Diversifying Web Service Recommendation Results Via Exploring Service Usage History

G. SUDHA REDDY¹, D. SAI DHAN², DR. P. SUNITHA³

¹PG Scholar, Dept of CSE, Siddhartha Institute of Engineering and Technology, Ibrahimpatnam, Hyderabad, Telangana, India.
²Assistant Professor, Dept of CSE, Institute of Engineering and Technology, Ibrahimpatnam, Hyderabad, Telangana, India.
³Professor & HOD, Dept of CSE, Institute of Engineering and Technology, Ibrahimpatnam, Hyderabad, Telangana, India.

Abstract: Web Service mining has become one of the predominant areas of Service Oriented Architecture. Web service discovery methods include syntactic based system and semantic based system. In the proposed work, both syntactic and semantic based approach is followed. The most widely used recommender technique is collaborative filtering. In this paper, author have a tendency to propose a novel net service recommendation approach incorporating a user’s potential QoS preferences and variety feature of user interests on net services. User’s interests and QoS preferences on net services area unit initial deep-mined by exploring the online service usage history. Then author have a tendency to cipher uncountable net service candidates by measure their connection with historical and potential user interests, and their QoS utility. Author have a tendency to conjointly construct an online service graph supported the purposeful similarity between net services. Finally, to have a tendency to gift an innovative diversity-aware net service ranking algorithmic rule to rank the online service candidates based on their scores, and variety degrees derived from the online service graph. In depth experiments area unit conducted based mostly on a true world net service dataset, indicating that our planned net service recommendation approach considerably improves the quality of the advice results compared with existing strategies.

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

I. INTRODUCTION

The primary goal of recommender’s is to supply personalized recommendations thus on satisfy users’ interests. A decent recommender system would provide less common papers that additionally draw the user’s interest though the user is powerfully fascinated by the papers written on a subject and also the recommender system is incredibly smart at ranking them so as of preference, it's a poor recommender system as a result of it shows similar pages repeatedly and not the varied one. Net service recommendation might be a way of proactively discovering and recommending applicable net services to finish users. Form of works is completed on service recommendation supported quality of service (QoS). Most of them used cooperative Filtering (CF) techniques, variety of them applied content-based approach, and plenty of them combined CF approach with content-based techniques. In recommender systems, once the k best recommendations unit of measurement very reasonably likes each other, many of them may even be useless to the user, and so the standard of k recommendations may even be very low. It's fascinating for a recommender system to return back a varied set of cases so as to produce the user with optimum coverage of the knowledge space.

Currently, diversity is taken into consideration as very important as similarity in many existing recommender systems. In recommender systems, once the k best recommendations are terribly the same as one another, several of them could be useless to the user, and so the utility of k recommendations may be terribly low. It’s fascinating for a recommender system to come back a various set of cases so as to provide the user with optimum coverage of the data space. Currently, diversity is taken into account as important as similarity in several existing recommender systems. As an example, Zhou et al, mentioned the diversity-accuracy perplexity of recommender systems, showing that hybrid methodology with diversity will improve the recommendation performance. Karl Waldemar Ziegler et al, planned that recommendation are often improved through topic diversification. Supported these facts, we tend to argue that diversity is additionally a crucial feature in internet service recommendation systems. During this paper, we tend to propose a completely unique service recommendation approach by taking diversity into thought. Author tend to incorporate the useful connectedness, QoS utility, and variety options of internet services for recommending well diversified top-k services to users.

Typically, a recommender system compares a user profile to some reference characteristics, and seeks to predict the 'rating' or 'preference' that a user would offer to associate item. These ratings or preference may be collected either actively or passively. Active user profile assortment includes: asking a user to rate associate item or product once usage, presenting 2 different things or product and asking user to rate them on a scale of ten. Passive user profile
G. SUDHA REDDY, D. SAIDHAN, DR. P. SUNITHA

assortment includes: Recording users’ history, analyzing his/her product purchased, analyzing social network profiles and discovering his/her likes and dislikes, etc. Since multiple internet Services give same practicality, another parameter should be introduced to be set as a deciding factor. QoS is that the appropriate deciding issue set of non-functional necessities like time interval, accessibility, throughput, convenience, etc. Current Universal description, discovery and Integration (UDDI) give support of internet Service retrieval by functional-requisition solely. Web Service mining supported cooperative Filtering and QoS is gaining importance.

