EXPLORATIONS ON ENGINEERING LETTERS (EEL)
VOLUME 1, ISSUE 1 (2016):PP.258-264
SANA ACADEMIC PRESS

Recommending Distinct Web Services by Functional and Non Functional Evaluation Using Clustering Methods

P.Jayalakshmi [1], N.Muthulakshmi[2], R. Latha[3]
[1]*Research Scholar, St.Peter's University, Chennai.*
[2]*Asst.Prof. & Head., Dept. of Computer Applications, St.Peter's University, Chennai*
[3]*Prof. & Head., Dept. of Computer Science & Applications, St. Peter's University, Chennai*
lakshmikaranp@gmail.com

***Abstract**: In RDWS by functional and non- functional evaluation using Clustering method I present a personalized-recommendation system that makes use of representations of items and user-profiles based on Diversifying web based recommendation. The diversifying process is followed up by hierarchical clustering the history, potential user interest here I am implementing the method which gives a best clustering result to get a distinct web service. The System analyses User functional interest and Potential user interest. Based on these two studies system will group the relevant information in a graphical structure. Grouping the relevant information first calculates the functional similarity between service candidates and then constructs a Web service graph with the computed similarity values between service candidates. Finally, the system will order the information by top-k ranking algorithm. To developed and implemented personalized-recommendation system that makes use of representations of items and user-profiles. Based on ontologies in order to provide semantic applications with personalized services. Ontologies: A structure of concepts or entities within a Domain, organized by relationship; A System Model.*

***Keywords**: RDWS, Ontologies, Web service recommendation, diversity, user interest, QoS preference, service usage history*

I. INTRODUCTION

Web services have been fastest developed in recent years and playing an increasingly significant role in all fields in the world like e-commerce, enterprise application integration, and other applications. With the growth of the N number of Web services on the Internet, Finding the distinct Web services has become a critical issue to be addressed in service computing community [1].Since there are many Web services launched with similar functionalities and different non-functional quality in past, it is important for users to select desirable distinct and high-quality Web services which satisfy both users' functional and non-functional requirements. Recently, suggesting qualified and preferred Web services to users has attracted much attention in terms of the information oppress problem. Web service recommendation is a process of essentially unearth and recommending suitable Web services to end users. A number of works have been done on service recommendation based on quality of service (QoS). Most of them employed Collaborative Filtering (CF) techniques [2-6], some of them applied content-based approach, and a few of them combined CF approach with content-based techniques. They focus on predicting missing QoS values of Web services used by similar users for an active user. However, there are drawbacks for these approaches. To begin with, they simply recommend users Web services with the best QoS values on a certain QoS criterion with-out exploiting the user's potential QoS preferences, which may likely be mined from his/her service usage history [7]. A user's QoS preference for services is certainly important for real service recommendation scenarios, since it can be used for measuring the QoS utility of a Web service in a more accurate and personalized way. Moreover, existing service recommendation approaches may have unneeded similar services in the top-k recommendation lists, since there is a default assumption that all the k results are independent of each other, which may not be true in many times. As a result, the user's satisfaction degree may decrease in their experience of selection in the recommended list due to the redundant services in the limited top-k recommendation list. For example, suppose there is a certain category of services with similar or related function (i.e., in the same service domain) which match a user's interests and have comparatively higher QoS than the services in other categories. It is probable that existing service recommendation approaches will only recommend services in this category to the user in the final short recommendation list. From the user's viewpoint, however, the recommended services with similar functionality are redundant, and services in the other categories which are interesting to the active user should also be incorporated as many as possible in the only limited top-k recommendation list. In order to remove the redundancy in service recommendation list, and at the same time maintain the quality of the recommended services, diversity should be considered in recommendation. In recommender systems, when the k best recommendations are very similar to each other, many of them may be useless to the user, and thus the usefulness of k recommendations may be very low. It is desirable for a recommender system to return a diverse set of cases in or-der to provide the user with optimal coverage of the in-formation space [8]. Currently, diversity is considered as important as similarity in many existing recommender systems [9-10]. For example, Zhou et al. [9] discussed the diversity-accuracy dilemma of recommender systems, showing that hybrid method with diversity can improve the recommendation performance. Ziegler et al. [10] proposed that recommendation can be improved through topic diversification. Based on these facts, we argue that diversity is also an important feature in Web service recommendation systems. In this paper, we propose a novel service recommendation approach by taking diversity into consideration. We incorporate the functional relevance, QoS utility, and diversity features of Web services for recommending well diversified top-k services to users. Specifically, the contributions are as follows;

