WindOwls – Adaptive System for the Integration of Recommendation Methods in E-commerce

Przemysław Kazienko[1] and Paweł Kołodziejski[1]

Wrocław University of Technology, Institute of Applied Informatics,
Wybrzeże S. Wyspiańskiego 27, 50-370 Wrocław, Poland
kazienko@pwr.wroc.pl, super@hot.pl

Abstract. WindOwls, the new hybrid system of personalized product recommendation in e-commerce, by integrating various methods, is 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, WindOwls can adjust the influence of individual methods. Testing implementation and evaluation of recommendation efficiency are also described.

1 Introduction

Recommendation systems are an important part of recent e-commerce. They enable the increase of sales by suggesting to users selected products from the offer. The problem how to choose the most suitable items, possibly with respect to the user's inclinations, is a challenging research problem. Four fundamental approaches to recommendation can be mentioned: demographic filtering, collaborative and content-based recommendation, and simplified statistical approaches [7]. In demographic recommendation, users are classified based on their personal data, which they themselves provided during the registration process [10]. Alternatively, this data can be extracted from the purchasing history, survey responses, etc. Each product is assigned to one or more classes with a certain weight and the user is suggested with items from the class closest to their profile. 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 are in the strongest correlation with the current user [3]. Content-based recommendation focuses on the similarity between products, usually taking into account their features like textual descriptions [6], hyperlinks, related ratings [12], or co-occurrence in the same purchased transactions or web user sessions [5]. Items that are the closest to the most recently processed, are ones suggested regardless of user preferences. In the statistical approach, the user is shown products based on some statistical factors; usually popularity measures like average or summary ratings, and numbers of sold units [13]. Most recommendation methods have significant limitations. 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 [5]. As the remedy for these and other shortcomings, many hybrid systems were proposed [2–4, 7]. The integration of recommendation methods was usually performed in the not adaptive way, i.e. the contribution of each method was either unchangeable over the course of time either independent from the user. The opposite approach was proposed in [11] 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. This approach was extended, concretized and first of all personalized in the described below WindOwls system, in which an individual set of weights is separately assigned to, and constantly updated for each user. The overall idea of the use of weights for items and their adaptation according to the user behavior was used by Bollacker *et al.* in the CiteSeer system for scientific literature [1].

2 Personalized Integration of Recommendation Methods

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 web page but be the same for all users. Persistent personalization uses the history of user's behaviour and generates different product list for each user, but it works only with logged in users [13]. The main goal of WindOwls system is to overcome shortcomings of a single 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 user's likes and dislikes (persistent personalization) as well as on effectiveness of previous recommendations for the given user (adapted personalization). To achieve full personalization, WindOwls system combines association rules for ephemeral content-based personalization and collaborative and demographic filtering for persistent one. Thus, WindOwls is a complete hybrid recommendation system integrating many independent recommendation methods in personalized and adaptive way. It exploits weights that are dynamically recalculated according to the effectiveness of 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 WindOwls system 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. When WindOwls is running for the first time 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 work of WindOwls starts with the user's interaction (Fig. 1). The context of interaction (the requested 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 while association rules require URL. The WindOwls 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 WindOwls 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 while demographic recommendation is based on matching personal data: likes and dislikes, pre-owned products, annual expenses on certain category, etc. To improve recommendation quality, also association rules were introduced. They reflect cases in which a given product was purchased together with the set of another ones 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 the other ones, e.g. news pages or so-called white pages [6]. 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 [6]. Note that all other considered recommendation methods are insensitive to the type of the requested page. Each method is independent from all others and it is provided by WindOwls only with the context data (URL, UID).