II. RELATED WORK

In this paper the issue of diversity was addressed and solutions were provided. When individual recommendations are taken into consideration, a key challenge arises that most objects are recommended based on user or object similarity. Hence a new algorithm is specified to address the problem of diversity in combination with an accuracy focused algorithm Recommender systems use data on past user preferences to predict possible future likes and interests. The previously defined diversity recommendation techniques have similarity as the basis for recommendation. The risk of an overlap approach for recommendation rather than difference would expose users to a narrow band of services while the relevant niche items would be unnoticed. The algorithm proposed in the system focuses more on diversity factor rather than similarity. However this may pose a risk to accuracy. So a combination of accuracy and diversity focused method is used to solve this problem. A heat-spreading algorithm is designed to address the issue of diversity. A combination of accuracy and diversity related metrics are employed to evaluate performance using three different datasets. The hybrid algorithm does better as compared to other recommendation approaches in enhancing personalization of the results of individual user recommendations.

The Quality of Service (QoS) is an important factor being considered these days by the users for finding appropriate web services amongst the wide range of services available. Most of the QoS-aware recommendation systems use a rating-oriented prediction approach for the services intending at predicting the potential ratings that an active user may assign to the unrated services as accurately as possible which may not be beneficial in some of the application scenarios. In order to overcome this problem a ranking-oriented hybrid approach was proposed by combining item-based collaborative filtering and latent factor models. Similarity computation between web services is done in terms of correlation coefficient between their rankings rather than that in between the traditional QoS parameters. Moreover improvement in this measure is done using NDCG (Normal Discounted Cumulative Gain) for computing accuracy of top-k recommendation results. The paper aimed at issue of ranking in predicting missing QoS values in a given data set. [3] It is not sufficient only using QoS properties as the recommendation standard due to following mentioned reasons. First of all, QoS properties just reflect the attribute relationship, rather than mine the user’s potential needs. Then, it requires service invocations and imposes costs for the service users. At the same time, it consumes resources of service providers. Finally, if the user’s intention to choose services is not clear, QoS properties could not be employed to make appropriate recommendations. So it is more appropriate to employ the user-independent ones to estimate whether the Web service could satisfy the users expected demands .Therefore an algorithm named as URPC-Rec, was proposed to fulfill the task of reducing the dimensionality of sparse matrix and solving cold-start problem of recommendation systems. Furthermore, social network and recommendation systems are combined so as to employ user relationships and references to make personalized recommendations for them.

And the target of this paper is: according to user feedback, transforming the user-service matrix into the user-service category one and reducing the matrix dimensionality at first then, making sufficient use of the user and service labels; at last, excavating wide range interest of users rather than the single one. Missing QoS values are predicted using CF algorithms. QoS values include performance parameters like response time, availability, throughput etc. The values of these QoS parameters are highly dependent on network distances between the services and the user’s i.e their locations. This factor is taken into consideration and thus a location-aware approach is proposed for web services recommendation. At first location of users as well as services are considered and a hybrid location-aware QoS prediction method is given. User location and service location are used for improving accuracy and performance of QoS prediction. Thereafter large scale datasets are used to demonstrate a relationship exists between QoS similarity and user (service) location. Considering the above two factors a hybrid method of CF is proposed.

III. PROPOSE WORK

This paper targets to develop a system which implements Web service recommendation approach with clustering to find desired Web services for user. Hence we are proposing an efficient system which will work on A-NN algorithm to cluster the data and then rank the services by using suitable ranking algorithm. The system overcomes the problem and limitation of the existing system. The main objective of our system is to recommend top-k services to the user according to the user preferences which are mined from the user history. The proposed system helps generate a far better web services recommendation than the existing systems. It used the functional and non-functional requirements of the user; service usage history. The system will provide the suggestions from dataset which is provided from the admin. If new services are available, then they are to be updated by the admin in the dataset. The architecture of our system is as follows the functional modules of system are:

ANN Algorithm: This is the main functional block of our system. Artificial Neural Network will learn from the user history and accordingly the top-k list of services to be recommended to the user will be updated,

International Journal of Advanced Technology and Innovative Research
Volume. 08, IssueNo.16, October-2016, Pages: 3142-3148
User Preference: The preference value for the service will be updated once that service is consumed or used by the user. So it’s basically a numeric value to be fed to the ANN back for re-ranking of services.

