A Personalization Recommendation Method Based on Deep Web Data Query

Tao Tan
School of Computer, China West Normal University, Nanchong, China
tantao99132@163.com

Hongjun Chen
Computer Department, Sichuan TOP Vocational Institute Of Information Technology, Chengdu, China
2009missxiaochen@163.com

Abstract— **Deep Web is becoming a hot research topic in the area of database. Most of the existing researches mainly focus on Deep Web data integration technology. Deep Web data integration can partly satisfy people's needs of Deep Web information search, but it cannot learn users' interest, and people search the same content online repeatedly would cause much unnecessary waste. According to this kind of demand, this paper introduced personalization recommendation to the Deep Web data query, proposed a user interest model based on fine-grained management of structured data and a similarity matching algorithm based on attribute eigenvector in allusion to personalization recommendation. Secondly, As for Deep Web information crawl, a crawl technology based on the tree structure is presented, with the traversal method of tree to solve the information crawl problems in the personalization service distributed in various web databases. Finally, developed a prototype recommendation system based on recruitment information, verified the efficiency and effectiveness of the personalization recommendation and the coverage and cost of Deep Web crawl through the experiment.**

Index Terms— **Deep Web, Personalization Recommendation, Similarity Matching, User Interest Model, data crawl**

I. INTRODUCTION

With the rapid development of internet technology, the number of web sites and web pages is growing in an explosive speed. Consequently, the web has now become a huge, widespread and global information center. The data contained in web is significant in many fields. The information seems disorderly and unsystematic, but seeing from the depth of the web information, it can be divided into two parts: the Surface Web and the Deep Web. Surface Web refers to the Web pages that can be searched by traditional search engines (e.g. Google, Yahoo, etc) through hyperlinks. Deep Web refers to the content that can not be indexed by traditional search engines, and these contents mainly stored in the online web databases. With the arrival of the era of web2.0, web database is widely applied in all kinds of dynamic web site, and web is also boosting its "deepening" process. At present the entire web has at least 450 000 accessible web databases and the number is still rapidly growing. The stored information covers every fields of the world, such as business, medicine, sports and others. Information in the Deep Web is 550 times over the Surface Web [1,2]. This makes Deep Web an important way for people to acquire information. Research on the Deep Web is attracting more and more attention from domestic and foreign scholars.

At present, the main way to obtain information from the Deep Web is to send a query through the query interface of dynamic web pages. Because of the special accessing method to web database, the content can not be indexed effectively by the traditional search engines. In order to help users effectively use tremendous information of the Deep Web, the researchers started researches on Deep Web data integration. Deep Web data integration can partly satisfy people's needs of information query in the Deep Web, but it is not intelligent, and cannot learn users' interest. As for the ones have specific professional interest, information needs to remain unchanged in quite a long period of time or changes a little, and users can only search the same content online repeatedly, causing much unnecessary waste. Driven by this demand, personalized service technology got rapid progress. If the query based on Deep Web data integration is in the service form of "people seeking information", then personalization recommendation is in the service form of "information seeking people". Find out users favor according to their behavior patterns, dynamically custom the query content for them and that will bring users great convenience.

This paper puts forward a user model based on fine-grained management of structured data. And build initial user interest model through the five element sets of user. Aimed at personalization recommendation, a similarity matching algorithm based on attribute eigenvector is proposed. As for Deep Web information crawl, a crawl technology based on the tree structure is presented, with the traversal method of tree to solve the information crawl problems in the personalization service distributed in various web databases. On the basis of the ideas above, we developed a prototype recommendation system based

doi:10.4304/jcp.7.7.1599-1606

on recruitment information and the experiments prove efficiency and effectiveness of the personalization recommendation and the coverage of Deep Web crawl.

II. RELATED PRINCIPLES

Deep Web consists of four basic components: web site, database, query interface and hyperlink. A website is a server, maintaining one or many backend database that saving addressable online information. Each backend database can be queried through one or many HTML Form; these query forms build their own query interfaces[17]. For example, Fig. 1 shows a job query form offered by a recruitment information site. Each query interface supports several attributes, for example, if you are searching for a position, it can be searched according to these attributes such as job categories, industry categories and work sites. These fields, attributes together with some semantic tags constitute the schema information of query interface. When you make a choice or input a keyword and then submit, the web site will return the result pages including information that meet your queries to you. The Fig. 2 shows the result page.

