2008 International Conference on Internet Computing in Science and Engineering

Personalized E-commerce Recommendation Based on Ontology

Peiguang Lin[1], Feng Yang[1], Xiao Yu[1], Qun Xu[2]
[1]*School of Computer & Information Engineering, Shandong University of Finance, Jinan 250014, P.R. China*
[2]*Qilu Software Park, Jinan 250041, P.R. China*
llpwgh@sdfi.edu.cn

Abstract

The current collaborative recommendation approaches mainly measure users' similarity by comparing user's entire interests and don't consider user's interest quality, especially interest span. With so many goods in the E-commerce web site, how to get the needed product quickly so as to promote the efficiency of E-commerce system? This paper presented a personalized recommendation method based on ontology. To improve the precision, we firstly divided users' interests into long-time interests and short-time interests; and then by use of the principle of partial similarity, the recommendation mechanism and algorithm were given. Lastly, based on the method above, a prototype system was presented and the system test was done. Experimental results indicate that this method can recommend related products in the majority to target users and it can be practical.

1. Introduction

With the popularization of the Internet and the development of E-commerce, the E-commerce system's structure becomes more complicated when it provides more and more choices for users. So users usually get lost in the vast space of commodity information and can not find the goods they really want. Under the increasingly intense competitive circumstance, the E-commerce recommender system can effectively reserve users, keep them from losing and increase the cross selling ability. According to the research, with the personalized recommender system used in E-commerce, the sales improved by 2-8%[1] , especially in those industries such as books, movies, CD etc, and great in extent of using personalized recommender system. The recommender system can greatly boost sales.

At present, there have been many collaborative recommendation systems [00]. There is a relatively large gap between the recommender function of E-commerce in China and that in other countries. And our theoretical research in personalized and automatic recommendation is almost a blank. If we search for the articles containing "recommender system" in CNKI (China Net Knowledge Intelligent), little can we find. It shows that our research on recommendation had fallen far behind others [0].

This paper talks about an E-commerce collaborate recommender method based on ontology. Firstly, it introduced user's interests model (UIM) based on ontology, and then given the calculation of similarity between the user and the goods in web site. Detailed recommendation Mechanism and algorithm were following presented. Lastly, the evaluation of this algorithm was done and the conclusion was given.

2. Ontology-Based User's Interests Model

To realize personalized recommendation based on ontology, the user-oriented interest model must be created firstly. According to Personality Psychics, human's interests are different in tendency, span, stability and functionality, which are called interest quality [00].User's interests are relative centralized and have several aspects of interests called **interest-points**. The existing recommendation systems don't consider user's interest quality, especially interest span. They construct user's neighbors based on user's entire interests by comparing their all history behavior and neglect all aspects of interests. Hence, the system can't have the expected result even if the user only expects to focus on an aspect of his interests[0].

To implement the system efficiently, we divide user's interests into several interest-points when constructing user's interest model. Using this model, we can measure users' similarity on a certain interest-point and carry on recommendation based on neighbors with partially similar interests.

Interest span reflects how many interest-points the user has and other characteristics of **interest quality** are reflected by the quality of each interest-point, including tendency and stability, as shown in Table 1.

978-0-7695-3112-0/08 $25.00 © 2008 IEEE
DOI 10.1109/ICICSE.2008.69

Based on above, this paper presented a new algorithm of User's Interest Model based on Ontology. This algorithm thinks about not only the interest quality, but the difference of long-time interests and short-time interests. By use of the ontology to express the model, users' interest can be expressed more exactly and the recommendation system can perfect more efficiently.

2.1. Ontology, Description Logic and Knowledge Base

Tim Berners-Lee proposed the concept of Semantic Web in 1998, whose target is to develop a series of languages and techniques which can express semantic information, and can be understood and processed by computers. The semantic web is to support the abroad and effective auto-inference on the net environment. The implementation of the semantic web is based on ontologies which are defined as a formal, explicit specification of a shared conceptualization by Studer [0] and some others. The ontology has perfect concept hierarchy and supports logic inference, so it is widely used.

Description Logics (DLs) are a well-known family of knowledge representation formalisms based on the notion of concepts (classes) and roles (properties). DLs have been proved useful in wide range of applications including configuration, databases and ontological engineering (i.e., the design, maintenance and deployment of ontologies) [0]. So we adopt DL to describe User's Interest and Knowledge Base of products.