Diversifying Web Service Recommendation Results Via Exploring Service Usage History

Fig.1. System Architecture.

Ranking: This block deals with the deciding the order of top-k services to be recommended to the user i.e. it returns the list of services by reordering it.

Feature Evaluation: This block deals with the similarity checking of the user query on the basis of type or name of the services from the services dataset. Hence other than the functional modules of the system user history and web service dataset are the storage systems. User and admin are two roles of a person where user can input a query and get the output whereas admin can add new services to the web services dataset.

IV. PERFORMANCE EVALUATION

In this section, we report the performance study of our proposed approach for Web service recommendation. Four parts are included: i) comparing our approach with the other Web service recommendation approaches including collaborative filtering approach, content-based recommendation approach, and their hybrid approach; ii) comparing our approach with the state-of-the-art diversified ranking methods; iii) precision evaluation; iv) studying the sensitivity of our approach under different tradeoffs of different parameters. We first describe the Web service dataset collected for the experiments and then report the experimental results.

A. Dataset Setup

To obtain solid experimental results, it is ideal to use a real world Web service dataset. Zheng et al, published a large-scale real world Web service dataset ac-quired in their WS-DREAM project1. WS-DREAM is a Web service crawling engine that collects publicly available WSDL file addresses from the Internet. It also collected QoS information of these Web services by using 339 distributed computers to monitor the Web services. This dataset has been widely used for performance evaluation by previous work on Web service recommendation. In our experiments, we used this dataset as our base dataset. Some Web services in the dataset are unavailable now, thus we only choose the available ones to form a new dataset which initially contains QoS data of 1982 Web services. The characteristics of the processed dataset are described in Table 1. There are 339 users and 1982 Web services in our dataset. The behavior of Web services in each invocation, as well as the observed QoS performance (response time and throughput) were recorded in our dataset.

Fig.2. QoS Values of Web services in the Processed Dataset (a) response time and (b) throughput.

TABLE 1. Characteristics of the Dataset

<table>
<thead>
<tr>
<th>Parameters</th>
<th>values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of users</td>
<td>339</td>
</tr>
<tr>
<td>Number of services</td>
<td>1982</td>
</tr>
<tr>
<td>Effective invocations</td>
<td>637595</td>
</tr>
<tr>
<td>Average effective</td>
<td></td>
</tr>
<tr>
<td>invocations per user</td>
<td>101</td>
</tr>
<tr>
<td>Observed QoS quality</td>
<td></td>
</tr>
<tr>
<td>Response time,throughput</td>
<td></td>
</tr>
</tbody>
</table>

For all the 1982 Web services in our dataset, both response time and throughput were employed in our experiments. The values observed by 339 users on the 1982 Web services are presented in Fig.2. Since Web services in the dataset may have different response time and throughput values for different users, for consistency, we use the average of the response time and throughput values of each Web service as its QoS.

TABLE 2. QoS Preferences of Users

<table>
<thead>
<tr>
<th>Users</th>
<th>$w_{rt}$</th>
<th>$w_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group 1 (68 users)</td>
<td>0.0–0.2</td>
<td>1–$w_{rt}$</td>
</tr>
<tr>
<td>Group 2 (68 users)</td>
<td>0.2–0.4</td>
<td>1–$w_{rt}$</td>
</tr>
<tr>
<td>Group 3 (68 users)</td>
<td>0.4–0.6</td>
<td>1–$w_{rt}$</td>
</tr>
<tr>
<td>Group 4 (68 users)</td>
<td>0.6–0.8</td>
<td>1–$w_{rt}$</td>
</tr>
<tr>
<td>Group 5 (67 users)</td>
<td>0.8–1.0</td>
<td>1–$w_{rt}$</td>
</tr>
</tbody>
</table>
In order to simulate the variety of QoS preferences from different users, we divide the 339 users into 5 groups. Users in different groups are supposed to have different weights on response time \( w\text{rt} \) and throughput \( w\text{tp} \) for the Web services. The user partition is presented in Table 2.