1) We mine a user's functional interests and QoS preferences by exploring his/her service usage history. The user interests are two-fold: the historical user interest and the potential user interest. The historical user interest is mined through its own service usage history, query logs and profile, while the potential user interest is derived through collaborative filtering approach. User interests and QoS preferences are used for measuring the functional relevance and QoS utility respectively for Web service candidates.
2) We compute a score for each Web service candidate using the functional relevance and QoS utility. Mean-while, we construct a Web service graph based on the functional similarity between service candidates with a certain level of user interest relevance. Here the relevance can be done using the hierarchical clustering method. The inputs are service candidates.A diversity measure is defined based on the Web service graph.
3) We perform a novel diversity-aware service ranking algorithm to find the optimal top-k Web services based on a proposed comprehensive ranking measure. The experimental results indicate that the proposed approach improves the performance of service recommendation compared with the existing methods.

I. A Clustering Principles

Our approach is based on two criteria: one is on the queries themselves, and the other on user clicks.
The first criterion is similar to those used in traditional approaches to document clustering methods based on keywords. We formulate it as the following principle:
Principle 1 (using query contents):
If two queries contain the same or similar terms, they denote the same or similar information need.Obviously, the longer the queries, the more reliable the principle 1 is. However, users often submit short queries to search engines. A typical query on the web usually contains one or two words. In many cases, there is not enough information to deduce users' information needs correctly. Therefore, the second criterion is used as a complement. The second criterion is similar to the intuition underlying document clustering in IR. Classically, it is believed that closely associated documents tend to correspond to the same query . In our case, we use the intuition in the reverse way as follows: Principle 2 (using document clicks):
If two queries lead to the selection of the same document (which we call a document click), then they are similar. Document clicks are comparable to user relevance feedback in a traditional IR environment, except that document clicks denote implicit and not always valid relevance judgments. The advantages are as follow efficient result from a query. Information gathered from domain Based on Users interest.High Accuracy & QOS.

II. LITERATURE SURVEY

Bringing Order to the Web: Automatically Categorizing Search Results Hao Chen - School of Information Management & Systems University of California. This model was then used to classify new web pages returned from search engines on-the-fly. This approach has the advantage of leveraging known and consistent category information to assist the user in quickly focusing in on task-relevant information. The interface allows users to browse and manipulate categories, and to view documents in the context of the category structure. Automatic Identification of User Goals in Web Search Uichin LeeUniversity of California.In this paper we study whether and how we can automate this goal-identification process. We first present our results from a human subject study that strongly indicate the feasibility of automatic query-goal identification. Query Recommendation using Query Logs in Search Engines - Ricardo Baeza-Yates1, Carlos Hurtadol. In this paper we propose a method that, given a query submitted to a search engine, suggests a list of related queries. The related queries are based in previously issued queries, and can be issued by the user to the search engine to tune or redirect the search process. Varying Approaches to Topical Web Query Classification.Steven M. Beitzel - Telcordia Technologies, Inc. One Telcordia Drive We have evaluated three differing approaches to topical web query classification. We find that training explicitly from classified queries outperforms bridging a document taxonomy for training by as much as 48% in F1. Context-Aware Query Suggestion by Mining Click-Throughand Session Data - Huanhuan Cao1 Daxin Jiang2. In this paper, we propose a novel context-aware query suggestion approach which is in two steps. In the *o²ine model-learning step*, to address data sparseness, queries are summarized into concepts by clustering a click-through bipartite**.**

III. FRAME WORK OF OUR APPROACH

Here we discuss the Structure of the service recommendation intimation which takes clustering into consideration as shown in Figure 1. In this Structure, Recommendation with Diversity using clustering is the key component. Content-based similarity is acquired by text similarity. This work considers Web service. These are explained by the Web Service Description Language (WSDL).