2.2. UIM-O：User's Interests Model Based on Ontology

To realize the personalized recommendation and promote the accuracy of the recommendation system, with thinking about the quality of the interest, we divided user's interest into multi-IPs (**Interest Point**), while long-time interest and short-time interest are given. Then bi-dimension of user's interest model is created by above. In addition, the E-commerce website usually classified the products by some standards, as China-pub (http://www.china-pub.com) divided the books into computer, foreign language and economics and management etc, and each book is belong to one catalog; A top catalog can be divided into more sub ones, as the catalog of computer is divided into database, operating system, software engineering etc. As user viewing books, it also embodies that the user is interested in this catalog and this catalog is named as one *IP* (Interest Point). Then we can create the following knowledge base about products: *EC_KB*= (*Tbox, ABox*). From the view of ontology, *TBox* represents the "Terminology Box" which can describe the general knowledge (concept). Here it is used to record the information of catalogs and the catalog levels; *ABox* represents the "Assertional Box" which can describe the relation of product and catalog.

Table 1. Representation of quality of interest-point

Quality	Representation	Description
tendency	Degree(W)	Interest degree to interest-point IP
	WeightS	Interest weight vector to each product of Interest Point *IP*
stability	Count(IP)	The number of access times to IP
	RAD	Recent access density on IP

For example, there is a book whose number is *B25351*, and its catalog is Computer/Database/Oracle (Catalog number: *C08-04*), then *C08-04* (*B25351*) can be used to describe that is *B25351*an instance of catalog *C08-04*, and we can use *Rela* (*B25351, B098*) to represent the book *B25351* and *B098* are often saled at the same time.

So we get the following model of user's interest: $O(u_k)=(IP_1,IP_2,...,IP_n)$. In this model, $O(u_k)$ is used to describe the n-IPs of user u_k, and $IP_i=(C_m, W_i, RAD_i)$, which means user u_k be interested in the catalog C_m, W_i is the **interest degree** of the catalog C_m, the bigger W_i is, more interested the user is in catalog C_m. We can get the value of W_i by Eq. 1.

$$W_i = \frac{count_{k,m}(u_k, C_m)}{\sum_m count_{k,m}(u_k, C_m)} \text{ and } \sum_{i=1}^{n} W_i = 1 \quad (1)$$

RADi is used to describe the recent access density of user u_k which represents the translation of user's interest point. Given user u_k and an interest-point *IP*, the first session user A accessed *IP* is *First(IP)*, the last session u_k accessed *IP* is *Last(IP)*, the current session is *Sc*, the number of all sessions between *First(IP)* and *Sc* is *Ns*. *Count (IP)* is the number of sessions that contain *IP between First(IP)* and *Sc.* Then user u_k's recently access density on *IP* is:

$$RAD_i = \frac{count(IP)}{Ns} = \frac{count_{k,m}(u_k, C_m)}{\sum_m count_{k,m}(u_k, C_m)} \quad (2)$$

Here, we use the expired time(denoted by θ_1) to restrict the interval between the last and the current session. When the interval exceeds θ_1, both *First(IP)* and *Last(IP)* are changed to the current session.

Given thresholds of the minimal *RAD*(denoted by α) and the minimal *Count(IP)* (denoted by β), for an interest-point *IP* of user u_k, if the *Count(IP)* and *RAD* are both greater than their thresholds, the interest-point *IP* is the user u_k's long-time interest, otherwise if the *RAD* is greater than α but the *Count(IP)* is less than β, the interest-point *IP* is the user u_k's short-term interest. If the *Count(IP)* is greater than β, but the *RAD* is less than α, this means that user u_k has been interested in *IP* once, but not interested in it now.

To compute W_i and RAD_i conveniently, this system maintain two matrixes, which record the user's history of visited information and recent visited information, as showed in Eq. 3.

$$\begin{pmatrix} & times\ of\ user\ u_k\ visiting\ catalog\ C_m \\ \text{User Information} & count_{k,m}(u_k, C_m) \end{pmatrix} \quad (3)$$

This matrix is updated while user visits the web site.

Lastly, if a user has some long-time interest, generally he has rich experiences then his visiting history can be recommend to others with high reliability. The following is the equation to quantify user's experiences.

$$E_{u_k,IP_i} = \lambda_1 \times W_i + \lambda_2 \times RAD_i \quad (4)$$

If Eq. 4, λ_1, λ_2 is the coefficient and $\lambda_1 + \lambda_2 = 1$, which can be defined by user himself.