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

$$f_{jkl} = \frac{1}{\max_{kl}} \sum\nolimits_{i=1}^{M} w_{ik} \cdot s_{ijkl}, \quad s_{ijkl} \geq 0, \tag{1}$$

where: f_{jkl} — the final score of product j for user k in context (page) l; w_{ik} — the current weight of the 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 methods, in the implementation $M = 5$; $\max_{kl}$ — maximum value of score s_{ijkl} among scores returned by method i — the top one in ranking i. Factor $1/\max_{kl}$ 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 the context determines also the product, in the case when the user requests a product page. Each method delivers only K items (products) to reduce

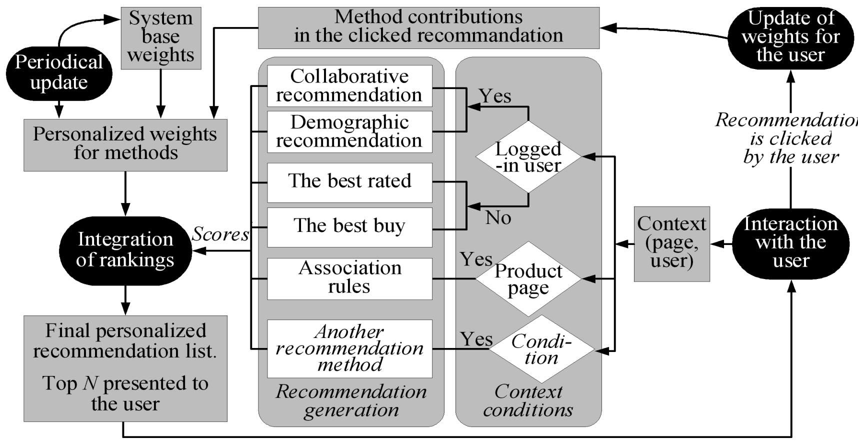

Fig. 1. Personalized integration of various methods of recommendation

processing and it appears that K equal to about $N \cdot M$ would be enough; where N is the number of recommendations suggested to user. $N = 3$ and $K = 10$ were assumed in the implementation. Due to method preconditions, only three recommendation methods are able to supply suggestions at the same time. The greater K is, the less efficient is the system but the more accurate are obtained scores.

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. If a user chooses one of recommendations linking to product j, WindOwls checks what score s_{ijk} had each i-th method in recommending this product and it adequately updates weights of all methods in the set of user k:

$$\begin{aligned} w_{ik}^{(1)} &= w_i^{(0)} + s_{ijk}, \text{ after the first click on recommendation by user } k, \\ w_{ik}^{(n+1)} &= w_{ik}^{(n)} + s_{ijk}, \text{ after the } (n+1)\text{-th click.} \end{aligned} \tag{2}$$

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

After user's interaction, the described cycle repeats.

3 Implementation and Evaluation

The WindOwls system was implemented as an e-commerce windsurfing website (www.windowls.smellme.netlook.pl). It contains sections with news, articles, shop and settings. On the settings page users can change their personal information about their interests used for demographic filtering. A typical page in the shop contains a description of a single product with the possibility of

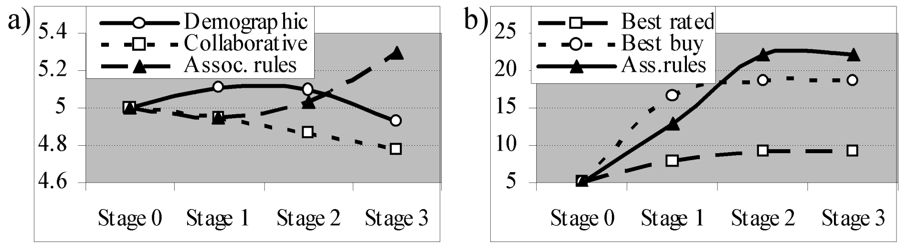

Fig. 2. System base weights for logged in (a) and not logged in users (b)

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 navigation. Furthermore, even the same user on the same page can be proposed with different products due to the possible changes in their personal weights (2) or updates in methods source data.

