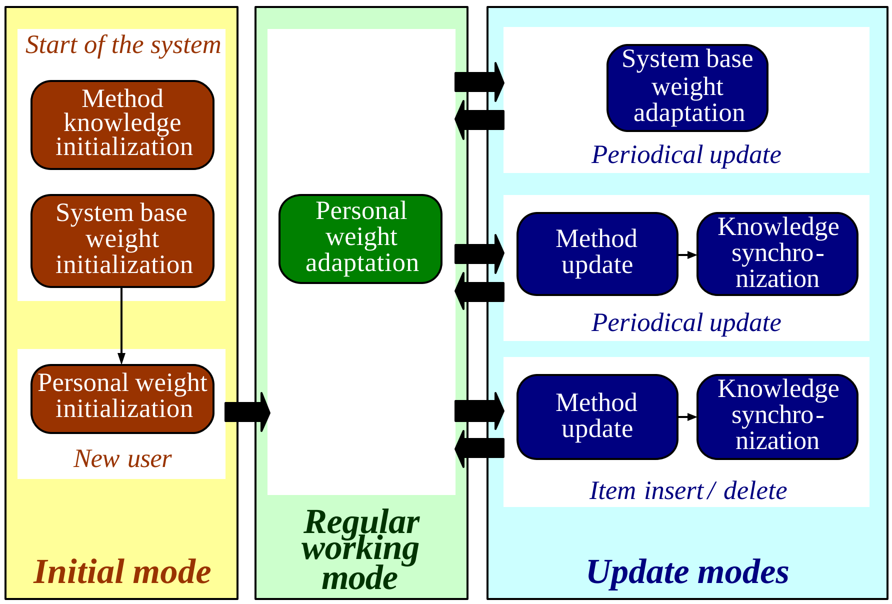

Figure 2. Working modes

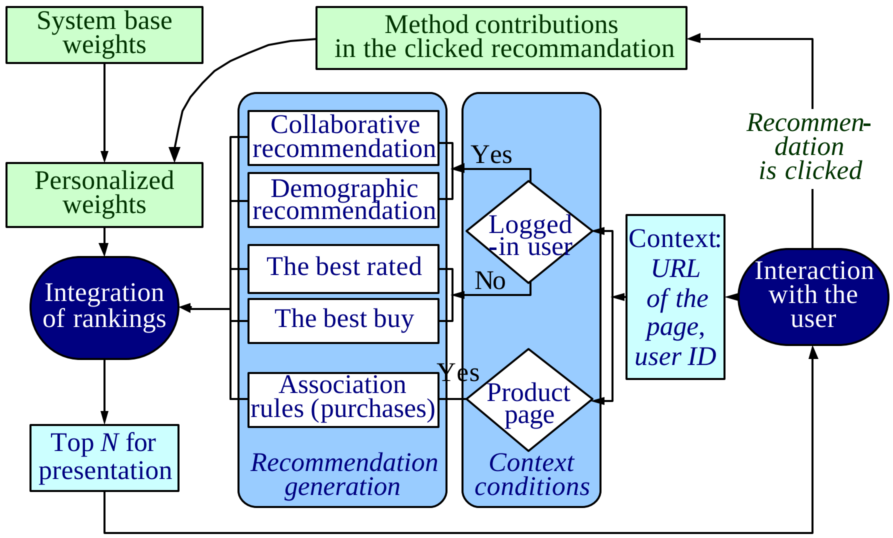

Figure 3. Personalized integration of various methods of recommendation in the WindOwls systems

5 WindOwls – Test Environment

The described above recommendation concept was implemented in the WindOwls system as an ecommerce windsurfing website (www.windowls.smellme.netlook.pl). It contains sections with news, articles, shop and settings. On the settings page users can modify the look of the interface (Fig. 4), fill in and change their personal information about their interests (i.e. profiles used for demographic filtering). The most important section of the website is the shop. A typical page in the shop contains a description of a single product with the possibility of buying or rating it. The average rating of the product provided by other users and three or less (if not available) recommendations are always visible. Every user is presented with an individual recommendation list that changes on each page during website navigation. The WindOwls system consists of five independent recommendation modules (Fig. 3):

1) **Collaborative recommendation** that delivers items best ranked by the users with other rankings similar to the current user. Each user is represented by one vector which coordinates correspond to ratings of particular products. First, the system clusters offline users i.e. vectors by their rankings to gain a number of neighbourhoods. For each cluster one centroid is calculated – the mean vector of all member vectors. The ranking list for the cluster is obtained directly from the centroid – this is the list of products with the highest value of its coordinates. Next, the current user is assigned to the closest cluster-neighbourhood based on the typical Euclidean distance.
2) **Demographic recommendation** consists in the recommendation of items most frequently bought within the group of users. For each user one vector of their features, like experience level, preferred wind strength, annual expenses on windsurfing, favourite water region, etc., is created. The system clusters users to groups, fixes the list of the most popular products for each group and finds the one which is the closest to the current user based on the current user's demographic features.