3. Recommendation Mechanism & Algorithm

3.1. Partial Similarity of Users' Interests

The existing recommendation systems construct user's neighbors based on user's entire interests and don't consider interest quality, neglect that users usually have several interest-points. Hence, when measuring the similarity of users, even if two users are very similar in an aspect of interests, the whole similarity is not always great. In order to obtain the similar users, these systems have to reduce the similar threshold to cluster users, which causes the dissimilar users clustered into the same group and then reduces the precision of the system. When comparing users' similarity, the long-term interests inevitably act as kernel role. For happening very casually and stochastically, short-term interests always are the noise data in measuring the users' similarity.

For example, as Fig. 1 shows, user u_1's interest set is IP_1 = {Software, Hardware}, and user u_2's interest set is IP_2 = {Database, Software}. According to existing recommending algorithm, if u_1 and u_2 are highly similar on "Software", and the entire interest similarity also is higher than the system threshold, then "Hardware" will be recommended to u_2 and "Database" will be recommended to u_1. It is inappropriate obviously. Moreover, when calculating users' similarity, although u_1 and u_2 are highly similar on "Software" and can be neighbors mutually, the existence of user u_1's "Hardware" and user u_2's "Database" may cause that u_1, u_2 are not neighbors, and then lose the significant content to recommend.

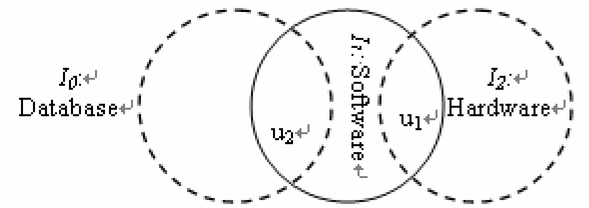

Fig. 1. Clustering users into IG (Interest Group)

In view of user's interest span, this paper constructs neighbors by clustering users into groups based on partial similarity of users' interests and assigns target user to a certain interest-group for one of his interest-point, so a user can belong to many interest-groups. Here **interest-group (IG)** is defined as the user group with high similarity on certain aspect of interests. Figure 1 demonstrates how to cluster users into IGs. I_0, I_1, I_2 respectively represents "soccer", "personalization", "music" IG. The target user u_1 is similar to user $u_2 u_3 u_4$ on interest-point I_1, and similar to $u_4 u_5 u_6$ on I_2.

In order to implement this, we give the user u's vector of Interest Point: $W_{u,IP} = < c_{u,1}, c_{u,2}, ..., c_{u,n} >$, in which $c_{u,i}$ is the frequency user u has visited the product c_i of Interest Point *IP*, and this variable can be computed from the history matrix.

Given an interest-point *IP*, the similarity of user u_1 and u_2 on *IP* is defined as:

$$Sim(u_1, u_2, IP) = \frac{\sum_i c_{u_1,i} c_{u_2,i}}{\sqrt{\sum_i c_{u_1,i}^2}\sqrt{\sum_i c_{u_2,i}^2}} \quad (5)$$

3.2. Recommendation Mechanism

We adopt different recommendation mechanisms for user's long-time interests and short-term interests. The recommendation to target user's long-term interests is based on neighbors with partially similar interests and the recommendation to user's short-term interests is based on experienced users.

If target user u's current interest-point IP_i is one of his long-time interests, we provide recommendation to him based on neighbors with partially similar interests on interest-point IP_i. Suppose n is the members of interest-group IP_i, u_k is user u's neighbor, $Sim(u, u_k, IP_i)$ is similarity between user u and user u_k on interest-point IP_i, $E(u_k, IP_i)$ is the experience of user u_k to interest-point IP_i, then the recommendation score $R_{A,IPi}$ for user u of interest-point IP_j correlated to IP_i is defined as follows:

$$R_{A,IP_i} = \frac{1}{n}\sum_{k=1}^{n} Sim(u, u_k, IP_i) * E_{A,IP_i} \qquad (6)$$

After obtaining the recommendation score $R_{u,IPi}$ according to Eq. 6, we sort IP_j in descending order of recommendation score and select the *TOPN* interest-points to recommend to target user u.

