The Design of Web Portable Personalization Framework based on Iterative Profiling Algorithm with Time Unit of Weighted Keywords

Shuqing Li *, Enlai Ma
School of Information Engineering, Nanjing University of Finance & Economics, China
*Corresponding author: leeshuqing@gmail.com

***Abstract*—Portable personalization service has been proved very useful in the sharing and reusing of user personalized profiles among different platforms. We present an approach of a Web requesting framework for portable personalization service based on iterative profiling algorithm with time unit of weighted keywords and give a detailed explanation about its ideas and design principles. Some important challenges are also proposed in this paper, which involve how to acquire user personalized information to construct information storage, how to submit user interests to Web servers, and how to update user profiles with an iterative algorithm based on time unit of weighted keywords for discovering main user interests. Finally, we report some related experiments and evaluation of users' satisfaction.**

Keywords—Information recommendation; Personalized profile; Portable personalization

I. INTRODUCTION

In recent years, personalized service has attracted many interests in the research community as a means to decrease ambiguity and return results that are more likely to meet user requirements effectively. We can see that more and more websites begin to use the information of their users' interests to provide personalized service [26]. However, we also notice that there still have been many problems to be addressed. One problem is that current personalized services on the Web can be described as disordered and fragmental in general even if it might be well-organized in one particular website. In fact, different websites always use different interest profiles of their own users and there is no universal and standard user profile which can be shared but one only limited in specific platforms. It is obvious that this duplication of design produces great waste of resources and redundancy of functions. And we can see these dilemmas also exist in some other research fields such as sharing scientific data recourses [3].

The proposal of portable personalization service aims at designing a standard and independent user profile which can be reused and shared in many different websites and applications. Users can download it to their devices on the client-side for supporting local personalization. All of these bring forward a more feasible and potential research area in Web application developing [13]. As opposed to usual Web personalization technology, portable personalization service often requires a general fundamental framework which has some necessary functions such as designing standard user profile, effective means of submitting and retrieving user personalized information and so on. Each Web platform or mobile device can communicate with this framework and share all of user personalized profiles. It constitutes the key content in this paper and we will make some exploration in this research area following our previous research works [12][14][15].

The paper is organized as follows. In the next section, we will discuss some related background work and our motivation. And necessary principles for building the framework for portable personalization service are also described in this section. In Section 3, we discuss the detailed process of designing each component and their connections. The experimental evaluation and results are presented in Section 4. In the final section, we share our conclusions and plan for future work.

II. BACKGROUNDS

The proposal of portable personalization service comes from the requirement of Web applications. In recent years, portable personalization service has attracted many interests in the research community as a means to address the problems in data sharing and reusing, and even help us to find more available user interests which cannot be gotten from only one website. For example, the purchasing information of one user in one shopping website may be so little that it is not enough for us to judge user interests. But the information in many other shopping websites would be useful for providing a more accurate and detailed description of this user. Some researchers call this information as Out of Band Information [6]. More this information is, more precisely we can describe the user interests. And we can see that this is just what portable personalization service refers to.

The complete framework for portable personalization service has three basic components which will be introduced in the remainder of this paper.

A. The Integration of Heterogeneous Personalized Information

Effective integration of heterogeneous personalized information involves two important challenges: acquiring enough information of users' interests, and building universal model suitable to all kinds of user profiles. Although some scholars use groups to synthesize user information as a means to decrease the computational complexity, most researchers have been attempting to utilize independent user profile for

978-1-4673-8783-5/15/$31.00 ©2015 IEEE

providing personalized service to each user [23]. While there are many factors that may contribute to the delineation of the user profiles, here we consider two essential elements that collectively play a critical role in personalized information services.

One is what areas of information we should select. The current information recommendation may refer to a diverse range of applications depending on the nature of the work being performed and always relates tightly with requirement of specific applications such as music recommendation, movie recommendation and so on [5]. Of course, we can construct more general user profiles by mean of integrating these user profiles into higher-level semantic structure. However, different applications always have different requirements and use different data so that the difficulty of information integration and analysis will be too great to overcome for achieving reliable results [10][18]. So it has been proved that it is more feasible to limit data sharing in some specific areas. In this paper, the area of study is mainly about information service in academic documents recommendation, which composes the main field of data sharing in our experiments.