The user's historical interest must be mined from his/her own query history or service usage. Functional Evaluation predicts the user's potential interest and evaluates its similarities with Web services by implementing collaborative filtering based user similarity. The user similarity is measured depends on the service invocation history of all service users and apply clustering..

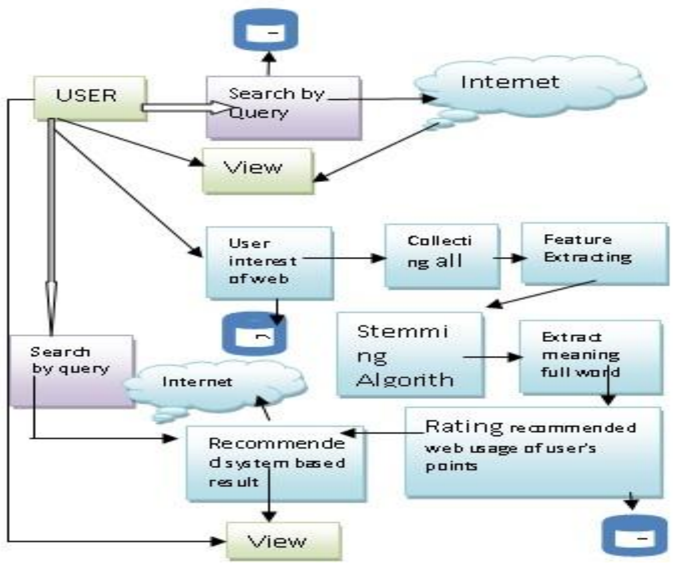

Fig. 1 : Structure of Service Recommendation Approach

Non-functional Evaluation first infers the user's potential QoS preference on a service candidate through mining the service's usage history, then compute the QoS utility of the Web service with the QoS information. Diversity calculation first take the functional relevance between service candidates, and then constructs a Web service graph with the calculated similarity values between service candidates.The components of the approach is Classified into following manner 1. Creating Search History 2. Query Clustering 3. Query Reformulation 4. History Grouping. History grouping deals with the clustering technique.

A. Creating Search History

Any personal documents such as browsing history and emails on a user's computer could be the data source for user profiles. This focus on frequent terms limits the dimensionality of the document set, which further provides a clear description of users' interest. This module allows the search engine to better understand a user's session and potentially tailor that user's search experience according to her needs. Once query groups have been identified, search engines can have a good representation of the search context behind the current query using queries and clicks in the corresponding query group.

B. Query Clustering

User's queries can be classified into different query clusters. Concept-based user profiles are employed in the clustering process to achieve personalization effect. The most similar pair of concept nodes, and then, merge the most similar pair of query nodes, and so on. Each individual query submitted by each user is treated as an individual node and each query with a user identifier. we perform the grouping in a similar dynamic fashion, whereby we first place the current query and clicks into a query group.