**B. Evaluation Metrics**

In our experiments, we randomly choose 10 service users from the dataset as the active (test) users. We report the average performance evaluation from several aspects. In our Web service recommendation approach, score and diversity are incorporated. The score involves the functional relevance (w.r.t user interest) and non-functional relevance (w.r.t user QoS requirements) of Web services. Thus, in our experiments, we mainly evaluate the performance on the total score, diversity of the recommendation list, and the overall diversified ranking measure defined. The precision metric will be presented later in Section 4.5. When comes to weighted summation, we do normalization processing with similar method to before weighted summation for each factor. We employ two metrics to measure the diversity. One is proposed, which makes use of the density of the induced sub-graph by the top-k ranked nodes. The density of a graph is defined as the number of edges (excluding self-links) presenting in the graph divided by the maximal possible number of edges in the graph. The density of sub-graph \( S \) formed by the top-k ranked nodes is defined as Formula (1).

\[
d(S) = \frac{|\{(u,v)\mid u \in S, v \in S, (u,v) \in E\}|}{\beta \times |S|(|S|-1)}
\]  

The second metric is the expansion ratio given in Def. 3. The rational is that the larger expansion ratio of the top-k ranking nodes indicates the better diversity. Here, we present the parameter settings and experimental environment. In our proposed approach, there are several parameters: \( \text{Num} , \psi_1, \psi_2, \theta_u, \theta_p, \alpha, \beta, \gamma, \lambda, \) and \( \tau \). We set \( \text{Num}=10, \psi_1=\psi_2=0.5, \theta_u=\theta_p=0.6, \alpha=\beta=0.4, \gamma=0.2, \lambda=0.5, \) and \( \tau \) equals to the average similarity calculated with Formula (2). Therefore, in our experiments, we take the score and diversity as equally important factors. While we also test the impact of variance parameters in our approach. All the experiments are conducted on a 2.50 GHz CPU and 4 GB PC running Windows 7. All algorithms are implemented by MATLAB 2009a and Visual C++ 6.0.

\[
\tau = \frac{2 \sum s_{ij}^w}{N(N-1)}
\]  

**C. Comparison with Other Service Recommendation Approaches**

1. **Baselines**: We compare our proposed approach with the Content-based Web service recommendation approach (rank Web services according to their historical user interest relevance and QoS utility), CF-based Web service recommendation approach (rank Web services according to their potential user interest relevance and QoS utility), and Hybrid approach (rank Web services according to the combination of historical user interest relevance, potential user interest relevance, and QoS utility) under the diversity, score, and the ranking measurement defined before.

2. **Diversity Evaluation**

In this section, we conduct experiments to study the performance of our approach with the diversity metric and make comparison with its competitors. As shown in Fig.3 (a), WSRD achieves the best diversity under the density, followed by the CF-based approach, then Hybrid and Content-based approaches. Therefore, we can conclude that Hybrid and Content-based approaches exhibit poor performance in density evaluation. With expansion ratio measure, WSRD also outperforms the other competitors, which can be seen from Fig.3 (b). The expansion ratio of CF-based and Hybrid approaches are comparatively close. When \( k \) becomes larger, the effect is more obvious. Content-based approach achieves the worst expansion ratio. Based on the above observations, we can conclude that WSRD achieves the best diversity performance than its competitors.

3. **Functional and Non-functional Evaluation**

The score defined in describes the user interest relevance and QoS utility of Web service candidates, so it can be used to evaluate the functional and non-functional quality of recommended Web services. In this subsection, we conduct experiments to study the performance of WSRD with the overall score of Web services in the recommendation list compared with its competitors. As shown in Fig.3(c), WSRD, Content-based, CF-based, and Hybrid approaches achieve very similar over-all scores for their recommendation lists. Our approach is slightly worse than the Hybrid approach, with a very small difference so that it can be neglected. Content-based approach is slightly worse than WSRD when \( k \) is larger than 50. The CF-based approach has the most similar performance as WSRD.

**Fig.3. Comparison of Web Service Recommendation Approaches**

4. **Overall Evaluation**

The diversified ranking measure defined it captures the functional, non-functional and diversity feature of Web
Diversifying Web Service Recommendation Results Via Exploring Service Usage History

services in a recommendation list, so it evaluates the overall quality of a Web service recommendation list. In Fig.3 (d), WSRD achieves the best performance under the diversified ranking measure, followed by the CF-based approach, then Hybrid and Content-based approaches. Content-based approach shows the worst performance under the diversified ranking measure. Therefore, WSRD outperforms the competitors when both the diversity and the score of the Web service recommendation list are taken into consideration.