Figure 1. Query Interface Form of a Recruitment Information Site

Figure 2. Query Result Page of a Recruitment Information Site

Personalized service technology use user model to describe users' interest, calculate similarity of information and user model, recommend high correlation information to users. Because users' interests keep changing, so the construction of user interest model is a learning process, it updates according to the feedback information from users. The user interest model is put forward in the personalized service system, and it is a description form of user information demands. Its role is just like the query request of the Deep Web query, but user interest model is not the same with query request. In the personalized service system, the requirement information of users need to stay in the server for a long time, these information will update while the users' interest adjusting and refining; In the Deep Web data integration, the demand information of users is the query information which submitted by users, as for Deep Web data integration, every two queries do not related to each other, because the server does not make special reserve for this kind of information, it is a one-off thing.

In order to get accurate and necessary information from large-scale dynamic Deep Web, scholars started researches on data integration in recent years, and it has gradually become a hot topic in the area of database. So far, they have made achievements in this field, such as Deep Web pages obtain [4, 5, 6], query interfaces integration [7, 8], results data extraction and annotation [9, 10], etc. Moreover, personalization recommendation is getting mature in application, for instance, the construction of user interest model based on ontology[11],the user interest modeling and update strategy of extended semantic[12].They also made some achievements in the personalization recommendation based on Deep Web data. The literature [13] put forward a service frame based on the Deep Web personalized service, and it has been applied to the scientific literature retrieval, constructed the recommendation system of the science and technical literature.

Simultaneously, the development of Web enables network recruiting gradually become the main model for enterprise recruiting and individual job seeking, all kinds of recruitment websites flourish vigorously. Domestically, websites that provide recruitment information retrieval service are ChinaHR (http://www.chinahr.com/index.htm), Zhao Pin (http://www.zhaopin.com/), Jzgo (http://www.jzgo.cn/) etc. Beyond seas, there are Flipdogs (http://www.flipdogs.com)and so on. "Network and Mobile Data Management Laboratory" led by Professor Meng Xiaofeng in Information Institute of Renmin University of China use recruitment information as an example to develop a "Job Tong" prototype system, aims at study on the recruitment information integration and summarize a set of data integration solutions. At present, the "Job Tong" prototype system has been released on the internet, including recruitment information of 51 JOB, Chinahr.com, ZhiLian ZhaoPin, domestic universities and large work information websites. In the recruitment information retrieval service, as there are characteristics of two-way choice of talented person and the position between job seekers and employers, this paper designs the personalization recommendation into two-way recommended services. Both provide personalized job recommendations for job seekers, and provide personalized talents recommendation for recruiters,

implementation ideas of the two way service are the same. The biggest difference is user interest model content: as for job seekers, position relevant information is recommended; as for recruiters, talents relevant information is recommended.

The construction of personalized service based on Deep Web is divided into two stages: firstly, constructing and updating of user interest model, and then Deep Web data crawling with the constructed interest model. The Fig. 3 shows the construction process.