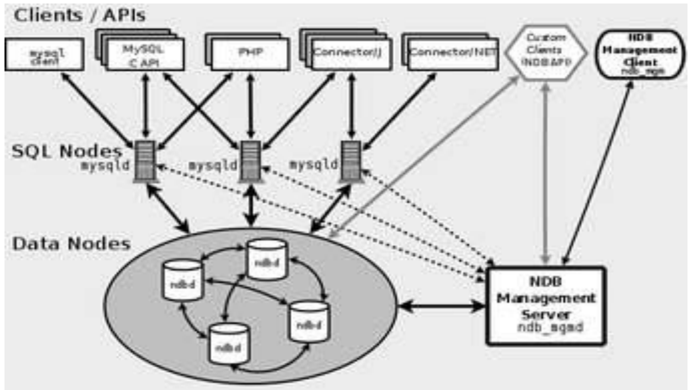

Fig. 2: clustering mechanism is shown in the fig

C. Query Reformulation

To ensure that each query group contains closely related and relevant queries and clicks, it is important to have a suitable relevance between the current query groups. We assume that users generally issue very similar queries and clicks within a short period of time. The search history of a large number of users contains signals regarding query relevance, such as which queries tend to be issued closely together. This captures the relationship between queries frequently leading to clicks on similar URLs. Query reformulation graph and the query click graph from search logs, and how to use them to determine relevance between queries or query groups within a user's history.

D. History Grouping

Query groups is to first treat every query in a user's history as a query group, and then merge these query groups in an iterative fashion (in a hierarchal clustering).However, this is impractical in our scenario for two reasons. First, it may have the undesirable effect of changing a user's existing query groups, potentially undoing the user's own manual efforts in organizing her history. Second, it involves a high-computational cost, since we would have to repeat a large number of query group similarity computations for every new query.

IV PERFORMANCE EVALUATION

In this section, we conduct experiments to study the effects of parameters in our approach. Specifically, we study the impacts of α, β, γ, and λ on our approach. When we study the effects of α, β, γ, we keep λ unchanged with the default value. And When we study the effects of λ, we keep α, β, γ, unchanged with the default values. We study the effects of parameters α, β, γ in Formula (3) to our approach, which are leveraged to tradeoff the historical user interest relevance, potential user interest relevance, and QoS utility, respectively. The results are presented in Figure 3. In this experiment, α and β are both weights to the functional relevance of Web service candidates, thus we set the equal value to them. In Figure 3(a), we can see that the density decreases as γ increases, and the effect is especially obvious when k is small. From this observation, we can conclude that larger γ causes better diversity. However, this phenomenon is only obvious when k is less than 20 under expansion ratio evalua-tion, as can be seen from Figure 3(b). As for the score evaluation, we can see from Figure 3(c) that larger γ tends to cause higher score, especially when k becomes large. In Figure 3(d), larger γ tends to cause slightly better diversified rank-ing. And the gap is relatively large when k is less than 20. This phenomenon is very similar to Figure 3(b) when k is less than 20, since the scores are nearly the same when k is less than 20 in Figure 3(c). Theoretically, in our approach, larger γ means the score tends to be more dominated by the non-functional quality (e.g., QoS utility), so the resulting recommended Web services tends to be more dissimilar to each other (indicating better diversity). Therefore, the experimental results in Figure 3 verified the fact.