3) **The best rated** or the most popular is the static ranking of items that obtained the greatest sum of ratings (only positive ones) with respect t o their input time.
4) **The best buy** is also the static list of products that are most frequently bought. However, the old purchases have less influence on the position of rank due to the introduction of a special time fa ctor.
5) **Association rules** is the typical basket analysis method in which products purchased together frequently enough are more likely to be recommended. First, all simple association rules that exceed minimum support and minimum confidence are extracted from the historical shopping baskets. A simple rule means that only one product can occur on each side of the rule. Next, for the product from the left side of the rule, the system sorts decreasing its right sides by the confidence value. The obtained ordered list is the static recommendation ranking list.

Some recommendation methods like collaborative and demographic recommendation can operate only for identified (logged-in) users. Additionally, association rules method is based on the co-occurrence of products in the same transactions – item-to-item correlation, so it requires an input product. For that reason, it can work only for product pages, whereas all others methods are either user sensitive (user-to-user correlation) like collaborative and demographic filtering or context independent like the best rated and the best buy. These four methods can be used on all web pages in e-commerce portals, i.e. also on pages that are not related with any particular product i.e. on white pages. The last two methods are used only for non-logged users to enable easier comparison between different methods for identified users. Furthermore, the best rated method can be treated as the simplified and impersonal version of collaborative filtering.

Note that the context (Fig. 1, 3) also determines the product, in the case when the user requests a product page. Each recommendation method delivers only L items (products) to reduce necessary processing. L equal about $N*M$ appears to be quite sufficient to include most cases, where N is the maximum number of recommendations suggested to the user and remind them that M is the number of recommendation methods. N=3 and L=10 were assumed in the implementation. Due to the method preconditions, only three recommendation methods are able to supply suggestions at the same time. The greater L is: the less efficient is the system but the more accurate are the obtained scores.

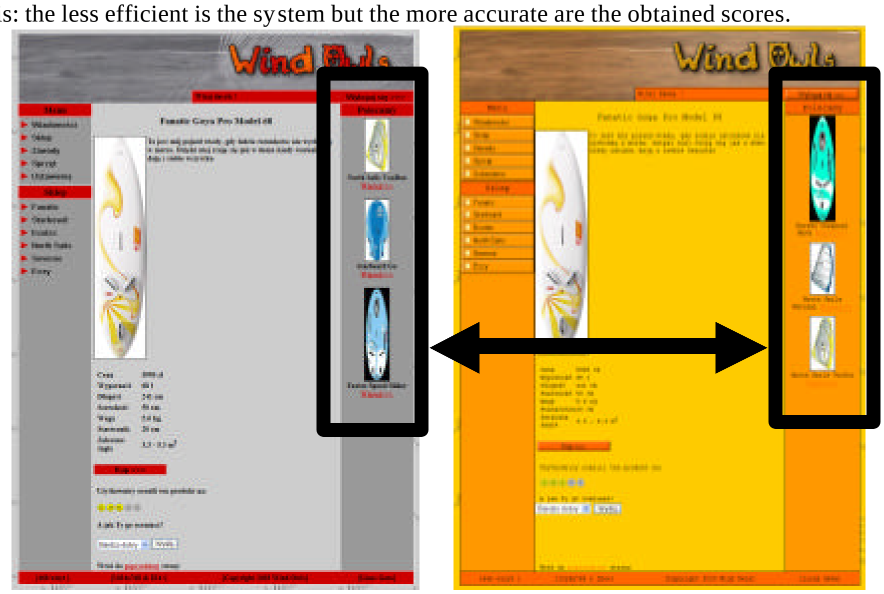

Figure 4. The same product page from the WindOwls website for two different users with different recommendations – fields with the black frames. Users can change the layout of their

interface

Figure 5. Two different pages with different recommendations (black frames) for the same user during one session

Two different users can be given completely different recommendations on the same page this overcomes the problem of ephemeral personalization [7] (Fig. 4). Additionally, the same user is suggested different lists of products on each page even during single user sessions (Fig. 5). Moreover, the same user on the same page can be proposed with distinct products due to the possible changes in their personal weights (2) or updates in method source data – Fig. 6.In consequence, based on the WindOwls system, we can deliver full personalization.