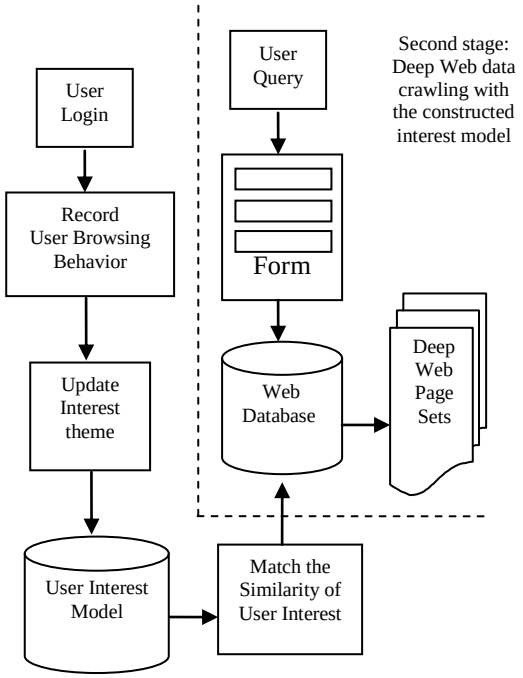

Figure 3. Construction Process of Personalized Service based on Deep Web

III. CONSTRUCTING AND UPDATING OF USER INTEREST MODEL

The key for Deep Web query personalized service system to operate normally lies in user modeling. That is to collect all kinds of users' information, analysis these data, and then create access mode and interest mode that fit the users' characteristics. After that, find a matching mode with the current mode from the user mode library, and then recommend corresponding data sets according to the pattern matching results. At present, the most commonly used expression of the user interest model is the method based on keywords vector space model. This method is widely applicable and intuitive, yet, as for the Deep Web that based on the structured database was not accurate in reflecting the semantic information of user interest.

A. Expression of User interest model

The user interest model represents users' relatively stable interest demand for specific theme information. It can reflect the main tendency of information demand for quite a long time, this kind of information demand stay in the server for a long time. While collecting feedback information from user and tracking the user behaviors, personalized service system will use the machine learning method to adjust and update the current user interest model in order to make it better, all these are in line with the users' real needs.

In practical applications, users and the interests enjoy a many-to-many relation, that means, one user can own many interests, and one interest can be shared by many users. But as for the personalized service system, interests are meaningful rather than users, so information recommendation is based on interests. If one registered for several interests at the same time, the system will set several user models for him. It may be possible that different users registered for the same interest category message, which means they use the same interest template. At the same time, system collects the users' information that registered for the interest, and provides it to all the other users that registered for the interest, so as to let the users who have the same interests communicate and learn with each other.

Users' interests are various and ever-changing, it would be too simply to describe them as interested or uninterested. It can neither effectively describe users' multiple interest features, nor timely tracking users' interests changing, especially some interests update frequently and change shortly. Considering the factors mentioned above, in the period from users submit information needs to log off, do integrated description about one's interests. Thereinto, that includes the process of the user interest model make dynamic update as demand adjusts, in order to achieve the purpose of reflecting user interest information needs timely and accurately. The paper constructs fine-grained management of structured data. The user interest model concludes and summarizes from bottom to up based on the domain ontology, the forming process is simple and the description is accurate. At the same time it can describe the users' interest differently, reduce the interference between different categories, help frequent short-term theme interest changes, and improve the precision of the model updating.

This paper describes the user interest model as five-tuple M: M={S, K, L, W, T}, thereinto, K=(K_u, K_c); S stands for establishes state of the user interest model, K stands for the feature of the corresponding theme. It composed of two parts: the K_u stands for feature model before renewal; K_c stands for feature after dynamically updated. In the initial user model state S_0, there is no feedback updates, so K_c=∅. L stands for K's semantics, described using the recruitment information. Such as position, industry, work experience, etc.; W means the feature weight of K; T means the updating time, which is primarily used to analysis user interests change. For example: M= {S_i, K_i, L_i, W_i, T_i}， S_i is the thematic state of user interest model on the dynamic adjustment renewal. The corresponding feature set K_i, K_i = (K_{iu}, K_{ic}). L= {L_{i1}, L_{i2}, ..., L_{im}}are the semantics of corresponding K_i. W_i,

represents the feature weight of K_i, is a value between [0, 1]. T_i represents the update time of corresponding S_i.

B. Similarity matching of user interest pattern

1）Similarity calculation

After establishing the initial user interest model, similarity and personalization recommendation based on the matching results is what we need to work on. The basic idea is: regarded highly related information as seed, and extended neighbors in mode library using similarity, seek for similar interest information in the user interest library, so as to improve the recall rate. We also tried the statistics and machine learning method to improve the classic similarity so as to get better effect.

According to the established access mode and interest mode in the library, it recommends timely potential but untouched data for each user. Two main tasks were to be finished: Firstly, user identification: analysis the host addresses and request pages and referenced pages URL of each record in the log to identify the users. Secondly, match model and generate recommend sets: search the most similar model for the current user from mode library. Pick recommended sets based on matching results, and delete the data in the recommend sets that current users accessed, and then finally return to the users. In this way, during the query process, the request of browsing behavior, collection operation, query operation, save and print etc. will be passed onto the personalization recommendation module. Personalization recommendation module would recommend sets according to the information for current users, and read from the web server and return to the user.