The WindOwls system was evaluated by 40 registered, logged in users in a real life simulation. They were invited on the specialized news group to use the website, and to rank and purchase products. In total, 42 products were bought and 63 ranks were delivered. Besides, 25 not registered users, who only browsed through the e-commerce offer, used the system. Test data, which consisted of the set of 273 users clicks on recommendation within 102 user sessions, was divided into two groups: related to logged in users (Fig. 2a) and not logged in ones (Fig. 2b). System base weights on four stages were considered: at the beginning $w_i^{(0)} = 5$ were assigned to every method (stage 0), after 1/3 of all clicks (stage 1), after 2/3 of clicks (stage 2) and for all registered users clicks on recommendations (stage 3). System base weights were recalculated before each stage based on all users' personal weights (w_{ik}), but normalization was performed only in the logged in mode. After stage 1, with a very limited number of users and their interactions, demographic filtering provided best recommendations for logged in users (Fig. 2a). After more users created accounts and delivered much information to the system, association rules started to gain an advantage. The best buy at first and association rules after stage 1 appeared to be the most effective recommendation method for not logged in users (Fig. 2b). Other tests and results can be found in [9].

4 Conclusions and Future Work

The WindOwls hybrid system appeared to be effective in adapting to the user's needs and its main advantage over single recommendation method is 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, WindOwls includes new adaptive capabilities that allow it to reward most efficient methods and discard others. It is open for introduction of new recommendation methods based for example either on user navigation patterns [5,7] or on textual content of web pages [6].

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 K best items from each method, because it cuts off most of negative rated products. All recommendation methods need their base knowledge to be periodically updated offline because new ratings, users and products would appear. Some of them share the same source data, so the update process should be synchronized among all methods [8]. Differences in normalized weights of methods are very small; at the test end, they did not exceed 10% (Fig. 2a). It is the effect of respecting weights of users that did not click any recommendations. It will be improved in the next version of the system.

References

1. Bollacker K.D., Lawrence S., Lee Giles C.: Discovering Relevant Scientific Literature on The Web. IEEE Intelligent Systems, **15** (2) (2000) 42–47.
2. Cho Y.H., Kim J.K., Kim S.H.: A personalized recommender system based on web usage mining and decision tree induction. Expert Systems with Applications **23** (2002) 329–342.
3. Ha, S.H.: Helping Online Customers Decide through Web Personalization. IEEE Intelligent Systems **17** (6) (2002) 34–43.
4. Jung K.-Y, Jung J.J., Lee J.H.: Discovery of User Preference in Personalized Design Recommender System through Combining Collaborative Filtering and Content Based Filtering. DS'03, LNCS **2843**, Springer Verlag (2003) 320–327.
5. Kazienko P.: Product Recommendation in E-Commerce Using Direct and Indirect Confidence for Historical User Sessions. DS'04, LNAI **3245** (2004) 255–269.
6. Kazienko P., Kiewra M.: Integration of Relational Databases and Web Site Content for Product and Page Recommendation. IDEAS '04, IEEE Computer Society (2004) 111–116.
7. Kazienko P., Kiewra M.: Personalized Recommendation of Web Pages. Chapter 10 in: Nguyen T. (ed.) Intelligent Technologies for Inconsistent Knowledge Processing. Advanced Knowledge International, Adelaide, South Australia (2004) 163–183.
8. Kazienko P., Kiewra M.: ROSA — Multi-agent System for Web Services Personalization. AWIC'03, LNAI **2663**, Springer (2003) 297–306.
9. Kołodziejski P.: Methods of Product Recommendation in E-commerce. Master Thesis. Wrocław Univeristy of Technology (2004), in Polish.
10. Krulwich, B.: Lifestyle Finder: Intelligent User Profiling Using Large-Scale Demographic Data. AI Magazine **18** (2) (1997) 37–45.
11. Lim M., Kim J.: An Adaptive Recommendation System with a Coordinator Agent. WI 2001, LNAI **2198**, Springer Verlag (2001) 438–442.
12. Mooney R. J., Roy L.: Content-based book recommending using learning for text categorization. Fifth ACM Conference on Digital Libraries (2000) 195–204.
13. Schafer, J.B., Konstan, J.A., Riedl J.: E-Commerce Recommendation Applications. Data Mining and Knowledge Discovery **5** (1/2) (2001) 115–153.