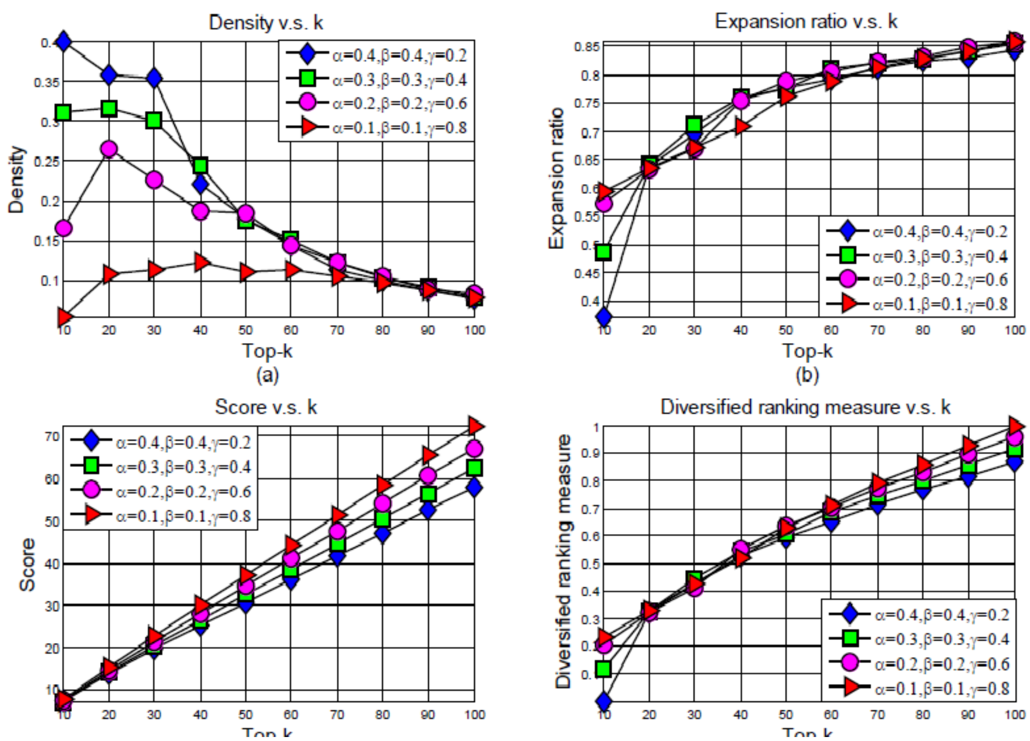

Fig 3(a),3(b),3(c),3(d) shows the comparision graph

Figure 3(d) shows the results for diversified ranking after the clustering it gives the best result when compare to all other combination of the algorithm and scores.

V. CONCLUSION

In this paper I presented the Clustering mechanism to sort out the distanced neighbor clustering using the k-means (which is suggested in base paper 1) and hierarchal. Web service clustering methods to improve the similarity computation and conduct real user survey to evaluate the usefulness of our method further. In addition, our proposed clustering measure ($F\ S$) mainly focuses on the neighborhood information of S in the Web service graph. In future the exact match pair in the N distance can be added to the cluster the future work will be done in the above contest.

REFERENCES

[1] Y. Jiang, J. Liu, M. Tang, X. Liu. "An effective Web service recommendation based on personalized collaborative filtering". *Proceedings of International Conference on Web Services.* IEEE Computer Society, pp. 211-218, 2011.
[2] M. Tang, Y. Jiang, J. Liu, X. Liu. "Location-Aware Collaborative Filtering for QoS-Based Service Recommendation". *Proceedings of International Conference on Web Services.* IEEE Computer Society, pp. 202-209, 2012.
[3] X. Chen, X. Liu, Z. Huang, H. Sun. "RegionKNN: a scalable hybrid collaborative filtering algorithm for personalized Web servicer ecommendation". *Proceedings of International Conference on Web Services.* IEEE Computer Society, pp. 9-16, 2010.
[4] Z. Zheng, H. Ma, M. R. Lyu, I. King. "Wsrec: a collaborative filtering based web service recommender system". *Proceedings of International Conference on Web Services.* IEEE Computer Society, pp. 437-444, 2009.
[5] L. Shao, J. Zhang, Y. Wei, J. Zhao, B. Xie, H. Mei. "Personalized qos prediction forweb services via collaborative filtering"
[6] M. Gong, Z. Xu, L. Xu, Y. Li, L. Chen. "Recommending Web Service Based on User Relationships and Preferences". *Proceedings of International Conference on Web Services.* IEEE Computer Society, pp. 380-386, 2013.
[7] S. M. Mcnee, J. Riedl, J. A. Konstan. "Being accurate is not enough: how accuracy metrics have hurt recommender systems". *Proceedings of CHI'06 extended abstracts on Human factors in computing systems.* ACM, pp. 1097-1101, 2006.
[8] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J. R. Wakeling, Y.-C. Zhang. "Solving the apparent diversity-accuracy dilemma of recommender systems". *Proceedings of the National Academy of Sciences*, Vol. 107, No. 10, pp. 4511-4515, 2010.
[9] C.-N. Ziegler, S. M. Mcnee, J. A. Konstan, G. Lausen. "Improving recommendation lists through topic diversification". *Proceedings of Proceedings of the 14th international conference on World Wide Web.* ACM, pp. 22-32, 2005.