Definition1. The query Q submitted by users: a Deep Web query was composed of a group of key words, $Q=\{q_i|q_i\in Q, 1<=i<=k\}$, thereinto, Q is the key word set for the users' query.

Definition2. The web database query interface WDBI: it was composed of attribute name, data types and the corresponding candidate value. Definition as follow: $WDBI=\{<A_i,ADT_i,AVA_i>|A_i\in A,ADT_i\in ADT,AVA_i\in AVA,1<i<k\}$. Among them, A stands for the attribute name set of query interface of the Web database; ADT stands for the data type of the corresponding attributes, including TEXT, NUMBER and DATETIME; AVA stands for the candidate value set of corresponding attributes.

Definition3. The given query set Q_1 and the state set S_2 in user interest model, Q_1 and S_2 contains separately m and n data. The weight of each data in Q_1 is respectively: $w_{q1},w_{q2},\ldots w_{qm}$, The weight of each data in S_2 is respectively: $w_{s1},w_{s2},\ldots,w_{sn}$, Match the similarity of data q_i in Q_1 and data s_j in S_2, definition of the similarity measurement is: $Sim(Q_1,S_2)=((\max \sum_{i=1}^{m}\sum_{j=1}^{n} Wqi*Wsj)*a_{ij})*v_{Ai}$ .Among these, the purpose of introducing a_{ij} is to make sure w_{qi} and w_{sj} participate in similarity match for only once. When w_{qi} and w_{sj} matches, $a_{ij} = 1$; Otherwise $a_{ij} = 0$.

Thereinto, v_{Ai} stands for the similarity metric of different attributes caused by different data types in the web database query interface. For TEXT types: we use vector composed of keywords to express, and suppose W_i to be the feature vector of user interest model S_i based on the attribute A_i. W_j stands for the feature vector of data query based on attribute A_i, we use the cosine value of vectors between intersection angles, and formula is:

$$v_{Ai}= (\sum_{k=1}^{n} w_{ik} * w_{jk})/\sqrt{\sum_{k=1}^{n} w_{ik}^2 \sum_{k=1}^{n} w_{jk}^2} \quad (1)$$

About NUMBER and DATETIME types: As for the two kinds of types we can establish precise query .

$$v_{Ai}=\begin{cases} 1 & \text{when query condition is perfectly matched} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

2）Personalized theme recommendation

Generally, when publishing recruitment information, publishers usually were asked to fill in the corresponding information as fields on the web in order to adopt the database means, and information would be displayed in sub area of web page. At this moment recruitment information query can get all the domain information through the analysis that custom-tailored templates made to the web page HTML or XML code. There are some domains possess relative atomic information and fixed formats, such as work site, the minimum education background, work experience, and contact information, place of work. Their values are digital or mode that can be exhausted, they are neither synonymous nor homonymy, when querying, we can use exact match to search or restrict. Let's take work place matching as an example, when the user input desirable working place, only exact match items will be found out. This is a simple Boolean logic. And some other domains are texts consist of natural languages, such as job description, job responsibilities. Having the publisher's individual style, the styles are relatively free and can not be exhausted, synonymous and homonymy and other inherent characteristics of natural language text. These texts are usually very long and often contain very important information. Unfortunately, existing natural language processing technology can not analysis and understand thoroughly, and it is obviously not appropriate to retrieve these domains with the Boolean matching operation [3]. Therefore, the determining of interest theme of user interest model is divided into two categories, the fuzzy one and the precise one. As for this kind of interest theme recommendation, this paper classify users' record by recording users' browse and query, and count users' interest degree, so as to analyzes and get their high interest degree and finally to determine the user interest theme.

Personalization recommendation based on the Deep Web uses the user queries as training set. Through the similarity matching between query data and user interest model, recommend higher interest degree to the users. Methods are as follows:

Input: The query data of user

Output: The result sets of query

Steps:

Step 1: Get query interface data of the Web database.

Step 2: Take an interest theme T_i from the user interest model.