Another is what kind of information we should select. User personalized information will be so diverse that they include short-term interests and long-term interests, or static interests and dynamic interests, et al [20]. All of this information tightly relates with time. We can conclude that static interests and long-term interests often change slowly with user's ongoing behavior, and dynamic interests and short-term interests will have a higher temporal variability [21]. So it is very important to consider the influence of time for knowledge discovery from user profiles and recognition of user main interests. And some scholars combine semantic analysis with time-weighting strategies to improve the expressing effectiveness [11]. More accurate results can be achieved with exploring user interests by weight spreading approach [24]. Some other researchers discover more interesting results of weighting convergence characteristics about user interests based on machine learning [28]. Based on these previous researches, the information we select in this paper includes three parts: semantic information, time information and corresponding weight information.

And we can also see some project teams have attempted to propose a standard for user profiles' sharing such as Data Portability. This project tries to distribute universal user profiles with a self-defined unified format in different platforms. It assigns the information of user profiles as four groups such as User Details, Friend List, Interests, and Updates. And it also gives a clear format definition and explanation of inner components. Since lots of necessary manual interventions are involved, its flexibility is also challenged by the huge diversities of many different platforms [2]. So some other projects want to achieve a balance between flexibility and effectiveness such as APML (Attention Profiling Markup Language) which can allow platforms to define their own user profiles with a standard markup language [8]. But we still have never seen a widely accepted criterion of user profile in portable personalization service by now.

B. *The Design and Constructing of User Profiles*

This function is also called as content management of user profiles which often uses semantic analysis and knowledge discovery technology for describing the main interests of users. Some increasingly popular methods to implement user profiles include vector space model [27] and the utilizing of ontology [24]. No matter which method is used, the semantic analysis always plays an important role in them.

Some researchers adopt three-tiered model of RDF based on SPARQL which is easy to extend, compatible to heterogeneous data, and flexible in structure for designing of storage model of user profile. This model can express user profile with Web service. But some of them also need experts to construct pre-existing domain ontology manually which will be used as the basic storage framework for all user profiles [6]. Meantime, considerable amount of other researches aim at constructing user profile automatically based on this model in traditional commercial areas and they have been proved better than manual ones [17].

The user profiles in servers are not only information storage bank but also the source from which we can abstract more valuable and latent interests with other novel technologies [7]. Some researchers design a personalized system aimed at navigating in buildings with the semantic reasoning technology based on ontology. It can guide users with correct direction automatically, and even can give more reasonable suggestions and adjust the recommendation content based on the relationship of users. Meantime, it also uses the weight spreading approaches in personalized ontology [19].

The method proposed in this paper combines vector space model and ontology. We use vector space model to express the basic user profile, and design a weight spreading approach based on domain ontology constructed automatically for deducing latent user interests.

C. *The Acquisition and Management of User Profiles*

We also call this function as profiles distribution, with which all of supporting platforms can get the particular user profiles directly [25]. In current applications, the common methods of acquiring user profiles include manual submitting and automatic acquisition [20]. And the automatic acquisition methods also include explicit ones and implicit ones [1]. Since each method has its own characteristics, we should take into account the requirement of application and decide which one is better. We adopt the method which submits implicit information automatically by clients. Web request is utilized as the submitting means since it is independent of platform and easy to access [22]. Although many portable personalized systems use mobile clients as storage devices [17], it is hard to manage this information completely and consistently. So the method in this paper uses Web servers as storage center and lets all clients to get user profiles through Web requesting. And it can provide portable personalization services in this way.

Considering the requirements of user privacies, Web request should be validated before submitting and retrieving user profiles [9]. The goal of submitting includes two key aspects. The first one is to submit new user profiles to Web servers in accordance with the predefined formats. The second

978-1-4673-8783-5/15/$31.00 ©2015 IEEE

one is to retrieve user profiles through Web service. We do not need to update or delete existed user profiles. In fact, we can change user profiles in a dynamic and gradient process with this submitting method and weight spreading algorithm mentioned below. And we also believe that it will be hard to acquire accurate user profiles with operations of arbitrary deletion and direct modification. Of course, users can also decide whether they want to share their profiles and their interests from the servers in our system.