Figure 6. The same page with partly different recommendations (black frames) for the same user but at another time

6 Evaluation

The prototype WindOwls system was evaluated by 40 registered, logged in users in a real life simulation. They were invited to the specialized news group to use the website, and to rank and purchase some products. In total, 42 products were bought and 63 ranks were delivered. Besides these 25 not registered users, who only browsed through the e-commerce offer, used the system. Test data, which consisted of a set of 273 users clicks on recommendation within 102 user sessions, was divided into two groups: related to logged in users (Fig. 7a) and not logged in ones (Fig. 7b,c). System base weights on four stages were considered. At the beginning $w_i^{(0)} = 5$ were assigned to every method (0% of clicks). System base weights were recalculated before each stage based on all users' personal weights (w_{ik}), using formula (4), after 1/3 of all clicks (33%), after 2/3 of clicks (67%) and for all registered users clicks on recommendations (100%). Formula (1) was used for integration of recommendation methods instead of (5) i.e. without an additional component. The system worked either with normalization of personal weights (2) and (3) – Fig. 7a, 7c or without– using only (2), Fig. 7b.

Since scores of recommendation methods s_{ijk} are from the range [0,1], the relatively high value of the initial base weights $w_i^{(0)} = 5$ significantly reduced the influence of a single click on recommendation in (2) compared to e.g. $w_i^{(0)} = 1$.

After the first stage (0%), with a very limited number of users and their interactions, the demographic filtering provided the best recommendations for logged in users (Fig. 7a). After more users created their accounts and delivered much information to the system, association rules started to gain an advantage. The best buy at first and association rules after some time appeared to be the most effective recommendation method for not logged in users, whereas the best rated method appeared to be the worst (Fig. 7b,c).

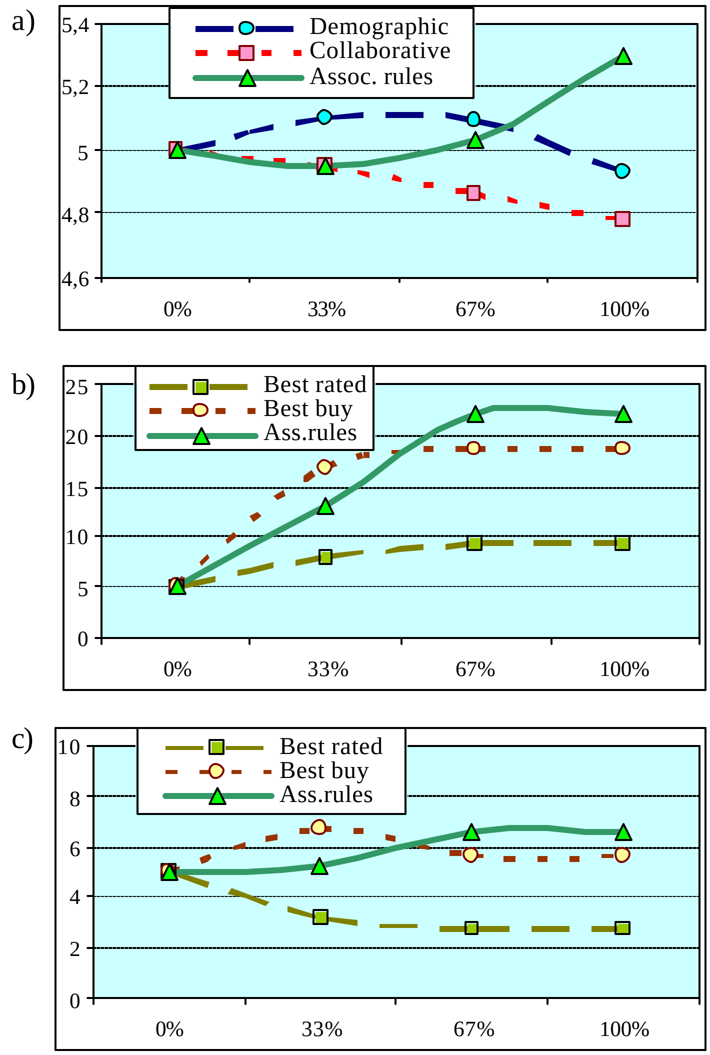

Figure 7. System base weights in relation to time, for logged in (a) and not logged in users without (b) and with normalization of component personal weights (c). The recalculation of system base weights was performed after 1/3 (33%), 2/3 (67%) and all (100%) monitored user clicks on recommendations.