Step 3: According to the similarity calculation formula calculate the similarity between query data and interest theme.

Step 4: Whether the interest model set empty, if not, go to step 2.

Step 5: Find out the maximum similarity, and then return the relevant data sets, end of program.

C. Updating user interest model

In order to enable the user interest model to timely and accurately reflect users' interests, comprehensive consideration is significant in the initial establishing stage, and it needs to dynamically change according to the interest change and browse behavior change, all these needs full use of the user feedback information.

There are two kinds of methods for getting users' interest: the implicit method and explicit method. The so-called implicit method refers to monitor users' behavior on the web page, and record browse number of a page, residence time, the query content of document, the collection operation and the URL address accessed by user to form the log file. The system obtain user interest concept through analyzing the log files, construct user interest model, system [15, 16] used this method. This method would not create any inconvenience for users, but the convergence of user interest model often needs a long time. And the explicit method requires direct participation of users, for example, in literature [14], users offer some of subjects which they are interested in, and the evaluation feedback of the current page, such as providing information when registering as the member of the web site. This method astringe quickly but users' burden was increased, and different users have different understanding about it. To some extent, the feedback accuracy and performance of this method would be reduced. Obviously, combination of the two methods can be much better.

User interest model updating strategy: When the system detects the user interest model status, it will query the theme characteristics under the status, if this model contains these features, it is enough to change its weight value and update time. If the interest is not included in the user model and user interest model has surplus storage space, it will generate a new interest model directly. If storage space is full, we need to eliminate the small weight of interest in the model state to add some new ones.

Feature weight value updating strategy: Suppose D to be the Deep Web data set with additional feedback information, $d_j \in D$, n_j stands for the browsing number of d_j, t_j stands for the browsing time of d_j, W stands for the weight value of feature item K. Weight value updates according to the formula (3) to adjust the system. That is to say according to the time interval between the current system time T_n and the original update time T of feature item and the users' browsing behavior to determine the new weight. At the same time, modify the corresponding update time to T_n.

$$w(T_n)= \frac{\alpha}{\alpha+(T_n-T)} * \sum_{dj\in D}(d_j * t_j) * W_T \quad (3)$$

$$T=T_n \quad (4)$$

The method above reflects users' interest more real and accurate mainly through the machine learning to implicit feedback update. Manual update functions should be provided. When users feel the system can not understand their intentions very well, they can edit interest theme manually.

Ⅳ. DEEP WEB CRAWL

Recruitment website includes two customer types: individual users and the unit users. The unit users often release recruitment information and inquire the talent information, individual users mostly release job seek information and inquire the recruitment information. Personalization recommendation service often built on the identity authentication, and only registered users of the web site enjoy the service, while visitors of the site do not enjoy this service. Thereinto, individual and unit authentication often realized through the username and password. This paper mainly discusses personalization recommendation of the two-way user information query.

When crawling the web database, after submitting query term to the query interface each time, the server will automatically retrieve the back-end database to generate a query result list page that satisfy the query data. And then get detailed pages of each data record from the list page, and these detailed pages in this paper were called Deep Web pages. Each Deep Web page contains a record corresponding to an entity in the real world, and each record is a semi-structured data organized by a number of data elements according to specific models. We call each record as Web data record [20].

The essence of web database crawling is to find a query set $Q = \{q_1, q_2, ..., q_m\}$,and to get the maximum query term coverage when the crawling cost<= constant under the constraint. The definition of query term coverage and the crawling cost of query term are as follows:

Definition1. Given query attribute q_i and web database WDB, the crawling cost of query attributes q_i is $Exp(q_i,WDB)$: $Exp(q_i,WDB)= ND(q_i,WDB)/k$ (5)

Thereinto $ND(q_i,WDB)$ stands for the results sets matched with q_i in the web database, K stands for the biggest record number of each query result page in the target web site.

Definition2. Given query attribute q_i and web database WDB, the coverage of query attribute q_i is $Cov(q_i,WDB)$: $Cov(q_i,WDB)=ND(q_i,WDB)/N_{WDB}$ (6)

Thereinto $ND(q_i,WDB)$ stands for the results sets matched with qi in the web database, N_{WDB} stands for the overall record number in the web data.