III. The Introduction of Framework

The whole framework is illustrated in Fig. 1:

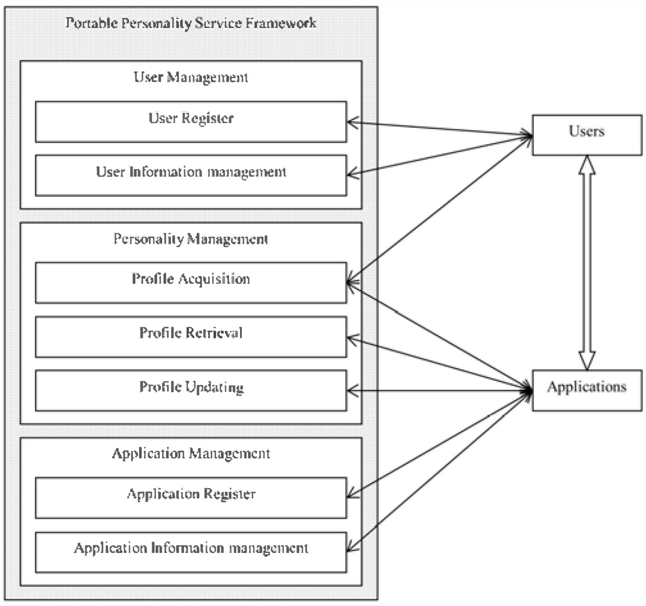

Fig. 1. The whole framework

A. Selecting a Template (Heading 2)

The framework proposed in this paper has three main components which include user management model, application management model, and profile management model.

The main function of user management model is to assign unique identifier to each user with which application can retrieve corresponding profile of particular user. Application management model also provides some similar functions such as assigning unique identifier but only for applications not for users. With these application IDs, applications can sign in Web servers and gain the authority of updating and retrieving user profiles. The most important model is profile management model which can update and retrieve user profiles. The available strategy of updating and retrieving user profiles may vary depending on selection of data and algorithm. We will discuss our method in the next section. The most essential point is that each application within this framework should adopt consistent strategy of updating and retrieving user profiles. Meantime, this model can also discover more valuable information from the original user interests based on whole data set. But we should notice that how to use these user profiles and how to apply them into each application is not the necessary task of profile management model. And all other clients should decide how to design some interfaces to communicate with profile management model and utilize retrieved user profiles to get useful results by themselves [25]. This paper will discuss the two most important parts in profile management model which include profile acquisition, and profile updating in detail.

The concern of privacy has also been paid greatly in this open framework. Both the requesting of user profiles and updating user profiles need Web authentication. Every application can access the information of user profile based on its permission granted by administrator of this framework. That is to say, some applications maybe only retrieve user profiles but others can update user profiles. As for users, they also have own freedom to select the level of presenting their profiles. If some users do not want to share their profiles, they can shield their profile when logging in framework.

B. Profile Acquisition

1) Storage Model

We utilize the RDF as data format since RDF is based on XML data model, and meets the requirement of the big data in Web, and has more advantages of data updating and concurrency controlling compared with traditional relational data model [4][16]. RDF format usually has two basic elements: property name and property value. And it is easy for applications to distinguish these values according to their property names. We can also depict RDF format with weighted XML model which nodes' weight can be used for expressing the degree of user interests.

2) The Selection of Information

As mentioned before, this paper uses semantic information, time information and weight information together. The semantic information can be abstracted from keywords which are stored in applications and can reflect user interests. We can get time information in two ways. One is by log records from which we can know when applications submit user profiles. Another is directly by the parameters of Web requests sent by applications. And the weight information is the sequential order of keywords which can express the importance of user interests. We do not use the original weights generated by each application, which we think could not be suitable for other applications since different applications always have different algorithms and there will be meaningless for sharing these original weights. But the sequential order has better universality which can eliminate the difference caused by different algorithms and can also express the degree of user interests. The basic weighted XML model for a user profile is shown as:

```
<?xml version="1.0" encoding="UTF-8"?>
<UserProfile>
 <User userID="1001">
  <Application appID="118.123.20.105">
   <Field value="research">
    <Keyword value="ontology">
     <Time>2012</Time>
```

```
      <Weight>5</Weight>
    </Keyword>
    <Keyword value="information retrieval">
      <Time>2012</Time>
      <Weight>4</Weight>
    </Keyword>
  </Field>
  <Field value="address">
    <Keyword value="buffalo">
      <Time>2013</Time>
      <Weight>5</Weight>
    </Keyword>
  </Field>
 </Application>
</User>
</UserProfile>
```