Generally, we can observe that trends of change have settled after the first period i.e. after 1/3 of clicks: association rules start to win consistently while all other methods either lose, like demographic and collaborative filtering (Fig. 7a), or stabilize their significance like best buy and best rated (Fig. 7b,c). This stabilization of trends is a sign of the certain balance of the entire system.

7 Related Work

The integration of recommendation methods was usually performed in the non adaptive way, i.e. the contribution of each method was either unchangeable over the course of time or independent from the user. The opposite approach was proposed in [21] by the introduction of a coordinator agent. It gathers ordered suggestion lists from three recommendation agents and it integrates them combining with weights. A weight corresponding to the particular agent is periodically updated according to the popularity of suggestions delivered by that agent. The described above approach is more personalized, since it exploits an individual set of weights that are separately assigned to, and constantly updated for each user.

Yet another approach was presented by Golovin and Rahm in [9]. They used rules of type:

<*CurrentContent*, *CurrentUser*, *CurrentTime*> => <*RecomendedContent*, *Weight*>

where *the content* denotes a product from the e-commerce offer. Weights of these rules are modified by the learning module based on the feedback from the user i.e. the weight for the selected recommendation is increased whereas it is decreased for all the others which have been presented. This approach includes learning facilities but only top ranked rules are used in recommendation. We can say that the individual pairs *product1*, *product2*, i.e. specific recommendations are adjusted, rather than recommendation methods. Consequently, after some time, items selected by the user will have greater weights assigned and they will be more likely to be recommended to this user in the future. Nevertheless, it appears that the user will not be interested in recommendation of items they have already visited. Hence, the usefulness of emphasizing the particular, already clicked items for individual users is questionable. In our approach we operate on methods rather than on recommended items and the feedback information is used for the adjustment of methods, not for particular products.

The overall idea of the use of weights for items and their adaptation according to the user behavior was used by Bollacker *et al*. to recommend scientific literature in the CiteSeer system [5].

Shahabi and Chen combined clustering of web usage data and content analysis techniques to achieve predefined *user wish-list* and genetic algorithms technique for the assignment of the current user to the most appropriate list in their Yoda system [31]. However, this approach is more similar to general concept of collaborative filtering and some other adaptive techniques of user assignment [15, 16, 23] rather than to the learning mechanism presented in this paper.

8 Conclusions and Future Work

The concept of adaptive recommendations and its implementation in the WindOwls system presented in the paper appeared to be effective in adapting to the user's needs and its main advantage over single recommendation method is the full personalization that provides users with a dynamic list of products most likely to be interesting.

Due to the update of weights of recommendation methods, the presented system includes new adaptive, learning capabilities that allow it to reward the most efficient methods and discard others. This concept appeared to be effective since the personal weights have significantly changed their average values at the beginning and less over the course of time – see greater changes for 33% of click (compared to 0%) and lower modifications for the rest, Fig. 7b,c. Thus, we can say that the system comes, in a sense, to the certain balance. Also the trend of changes on Fig. 7a has been settled after 33% of clicks: association rules method has permanently increased in importance while all other methods have lost their significance.

Additionally, the proposed method is open for introduction of new recommendation methods based for example either on user navigation patterns [13, 15] or on textual content of web pages [14].

The evaluation of the system provided some supplementary conclusions, namely, recommendation performed with association rules appeared to be the most useful method but only after some time the system has worked (Fig. 7).

Future work will focus on negative feedback available in some methods (e.g. badly ranked products in collaborative filtering). It would benefit the system to utilize such opinions and to lower the score of bad products even if other methods show them as recommendable. At this approach, the system would have to resign from using only L best items from each method, because it cuts off most of the negatively rated products.

Differences in normalized weights of methods are relatively small; at the test end, they did not exceed 10% (Fig. 7a) for logged in users and 25% for anonymous ones (Fig. 7c). The result can be more diverse, if a smaller value of initial weights were used. However, in such cases, the danger of overfitting may appear and some recommendation methods can be permanently excluded from recommendation.