When there are query attribute sets $Q= \{q_1,q_2,\ldots q_m\}$ in the query form, the coverage of attribute set $Cov(q_1 \vee q_2 \vee \ldots \vee q_m,WDB)=(ND(q_1,WDB) \cup \ldots \cup ND(q_m,WDB))/ N_{WDB}$. (7)

Thereinto $ND(q_1,WDB) \cup \ldots \cup ND(q_m,WDB)$ stands for the total of the query attribute $q_1,q_2,\ldots q_m$ of all query record of union set in the web database.

In order to find the query sequence and maximize the coverage under the limited number of queries, this paper presents a Deep Web data crawling method based on the tree-structure. This paper considers the web database as data tables of relational model [18]. Consideration is as following: A web database table T: the table is composed of the attributes set $D = \{d_1, d_2, ..., d_n\}$ defined in the result set $AL = \{a_{l1}, a_{l2}, ... a_{lm}\}$; at the same time, there is a group of query attributes set $Q = \{ q_1,q_2,\ldots q_m \}$ in the query form interface. By filling the specified query attribute values in the form item, these queries turn into SQL queries in the underlying database.

Definition3. The web database blocks. According to the equivalent relation theory, a web database table T can be divided into several nonempty subset $\{ T_1, T_2,\ldots T_m \}$, each element in T belongs to a subset of the components T_i. When $T_1 \cup T_2 \cup \ldots \cup T_m = T$ and $T_i \cap T_j = \varnothing$, we say T_i is the block of T. Based on the definition above, a web database is divided into several blocks which are non-intersect, and then each tuple of query results belongs to one particular block.

If the number of result record set D is very large, many queries will only return the first-k records satisfied the query conditions according to certain rules (k is a constant, such as 800 or 1000) [19]. So there are three kinds of return results: When $k< | T |$, overflow, the result record could not be returned wholly; when $| T | = 0$, underflow, the return record is empty; when $0 < | T | < k$, the system return effective result records. If there is overflow, the high coverage could not be guaranteed, and then each query shall be effective value. Thus in the division of web database, we should pay full attention to the effectiveness of queries.

Definition4. Multiway hierarchy tree. The hierarchical tree DT (T) of database table T is the multiway hierarchy tree about query term. Construction is as following: as for a group of query set $Q = \{q_1,q_2,\ldots q_m \}$, a node in the tree represents a query attribute Ai, the edge from the node represents an attribute value. If the attribute value set of attribute A_i is $V = \{ v_1,v_2,\ldots,v_n \}$, then V_i will be called the domain of attributes A_i, and then attribute node A_i has n edges. The layer m + 1 of the tree are the leaf nodes.

According to definition 4, each record in the database belongs to different leaf nodes, and the edge from the root node to leaf node constitutes the structured query of the record extract. On this basis, the Deep Web database crawling problem transformed into trees traverse problem, which is to extract effective record of leaf nodes through the traverse tree. Considering the edge from the root node to leaf node, we use depth-first traversal principle. The specific traverse process is as follows:

Step 1: Start from of any random query q_0 of the query term sequence of the root node; submit to the web database WDB.

Step 2: When the returning results $| T | > k$ (threshold), overflow, we continue to choose a query from the next layer, join it into the query path to make a new query; recycle the process until find a leaf node.

Step 3: If a leaf node was found, continue to use other attribute value of its father node to build the brother node of the traverse leave; otherwise, execute step 2.

Step 4: Judge the validity of the leaves, if it is effective, then extract and preserve the record, otherwise, discard the leaf node.

Ⅴ. EXPERIMENT

In order to give an objectively evaluate to the personalization recommendation methods based on Deep Web proposed in the paper, a prototype is realized according to the theory and algorithm above, and verify a web database in a real. For 15 registered users, each user has five initial interests, record and return data start from the users' first query. For each interest, the maximum return information item number that system setting gets from Deep Web data resource is 30. The following describes the data sets used in the experiment, and then gives the corresponding analysis and experimental results.