The node UserProfile denotes the root node and has many sub-nodes User. Each User node has many sub-nodes Application and each node Application has many sub-nodes Field which have property names defined by applications. These property names also have labels of identifiers with which other applications can get corresponding user interests. Each node Field has many sub-nodes Keyword. Each Keyword node has two sub-nodes: Time and Weight. We use year as time unit and limit the domain of weight within 5 to 1 which depicts the degree of relevance from the highest to the lowest. The algorithm mentioned below will process these raw discrete values into continuous ones.

Finally, with aggregating all the property values according to property names, we will get data collection of each property value which includes semantic information, time information and weight. Based on this data, we can discover the latent user interests or get more accurate presentation of user profiles.

The same properties in different user profiles can construct a property domain with which we can integrate and synthesize the characteristics of all user interests. Based on this, each user could be analyzed in each property domain.

3) Profile Submitting

We can use Web request to submit user profiles to Web server. The main information we need includes application ID, user ID, time, property name, and property value such as keyword. The standard request looks like these:

http://ServerIP/SetInterests?userid=1001&time=2012&label=research&keyword=ontology&keyword=information%20retrieval

http://ServerIP/SetInterests?userid=1001&time=2011&label=address&keyword=buffalo

These two Web requests can be used for expressing the user profile mentioned above. We can add the ordered keywords in the parameter list of query string. In order to limit traffic load of network, the prototype system only uses the first 5 keywords in query string.

All the information submitted will be aggregated based on user ID, application ID, property and time. We can call this result as aggregated information unit. Each aggregated information unit will be expressed in vector space model. For example, the aggregated information units in the user profile mentioned will be:

<1001, 118.123.20.105, research, 2012, <ontology, 5>, <information retrieval, 4> >

<1001, 118.123.20.105, address, 2011, <buffalo, 5> >

If there are many same information units, we can aggregate these information units into one unit in which the average weight is assigned as final weight for same keywords.

C. Profile Updating

This algorithm has two steps. One is to measure the relation of keywords in domain ontology. And the other is to calculate the weights of all keywords.

As for keyword relation, traditional methods are often based on extended TF/IDF algorithms such as Directional Affinity (DAff). The traditional DAff method only concerns the document frequency and ignores the effectiveness of each keyword. For example, we cannot distinguish two keyword pairs which co-occurring term frequency of one pair in one document is only 1 but another pairs is 10. Their document frequency will be same to calculate their DAff value since all are treated as 1. The new method we propose here combines the measuring of keyword weight based on DAff, and replaces document frequency with keyword weight, which is shown as:

$$relation_{keywordi,keywordj} = \frac{\sum_{dock}(weight_{keywordi,dock} \times weight_{keywordj,dock})}{\sum_{dock}(weight_{keywordi,dock})}$$

This method can measure relation of keywords better through integrating the document frequency and keyword weight. And this weight is asymmetric so that we have to calculate the co-occurring weight of both (A, B) and (B, A). And we find that this method has more advantages in recognizing similar semantics and synonyms. With these keywords and their weighted relation, we can generate the domain ontology for semantic analysis mentioned below.