If target user u's current interest-point IP_i is one of his short-term interests, we provide recommendation to him based on the experienced users on interest-point IP_i. The function $R_{A,IPi}$ is used to calculate the recommendation score of page P_j in I_i based on *TOPN* experienced users in *IG* IP_i when target user u's current interest-point IP_i is his short-term interests, where W_{u_k,p_j} is the interest weight of user u_k to page P_j, and $E(u_k, I_i)$ is the experience score of user u_k on interest-point IP_i, N is the variable of *TOPN*.

$$R_{A,IP_i} = \frac{1}{N}\sum_{k=1}^{N} E_{A,IP_i} * W_{u_k,IP_i} \qquad (7)$$

3.3. Recommendation Function

Recommendation Function (RF) is used to get the recommended result by the target user's request. Given the user u, IP_A is the set of long-time interest of user u, *rc* is the current request of user u. Firstly, *RF* call for the function *AIP(rc)* to extract the Interest Point *IPi* and user's interest model; Secondly, *RF* judges whether the interest point group *IPi* exists in the interest group database (*IGD*) or not. If not existed, new interest point group is created in IGD; if existed, RF judges the interest point be long-time interest or short-time interest further. If long-time interest, the recommended result is created by Eq. 6; else it is created by Eq. 7. The following gives the Algorithm.

Algorithm 1 *RF(IMD, IGD, user A ,* r_c*)*

Input:

IMD is the interest model database; IGD is the interest-group database; user A is the target active user; r_c is user A's current request.

Output:

Predicted recommendation items *R* for user A.
R=NULL;
Let I_i be user A's current interest-point;
I_i = AIP(r_c);
If (IG $I_i \notin$ IGD) *// IGD don't have IG* I_i;
  Create (IGD, I_i); R = NULL;
Else
  if ($I_i \in I_A$) *//* I_i *is user A's long-term interests;*
  Let S_I be candidate set to recommend, $I_j \in S_I$;
  for each $I_j \in S_I$ do $R_{A,Ii}(I_j)$;
  R = TOPN($R_{A,Ii}(I_j)$);
Else *// if* I_i *is user A's short-term interests;*
  Let P_I be candidate set to recommend, $P_j \in P_I$;
for each $P_j \in P_I$ do $R_{A,IPi}(P_j)$;
  R = TOPN($R_{A,IPi}(p_j)$);
Return R;

---

4. Architecture & Workflow

4.1. Architecture

The architecture for the E-commerce Collaborate system based on ontology is shown as Fig.2.

The kernel of this system falls into three parts: data processing, recommender processing and reasoning Engine. The former mainly processes the model of users' interests, the secondly generates the recommended product(s) and the lastly does the retrieval of the products from EC_KB.

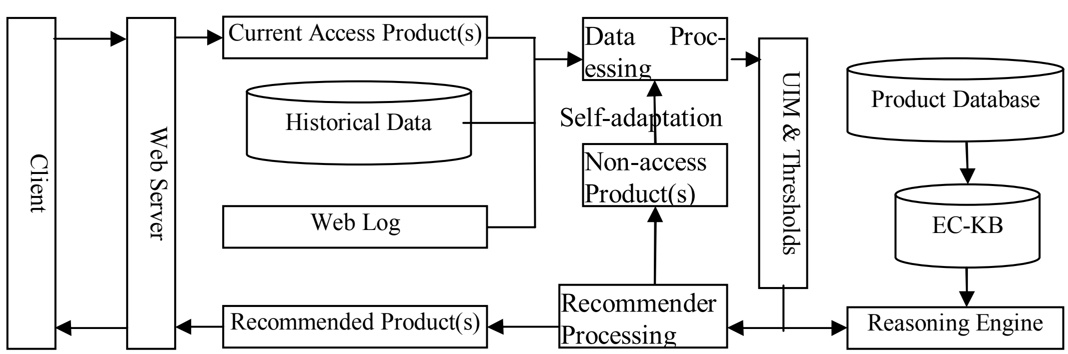

Fig. 1. Architecture for the E-commerce Collaborate system based on ontology

4.2. Workflow

The work flow of this recommender system goes like this: First, it preprocesses the current access list, historical trade data and Web log and so on, extracts the users' interests information, and constructs initial recommender model. By data processing, initial threshold is set. Then, it calculates the similarity between the initial recommender model and the products are obtained from EC_KB by Reasoning Engine. If the degree of similarity between the product and user's interests is bigger than or equal to the initial threshold, the products will join the recommender access list provided for the users. At last, according to the feedback information of recommender access list from the users, the system automatically adapts the recommender model and threshold so as to obtain the best recommender quality.