A. Deep Web data crawl

In this experiment, we use three Deep Web data sources to carry out Deep Web data crawl: ZhiLian ZhaoPin, ChinaHR, and 51 Job. As for user query q_i, the number of recruitment information item in the three data sources are 762,905 and 689. At the same time, it recorded cost for index page crawling and the information coverage. As in the Table I below:

Table I. EXPERIMENTAL RESULTS OF DEEP WEB DATA CRAWLING

Deep Web Source	Crawling Information Item Number	Cost for Index Page Crawling (%)	Information Coverage (%)
ZhiLian ZhaoPin	767	54	93
ChinaHR	905	70	96
51 Job	689	65	88

Data shows that: The crawling information coverage of three data sources are more than 85%.This shows that the Deep Web data crawling method proposed in this paper is effective under the special environment. From the aspect of cost for index page crawling, the crawling cost is the lowest in the "ZhiLian ZhaoPin", yet, it gains the higher coverage. This is because each Deep Web source is autonomous system completely, and each crawling cost is determined by its own query cost.

B. Personalization recommendation experiment.

According to the recommendation theory mentioned above, we recommended to the above 15 registered users, and it is carried out in two aspects: Real-time recommendation: when submitting the user query q_i, while returning the results record, recommend high similarity information based on the similarity matching;

Regular recommendation: during the fixed appointment time, when the server is idle, it focuses on the calculation, and recommend the new information to the users. In this paper the recall and precision are also regard as standard in personalization recommendation. For 15 registered users, each user has five initial interests, constructs the learning of the same kinds of interests and learning of the different kinds of interests. Supposed the initial interests of 15 registered users are all including "computer", records the learning model of the 15 users with the same interest. The experimental data is shown in Fig. 4. At the same time, the interest of each user will change, records the learning model change of user "user I", and the experimental data is shown in Fig. 5.

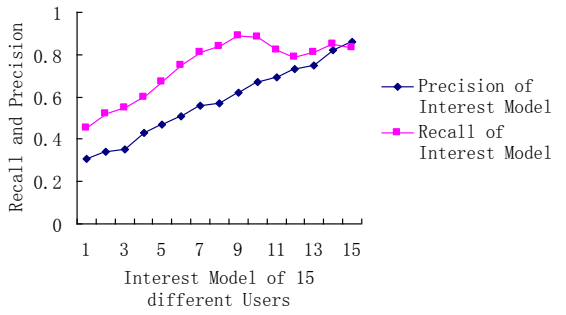

Figure 4.Recall/Precision Comparison between the learning of same kinds of interests

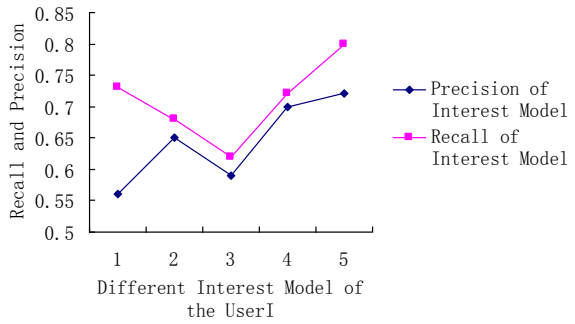

Figure 5.Recall/Precision Comparison between the learning of different kinds of interests

In Fig. 4, learning from the same kinds of interests, with the continuing learning of the user interest model, the precision and recall of recommendation are rising steadily. Therefore, the recommendation algorithm given in this paper can timely update user interest model as the learning progressed. In Fig. 5, as for the interest change of same user, the interest theme shows the tendency of dispersion. This is because the recent interest theme is always used to match in the recommendation algorithm.

VI. Conclusion

With the rapid development of Deep Web, a large number of Deep Web information often lead to "information overload" and "information disorientated ", yet, personalized service can solve this problem. According to the deficiencies of personalization recommendation for different users in the existing recruitment information service, this paper put forward a solution that apply personalization recommended service based on the Deep Web to recruitment services, and that enables the users to obtain better personalized service with few participation.

But, some details in this operation still needs further improvement and discussion in the future: First of all, we set overflow threshold in the Deep Web crawling process, the parameter values were got according to the experience realization, we still need to analysis it in theory; Secondly, as for users' satisfaction and quality of personalization recommendation algorithm still needs more reasonable evaluation methods; Thirdly, we will carry out experiments on more web database, to find out and improve the deficiencies in order to further improve the general recommendation.