This algorithm can spread keyword weights repeatedly and iteratively within different keywords in different time points. It also considers the temporary decaying effect so that we call it as iterative algorithm based on time unit of weighted keywords. The pseudo code is listed below:

```
Input: user ID (userID), property name (field)
Output: the vector of keywords and their weights ordered by descended weight

// Collect all the information of specific user and property submitted by
applications, this information also includes time, keywords and weight of
sequential order.
Collection collection=getAllInfoOfUser(userID, field);

// Aggregate all the data grouped by keywords and time and use average sequential
order as the final weights.
getAvgWeightByValueAndTime(collection);
```

```
// Get all the data in the previous time period such as keywords, time and weights.
Collection preInterests=getTimeInfoOfUser(userID, field, timei-1);

// Get all the time of specific user in ascended order
for each time timei of collection in ascending order {
// Get all the information of specific user in current time period such as keywords, time and weights.
Collection curInterests=getTimeInfoOfUser(userID, field, timei);

// Calculate the weight iteratively with spreading activation.
// The convergence can be achieved with at most 10 times.
for(int j=0;j<10;j++){
    // Normalize the weight on dividing all the weight with maximum weight.
  normalize(preInterests);
  normalize(curInterests);

  // Get all keywords and their weights in previous time period
  for each keyword keywordk, weight weightk of preInterests {
  // Get other related keywords and their weights of current keyword with domain ontology
  Collection relatedKeywords=getRelatedKeywordsByOntology(keywordk);

  // Update the weights of all keywords in current time period with Formula:
```

$$weightInCurInterests_{keywordi} = weightInCurInterests_{keywordi} + \sum_j weightIn\Pr eInterests_{keywordj} \times relation_{keywordi,keywordj}$$

```
        updateCurInterests(relatedKeywords);
  }

  // Get all keywords and their weights in current time period
  for each keyword keywordk, weight weightk of curInterests {
  // Get other related keywords and their weights of current keyword with domain ontology
  Collection relatedKeywords=getRelatedKeywordsByOntology(keywordk);
  // Update the weights of all keywords in period time period with Formula
```

$$weightIn\Pr eInterests_{keywordi} = weightIn\Pr eInterests_{keywordi} + \sum_j weightInCurInterests_{keywordj} \times relation_{keywordi,keywordj}$$

```
        updatePreInterests(relatedKeywords);
  }

  // Merge all keywords of current time period into keywords of previous time period with the weight's decaying coefficient 0.5 and Formula 4
  weight=weight×0.5(timei - timei-1)
  // Use average weights as final weights for same keywords
  updatePreInterests(preInterests, curInterests);
  }
}
  // Normalize the weights of all keywords
  normalize(PreInterests);
```

IV. Experiments

We have collected 28848 academic articles in 19 journals from Elsevier and JASIST databases. We want to test the prototype of this framework in some specific domains so that the data acquired has controllable size. All of these journals are about information and library science and the time span is about 60 years from 1950 to 2013. The available components in documents include title, abstract and keywords list.

A. *The Experiment of Discovering Main Interests*

We use keywords to express interests of authors. But the number of keywords is usually limited to 3 or 4 in each paper. So the keywords list is needed to extend before calculation. We assume that research work of one scholar is generally focused on a certain area. So we mark not only the occurrence and frequency information of the terms in the keyword list of the article, but also terms in keyword lists of the other articles of this author. We limit our analysis in three fields such as title, abstract and keywords list.

In order to verify the validation of experiment, we choose an author and his/her articles, and evaluate the corresponding results. The articles and their keywords of this author are listed in Table 1:

TABLE I. The Articles and Their Keywords of an Autho

DocID	Year	Keywords
16242	2005	Web searching/Session duration/Query language/Search engine evaluation
14777	2005	Automated assistance/Intelligent information retrieval systems/Explanation systems/Contextual help/Adaptive interfaces/Implicit feedback
14955	2006	Web search engine/Overlap/Google/Yahoo/MSN Search/Ask Jeeves/Dogpile/Infospace Inc
14855	2006	Web search engines/Web searching/Transaction log analysis
14923	2006	Web searching/Web search engines/Web search engine evaluation/Ecommerce searching/Paid searching/Sponsored results/Organic results/Non sponsored links
15200	2008	Collaborative information behavior model/Collaborative information behavior/Healthcare teams/Healthcare information behavior
15298	2008	User intent/Web queries/Web searching/Search engines
15379	2009	ARIMA/Box Jenkins model/Search engine/Time series analysis/Transactional log
15423	2009	Web searching/Information searching/To Anderson and Krathwohl's taxonomy/Bloom's taxonomy
27883	2009	Twitter/social networking/sentiment analysis/classification/marketing
14144	2011	Real time search/Real time content/Collecta/Twitter/Economic value of search/Search topics
14317	2013	Sponsored search/Keyword advertising/Pay per click/PPC/Online advertising/Search engine marketing/Gender targeting/Demographic profiling

After the extension of keywords and submission to server, we can see the final result of this user's interest. The data is listed in Table 2. We can see that they have some extended keywords in year 2000 which are not in Table 1. And the final weight of each keyword is also calculated by average sequential number.