5. Experimental Results & Its Analysis

In order to test this system, we choose 2000 books and 5 catalogs form one web site as experimental data. In the experiment, we take the title, the abstract of these books as the introduction of products; these books as products and users' trade record as users' historical data, which can be found in web log on the server.

Usually the evaluation standard in information retrieval field is adopted to judge the recommending quality of system [00] , i. e. precision and recall rate:

$$precision = \frac{number\ of\ right\ recommended\ items}{number\ of\ all\ recommended\ items} \quad (8)$$

$$recall = \frac{number\ of\ right\ recommended\ items}{number\ of\ all\ related\ items} \quad (9)$$

Precision and recall are contradictory index to a martin degree. High precision means low recall. To balance the two, overall evaluation index F-measure is adopted.

$$F-measure = \frac{2pr}{p+r} \quad (10)$$

We can set initial recommender mode and threshold according to the current users' access list; historical trade record and Web log in the experiment. Then recommend based on each abstract and provide recommend lists in time order for users to confirm. The system modified the model according to the feed back information and recommender process the articles of next year awarding to the modified model. The experimental result shows in Table 2 and Fig.3. The experimental data show that the method presented by this paper is practical, From Fig.3, the highest precision is 95%, highest recall is 92% and the F-measure reaches 92%.

Table2. Experimental result

Catalog	All Related Books	Recommended Result	
		Right	Wrong
Database	620	533	46
Software Engineering	350	322	17
AI	330	290	32
Information System	470	414	36
E-commerce	230	207	16

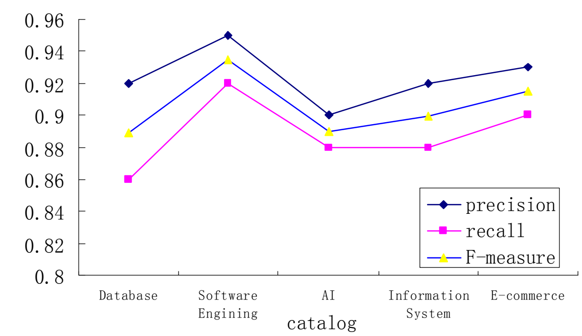

Fig. 3. Graph of system performance

6. Conclusion

The E-commerce personalized collaborate recommender system presented in this paper explored users' interest needs , recommended the products according to the qualitative value of products information, and automatically adapted to users' feedback information. In this way its comprehensive performance were enhanced.

Acknowledgement

This work was supported in part by the National Science Foundation Committee of P.R. China (60603070), the National Nature Science Foundation of Shandong Province (A200621), and the Education Department S&T Development Foundation of ShanDong (J06P07).

References

[1] R. D. Lawrence, G. S. Almasi, Y. Y Kotl, et al. "Personalization of Supermarket Personalization of Super-

market Product Recommendation," IBM Research Report, New York, 2004.

[2] C. Zeng, C. Xing and L. Zhou, "A Survey of Personalization Technology," *Journal of Software,* Vol. 13, No. 10, Science Press, Beijing, 2002, pp. 1952-1961

[3] 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*, Vol.7, No. 6, IEEE Computer Science, Washington , 2005, pp. 734-749

[4] W. He, Y. Cao, "An E-Commerce Recommender System Based on Content-Based Filtering," *Wuhan University Journal of Natural Sciences*, Vol. 11, No.5, Wuhan University Journal ,Wuhan 2006, pp. 1091-1096

[5] Y. Ye and K. Kong, *Personality Psychics,* East China Normal University Press. Shanghai. 1993

[6] Y. Guo, S. Bai, Z. Yang and K. Zhang, "Analyzing Scale of Web Logs and Mining Users' Interests," *Chinese Journal of Computers*, Vol.28, No.9, Science Press, Beijing, 2005, pp. 1483-1496

[7] R. Studer, V.R. Benjamins, D. Fensel, "Knowledge Engineering, Principles and Methods," *Data and Knowledge Engineering*, Elsevier Science Publishers B. V., The Netherlands, Vol. 25, No. 1 1998, pp. 161-197

[8] Baader F. et al, *The Description Logic Handbook: Theory*, Implementation and Applications. Cambridge: Cambridge University Press, 2002

[9] L Yu, L. Liu, "Research on Personalized Recommendations in E-business," *Computer Integrated Manufacturing Systems*, Vol. 10, No. 10, Beijing, 2004, pp. 1306-1313