References

[1] Chang KCC, He B, Li CK, Patel M, Zhang Z, "Structured databases on the Web: Observations and implications," SIGMOD Record, Vol.33, 2004, pp.61−70.

[2] BrightPlanet.com, "The deep Web: Surfacing hidden value," 2000, http://brightplanet.com.

[3] Wang Jingfan, "Two-Step Job Information Retrieval based on Document Similarity," 2007.

[4] Ma Jun, Song Ling, Han Xiaohui and Yan Po, "Classification of Deep Web Databases Based on the Context of Web Pages," Journal of Software,Vol.19,No.2, pp.267-274,February 2008.

[5] He B, Tao T, Chang KCC, "Clustering structured Web sources: A schema-based, model-differentiation approach," Springer-Verlag. Heraklion, pp. 536−546, 2004 [the 9th Int'l Conf. on Extending Database Technology].

[6] Peng Q, Meng WY, He H, Yu C, "WISE-Cluster: Clustering e-commerce search engines automatically," ACM Press. Washington, pp.104−111, 2004 [the 6th ACM Int'l Workshop Conf. on Web Information and Data Management].

[7] Wu WS, Yu C, Doan AH, Meng WY, "An interactive clustering-based approach to integrating source query interfaces on the deep Web," ACM Press. Paris, pp.95−106, 2004 [the 24th ACM SIGMOD Int'l Conf. on Management of Data].

[8] He H, Meng WY, Yu C, Wu ZH, "WISE-Integrator: An automatic integrator of Web search interfaces for e-commerce," Morgan Kaufmann Publishers. San Fransisco, pp.357−368, 2003 [the 29th Int'l Conf. on Very Large Data Bases].

[9] Zhai YH, Liu B, "Web data extraction based on partial tree alignment," ACM Press. Chiba, pp. 76−85, 2005 [Proc. of the 14th Int'l World Wide Web Conf.].

[10] Zhao HK, Meng WY, Wu ZH, Raghavan V, Yu C, "Fully automatic wrapper generation for search engines," ACM Press. Chiba, pp. 66−75, 2005 [the 14th Int'l World Wide Web Conf.] .

[11] Li Xueqing,Liu Ruihua, "An approach for User Interest Model Based on Ontology," Information Research.

[12] Li Shan, "The Representation and Update for User Profile in Personalized Service," Journal of the China society for scientific and technical information, Vol.29, No.1, 2010, pp.67-71.

[13] Xu Hexiang, "Research on Deep Web Integration and Its Related Several Technologies [Dr. Dissertation]," 2008.

[14] T.Joachims, T.Mitchell, D.Freitag, and R.Armstrong, "Webwatcher: machine learning and hypertext," GI

Fachgruppentreffen Maschinelles Lernen, Universtiy of Dortmund, August 1995.
[15] H.Lieberman, "Letizia: an agent that assists web browsing," 1995 [the 14th International Joint Conf. on Artificial Intelligence].
[16] M.Balabanovic and Y. Shoham, "Learning information retrieval agents: experiments with automated web browsing," Stanford, CA, March 1995 [AAAI Spring Symposium on Information Gathering].
[17] Liu Wei, Meng Xiao-Feng and Meng Wei-Yi, "A Survey of Deep Web Data Integration," Chinese Journal of Computers, Vol.30, No.9, 2007, pp.1475-1489.
[18] Liu Wei, Meng Xiaofeng and Ling Yanyan, "A Gragh-Based Approach for Web Database Sampling," Journal of Software, Vol.19, No.2, 2008.pp.179-193.
[19] Tian Jianwei and Li Shijun, "Retrieving Deep Web Data Based on Hierarchy Tree Model," Journal of Computer Research and Development, Vol.48, No.1, 2011, pp.94-102.
[20] Dong Yongquan, "Research on Key Issues in Deep Web Data Integration [Dr. Dissertation]," 2010.

Tao Tan born in 1981, lecturer. Her current research interests include Internet-based software systems, web crawling and deep web.

Hongjun Chen born in 1979, lecturer. Her current research interests include web application and distributed computing.