We can see the final keywords of interests and their weights listed in Table 3 with the vibrating algorithm of time unit mentioned before.

TABLE II. THE ARTICLES AND THEIR KEYWORDS OF AN AUTHO

Year	Keyword	Weight	Year	Keyword	Weight	Year	Keyword	Weight
2000	queries	5.0	2006	search	4.5	2009	time series analysis	4.0
2000	user	4.0	2006	log analysis	4.0	2010	information	5.0
2000	analysis	3.0	2006	web	3.29	2010	information searching	4.0
2000	users	2.0	2007	query	5.0	2010	retrieval	3.0
2000	web	1.0	2007	metasearch	5.0	2010	fields	2.0
2001	public	5.0	2007	business	5.0	2010	searching	1.0
2001	searching	4.5	2007	web	4.0	2011	search	5.0
2001	web	3.5	2007	dogpile	3.0	2011	time	4.0
2001	web searching	3.0	2008	queries	5.0	2011	real time search	3.0
2001	research	2.0	2008	behavior	5.0	2011	queries	2.0
2005	altavista	5.0	2008	information	4.0	2011	economic value	1.0
2005	assistance	5.0	2008	web queries	4.0	2013	sponsored search	5.0
2005	web	4.5	2008	web	3.0	2013	gender	4.0
2005	searching	3.33	2009	multimedia	5.0	2013	advertising	3.0
2005	search	3.0	2009	analysis	5.0	2013	search	2.0
2006	ecommerce	5.0	2009	query	5.0	2013	performance	1.0
2006	analysis	5.0	2009	twitter	5.0			

TABLE III. THE TOP 10 KEYWORDS OF INTERESTS IN THE WEIGHT-DESCENDING ORDER

Keyword	Weight
search	1.0000
information	0.2514
performance	0.2100
web	0.1967
retrieval	0.1480
sponsored search	0.1393
research	0.1391
gender	0.1307
user	0.1166
advertising	0.1117

From these data, it can be concluded that the researches of this author are mainly focused in Search and its related research areas. In fact, this research interest always has greater weights in all time points. Even if there are many related terms with different spelling such as Searching, Query and et al., this algorithm also can recognize the main interests and return a synthesized result. And some new interests also gain more weights than other interests since this algorithm is influenced by time decaying.

B. *The User Evaluation of Experiment*

We develop a test prototype system for user evaluation. In this system, each user is asked to choose one author whose research area he/she is familiar to. And all of the users are also required to measure their satisfaction with pilot evaluation. The satisfaction includes two aspects. One is whether these terms can reflect main interests of selected author more effectively. The other is whether these weights and the order of terms are right and satisfactory. The total number of users is 20. And the interests list of each author has 10 keywords with highest weights.

The rate of their satisfaction uses a 5 point Likert scale in which 5 is the most satisfied and 1 is not satisfied at all. The total average score of satisfaction is 3.68 and the total average score in first 5 lines is 4.24, which is 41.3% higher than normal average number 3. And we can see less lines are calculated, higher the average score is.

We have validated the effectiveness of these evaluation results with NDCG. The aim is to validate whether these users have higher consistence to give these evaluations. It will be better when the first record has the highest score and the last one has the lowest score. NDCG can measure how this method in our paper can rank the results effectively. The mean of all NDCG result is 0.938 shown in Fig. 2:

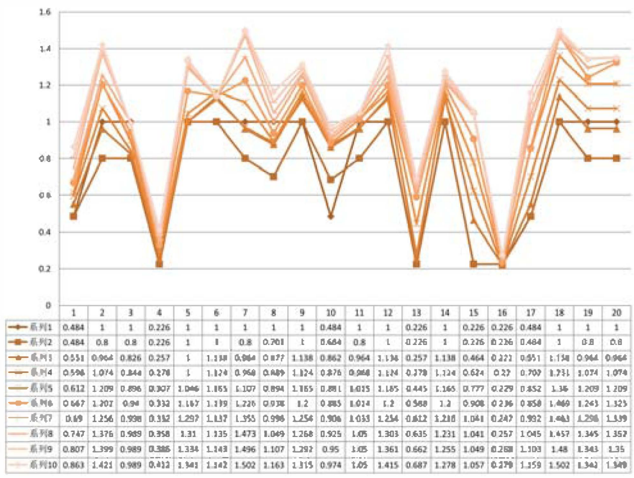

Fig. 2. The evaluation results of NDCG

We also use this portable personalization system to implement a recommendation system of advertisement. This system use the profiles from the Web server to decide what types of advertisement should be pushed down to mobile users. The request method is implemented in Web Service. Since Web service is based on SOAP protocol, it is suitable for expressing the weighted XML model. Other applications can also access these user interests in XML format which include keywords and their weights. In order to reduce the data transfer, this portable personalization system only uses the top 5 keywords with highest weights. The Web request format of retrieving user profile is showed as:

http://ServerIP/GetInterests?userid=1001&label=research.

V. Conclusions and Outlook

In our future work, we plan to continue evaluating the stability and expand our data collection in other areas for conducting a wider and more formal user evaluation. Since we focus on the design of user profile in Web servers, this profile needs to adapt according to the different requirement of each client. Since effective results lie on the coordination between servers and clients, our future work will involve designing experiments that will allow us to monitor the utilizing of user profiles over time to ensure the incremental updates to the interest weights more accurately and reflect changes in user interests for fitting the need of applications better.

VI. Acknowledgments

This work has been supported by Social Science Foundation of Jiangsu Province 2014SJB144 (2014), and Chinese National Natural Science Foundation 71103081 (2011).

References

[1] Amatriain, X. (2013). Mining large streams of user data for personalized recommendations. ACM SIGKDD Explorations Newsletter, 14(2), 37-48.

[2] Becker, A., Mladenowa, A., Kryvinska, N., & Strauss, C. (2012). Evolving Taxonomy of Business Models for Mobile Service Delivery Platform. Procedia Computer Science, 4(10), 650-657.

[3] Borgman, C. L. (2012). The conundrum of sharing research data. Journal of the American Society for Information Science and Technology, 63(6), 1059-1078.

[4] Bruns, V., Reymann, S., & Lugmayr, A. (2007, December). Distributed profiling in a partitioned ambient network. In Proceedings of the 6th international conference on Mobile and ubiquitous multimedia (pp. 34-41). ACM.

[5] Chen, T., Han, W. L., Wang, H. D., Zhou, Y. X., Xu, B., & Zang, B. Y. (2007, August). Content recommendation system based on private dynamic user profile. In Machine Learning and Cybernetics, 2007 International Conference on (Vol. 4, pp. 2112-2118). IEEE.

[6] Ghosh, R., & Dekhil, M. (2008, April). Mashups for semantic user profiles. In Proceedings of the 17th international conference on World Wide Web (pp. 1229-1230). ACM.

[7] Han, L., & Chen, G. (2009). A fuzzy clustering method of construction of ontology-based user profiles. Advances in Engineering Software, 40(7), 535-540.

[8] Hussny, H. A., Mateen, A., Nayyer, M. M., & Mustafa, T. (2013). E-Learner's Collective Intelligent System Framework: Web Mining for Personalization in E-Learning 2. 0 Ecosystem using Web 2. 0 Technologies. International Journal of Computer Applications, 66(4): 1-9.

[9] Kobsa, A. (2007). Privacy-enhanced personalization. Communications of the ACM, 50(8), 24-33.

[10] Kuflik, T., Kay, J., & Kummerfeld, B. (2012). Challenges and solutions of ubiquitous user modeling. In Ubiquitous Display Environments (pp. 7-30). Springer Berlin Heidelberg.

[11] Kwon, O., & Kim, J. (2009). Concept lattices for visualizing and generating user profiles for context-aware service recommendations. Expert Systems with Applications, 36(2), 1893-1902.