References

[1] G. Adomavicius, A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," *IEEE Transactions on Knowledge and Data Engineering* 17 (6), 2005, pp. 734-749.

[2] G. Adomavicius, A. Tuzhilin, "Using Data Mining Methods to Build Customer Profiles," *IEEE Computer* 34 (2), 2001, pp. 74-82.

[3] M. Balabanovic, Y. Shoham, "Fab: Content-based, collaborative recommendation," *Communications of the ACM* 40 (3), 1997, pp. 66-72.

[4] C. Basu, H. Hirsh, W. Cohen, "Recommendation as Classification: Using Social and Content-Based Information in Recommendation," in *Proc. 15th National Conf. on AI and 10th Innovative Applications of AI, AAAI/IAAI 98*, Madison, Wisconsin, United States, July 26-30, 1998, AAAI Press/The MIT Press, 1998, pp. 714-720.

[5] K.D. Bollacker, S. Lawrence, C.L. Giles, "Discovering Relevant Scientific Literature on The Web," *IEEE Intelligent Systems* 15 (2), 2000, pp. 42–47.

[6] Y.H. Cho, J.K. Kim, S.H. Kim, "A personalized recommender system based on web usage mining and decision tree induction," *Expert Systems with Applications* 23, 2002, pp. 329-342.

[7] J. Chun, J.-Y. Oh, S. Kwon, D. Kim, "Simulating the Effectiveness of Using Association Rules for Recommendation Systems" *Third Asian Simulation Conference, AsiaSim 2004*, Jeju Island, Korea, October 4-6, 2004, Revised Selected Papers. LNCS 3398 Springer, 2005, pp. 306-314.

[8] M. Géry, M.H. Haddad, "Evaluation of web usage mining approaches for user's next request prediction," *Fifth ACM International Workshop on Web Information and Data Management, WIDM 2003*, New Orleans, Louisiana, USA, November 7-8, 2003, ACM Press, 2003, pp. 74-81.

[9] N. Golovin, E. Rahm, "Reinforcement Learning Architecture for Web Recommendations," *International Conference on Information Technology: Coding and Computing ITCC'04*, Volume 1, April 5-7, 2004, Las Vegas, Nevada, USA, IEEE Computer Society, 2004, pp. 398-402.

[10] S.H. Ha, "Helping Online Customers Decide through Web Personalization," *IEEE Intelligent Systems* 17 (6), 2002, pp. 34-43.

[11] K.-Y. Jung, J.J. Jung, J.H. Lee, "Discovery of User Preference in Personalized Design Recommender System through Combining Collaborative Filtering and Content Based Filtering," *6th International Conference on Discovery Science DS'03*. Sapporo, Japan, October 17-19, 2003, LNCS 2843, Springer Verlag, 2003, pp. 320-327.

[12] P. Kazienko, "Multi-agent System for Web Advertising," *9th International Conference on Knowledge-Based Intelligent Information & Engineering Systems KES 2005*, Melbourne, Australia, September 14-16, 2005, LNAI 3682, Springer Verlag, 2005, pp. 507-513.

[13] P. Kazienko, "Product Recommendation in E-Commerce Using Direct and Indirect Confidence for Historical User Sessions," *7th International Conference on Discovery Science, DS'04*, Padova, Italy, October 2-5, 2004, LNAI 3245, Springer Verlag, 2004, pp. 255-269.

[14] P. Kazienko, M. Kiewra, "Integration of Relational Databases and Web Site Content for Product and Page Recommendation," *8th International Database Engineering & Applications Symposium, IDEAS '04*, Coimbra, Portugal, July 7-9, 2004, IEEE Computer Society, 2004, pp. 111-116.