[12] Li, S. Q., & Liu, X. Q.. (2012). The Matching Algorithm of Heterogeneous User Personalized Profile Based on Centripetal Spreading Weighted XML Model. New Technology of Library and Information Service, 28(5), 32-40.

[13] Li, S. Q., & Yang, X. N.. (2010). Construction Method of the Portable Personalized User Profile Based on Multi-dimensional Ontology. Information Studies: Theory & Application, 33(11): 116-121.

[14] Li, S. Q., Xu, X., Qian, G., & Han, W.. (2012). A Method for Automatic Recognition and Visualization of Main-paths in Academic Documents based on Vibration Algorithm and Domain Ontology. Journal of the China Society for Scientific and Technical Information, 31(7), 676-685.

[15] Li, S. Q.. (2012). Research on automatic construction of domain ontology in library and information science based on weighted co-occurrence of citation keywords. Journal of the China Society for Scientific and Technical Information, 31(4), 371-380.

[16] Liu, W., Xia, C. J., & Zhang, C. J.. (2013). Big Data and Linked Data: The Emerging Data Technology for the Future of Librarianship. New Technology of Library and Information Service, 29(4), 2-9.

[17] Lugmayr, A., Reymann, S., Bruns, V., Rachwalski, J., & Kemper, S. (2009). Distributing the personal digital environment throughout your entertainment environment: handling personal metadata across domains. Multimedia systems, 15(3), 187-199.

[18] Nakatsuji, M., Yoshida, M., & Ishida, T. (2009). Detecting innovative topics based on user-interest ontology. Web Semantics: Science, Services and Agents on the World Wide Web, 7(2), 107-120.

[19] Niu, W. T., & Kay, J. (2010). PERSONAF: framework for personalised ontological reasoning in pervasive computing. User Modeling and User-Adapted Interaction, 20(1), 1-40.

[20] Papadogiorgaki, M., Papastathis, V., Nidelkou, E., Waddington, S., Bratu, B., Ribiere, M., & Kompatsiaris, I. (2008). Two-level automatic adaptation of a distributed user profile for personalized news content delivery. International Journal of Digital Multimedia Broadcasting, 2(10):1-21.

[21] Park, Y. J., & Chang, K. N. (2009). Individual and group behavior-based customer profile model for personalized product recommendation. Expert Systems with Applications, 36(2), 1932-1939.

[22] Rospocher, M., & Serafini, L. (2013). An Ontological Framework for Decision Support. In Semantic Technology (pp. 239-254). Springer Berlin Heidelberg.

[23] Shtykh, R. Y., & Jin, Q. (2009, June). Integrating Search and Sharing: User-Centric Collaborative Information Seeking. In Computer and Information Science, 2009. ICIS 2009. Eighth IEEE/ACIS International Conference on (pp. 388-393). IEEE.

[24] Sieg, A., Mobasher, B., & Burke, R. (2007, November). Web search personalization with ontological user profiles. In Proceedings of the sixteenth ACM conference on Conference on information and knowledge management (pp. 525-534). ACM.

[25] Uhlmann, S., & Lugmayr, A. (2008, October). Personalization algorithms for portable personality. In Proceedings of the 12th international conference on Entertainment and media in the ubiquitous era (pp. 117-121). ACM.

[26] Wang, L. S., Zhang, S. W., & Lin, H. F.. (2012). An Information Recommendation Model Based on Item and Tag Random Walk Model. Journal of the China Society for Scientific and Technical Information, 31(3), 289-296.

[27] Zhou, Y. Q., Hu, Y. F., & He, H. C. (2007, August). Learning user profile in the personalization news service. In Natural Language Processing and Knowledge Engineering, 2007. NLP-KE 2007. International Conference on (pp. 485-490). IEEE.

[28] Zhuhadar, L., Nasraoui, O., & Wyatt, R. (2009). Dual Representation of the Semantic User Profile for Personalized Web Search in an Evolving Domain. In AAAI Spring Symposium: Social Semantic Web: Where Web 2.0 Meets Web 3.0 (p. 84).

978-1-4673-8783-5/15/$31.00 ©2015 IEEE