[15] P. Kazienko, M. Kiewra, "Personalized Recommendation of Web Pages," Chapter 10 in: T. Nguyen (ed.) *Intelligent Technologies for Inconsistent Knowledge Processing*. Advanced Knowledge International, Adelaide, South Australia, 2004, pp. 163-183.
[16] P. Kazienko, M. Kiewra, "ROSA - Multi-agent System for Web Services Personalization," *First Atlantic Web Intelligence Conference, AWIC'03*. Madrid, Spain, May 5-6, 2003, LNAI 2663, Springer Verlag, 2003, pp. 297-306.
[17] P. Kazienko, P. Kolodziejski, "WindOwls – Adaptive System for the Integration of Recommendation Methods in E-commerce," *Third Atlantic Web Intelligence Conference, AWIC 2005*, Lódz, Poland, June 6-9, 2005, LNAI 3528, Springer Verlag, 2005, pp. 218-224.
[18] P. Kazienko, K. Kuzminska, "The Influence of Indirect Association Rules on Recommendation Ranking Lists," *5th International Conference on Intelligent Systems Design and Applications, ISDA 2005, RAAWS 2005*, Wroclaw, Poland, September 8-10, 2005, IEEE Computer Society, 2005, pp. 482-487.
[19] B. Krulwich, "Lifestyle Finder: Intelligent User Profiling Using Large-Scale Demographic Data," *AI Magazine* 18 (2), 1997, pp. 37-45.
[20] R.D. Lawrence, G.S. Almasi, V. Kotlyar, M.S. Viveros, S.S. Duri, "Personalization of Supermarket Product Recommendations," *Data Mining & Knowledge Discovery* 5(1/2), 2001, pp. 11-32.
[21] M. Lim, J. Kim, "An Adaptive Recommendation System with a Coordinator Agent," in *Proc Web Intelligence, WI 2001*, Maebashi City, Japan, October 23-26, 2001, LNAI 2198, Springer Verlag, 2001, pp. 438-442.
[22] B. Mobasher, H. Dai, T. Luo, M. Nakagawa, "Effective Personalization Based on Association Rule Discovery from Web Usage Data," *3rd ACM Workshop on Web Information and Data Management WIDM01*, in Conjunction with ACM CIKM 2001, Doubletree Hotel Atlanta-Buckhead, Atlanta, Georgia, USA, ACM Press, 2001, pp. 9-15.
[23] B. Mobasher, H. Dai, T. Luo, Y. Sun, J. Zhu, "Integrating Web Usage and Content Mining for More Effective Personalization," *Electronic Commerce and Web Technologies, First International Conference EC-Web 2000*, London, UK, September 4-6, 2000, LNCS 1875 Springer Verlag, 2000, pp. 156-176.
[24] R.J. Mooney, L. Roy, "Content-based book recommending using learning for text categorization," *Fifth ACM Conference on Digital Libraries*, June 2-7, 2000, San Antonio, USA, 2000, pp. 195-204.
[25] M. Montaner, B. López, J.L. de la Rosa, "A Taxonomy of Recommender Agents on the Internet," *Artificial Intelligence Review*, Kluwer Academic Publishers. 19 (4), 2003, pp. 285-330.
[26] D. O' Sullivan, D. Wilson, B. Smyth, "Using Collaborative Filtering Data in Case-Based Recommendation," *Fifteenth International Florida Artificial Intelligence Research Society Conference FLAIRS 2002*, May 14-16, 2002, Pensacola Beach, Florida, USA, AAAI Press, 2002, pp. 121-125.
[27] S. Perugini, M.A. Goncalves, E.A. Fox, "Recommender Systems Research: A Connection-Centric Survey," *Journal of Intelligent Information Systems* 23 (2), 2004, pp. 107-143.
[28] J.B. Schafer, J.A. Konstan, J. Riedl, "E-Commerce Recommendation Applications," *Data Mining and Knowledge Discovery* 5 (1/2), 2001, pp. 115-153.
[29] J.B. Schafer, J.A. Konstan, J. Riedl, "Recommender systems in e-commerce," *ACM Conference on Electronic Commerce EC-99*, November 3-5, 1999, Denver, USA, 1999, pp. 158-166.
[30] A.I. Schein, A. Popescul, L.H. Ungar, D.M. Pennock, "Methods and Metrics for Cold-Start Recommendations," *Proc. of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, August 11-15, 2002, Tampere, Finland, ACM Press, 2002, pp. 253-260.
[31] C. Shahabi, Y. Chen, "An Adaptive Recommendation System without Explicit Acquisition of User Relevance Feedback," *Distributed and Parallel Databases* 14 (2), 2003, pp. 173-192.
[32] D. Wang, Y. Bao, G. Yu, G. Wang, "Using Page Classification and Association Rule Mining for Personalized Recommendation in Distance Learning," *Advances in Web-Based Learning, First International Conference, ICWL 2002*, Hong Kong, China, August 17-19, 2002, LNCS 2436, Springer Verlag, 2002, pp. 363-376.