D. Comparison with Other Diversified Ranking Methods

1. Baselines: We compare our proposed algorithm with three diversified ranking methods in graph domain again with the diversity, score, and the overall ranking measurement as evaluation metrics. The three baselines are: (1) Grasshopper: Grasshopper is a ranking algorithm that leverages an absorbing random walk to achieve diversity, (2) Manifold ranking with stop points (Mani_stop): The Mani_stop algorithm was proposed, which is very similar to the Grasshopper algorithm; (3) DivRank: The DivRank makes use of the stationary distribution of a vertex reinforced random walk to rank nodes. In this experiment, we implement the proposed algorithm and compare with three baselines described above over our dataset. From Fig.4 (a), we can see that WSRD achieves the best diversity under the density metric, followed by Mani_stop algorithm, and then Grasshopper and DivRank algorithms. There is a large gap between WSRD and Grasshopper, implying that the Grasshopper and DivRank algorithm exhibit poor performance to enhance diversity in our Web service dataset. And Mani_stop shows a little worse diversity than the WSRD. Under the expansion ratio measure (i.e., Fig.4 (b)), WSRD outperforms the competitors. Grasshopper and DivRank achieve nearly equivalent diversity.

E. Precision Evaluation

To further evaluate the effectiveness of our approach, we evaluated the precision of our approach and compared it with the state-of-the-art service recommendation approach-CF based approach as shown in Fig.3(d), and the state-of-the-art diversified ranking method — Grasshopper as shown in Fig.4(d). Since there is no ground truth in Web service datasets, we use the Hybrid approach as the ground-truth rank. The precision is defined by Formula (3).

\[ Pre = \frac{|S \cap S'|}{|S'|} \] (3)

where \( S \) denotes the set of services in the top-k diversified ranking list produced by the service recommendation approach or diversified ranking method, and \( S' \) denotes the set of services in the top-k ranking list by Hybrid ser-vice recommendation approach which always yields the k most relevant services as can be seen from Fig.3(c).

In this experiment, we select three groups of 10 active users to observe the average precision (i.e., each group with 10 active users). Fig.5 shows the results. From Fig.5 (a), we can clearly see that WSRD consistently outperforms CF-based approach and Grasshopper in the experiment of group 1. In group 2, we can observe that three approaches generate comparable rank, and the performance of WSRD is slightly better than Grasshopper, while CF-based approach is the worst. In group 3, Grasshopper does not perform well, and CF-based approach is in the middle. In contrast, the performance of our approach is very stable in three groups of experiments. With the above observation, we can conclude that WSRD is better than CF-based approach and Grasshopper.
F. Impacts of Parameters

In this section, we conduct experiments to study the effects of parameters in our approach. Specifically, we study the impacts of $\alpha$, $\beta$, $\gamma$, and $\lambda$ on our approach. When we study the effects of $\alpha$, $\beta$, $\gamma$, we keep $\lambda$ unchanged with the default value. And when we study the effects of $\lambda$, we keep $\alpha$, $\beta$, $\gamma$, unchanged with the default values. We study the effects of parameters $\alpha$, $\beta$, $\gamma$ in 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 Fig.6. In this experiment, $\alpha$ and $\beta$ are both weights to the functional relevance of Web service candidates, thus we set the equal value to them. In Fig.6 (a), we can see that the density decreases as $\gamma$ increases, and the effect is especially obvious when $k$ is small. From this observation, we can conclude that larger $\gamma$ causes better diversity. However, this phenomenon is only obvious when $k$ is less than 20 under expansion ratio evaluation, as can be seen from Fig.6 (b). As for the score evaluation, we can see from Fig.6(c) that larger $\gamma$ tends to cause higher score, especially when $k$ becomes large. In Fig.6 (d), larger $\gamma$ tends to cause slightly better diversified ranking. And the gap is relatively large when $k$ is less than 20. This phenomenon is very similar to Fig.6 (b) when $k$ is less than 20, since the scores are nearly the same when $k$ is less than 20 in Fig.6(c). Theoretically, in our approach, larger $\gamma$ 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 Fig.6 verified the fact.

Next, we study the effect of parameter $\lambda$ in, which is leveraged to tradeoff the score and diversity. The experimental results are presented in Fig.7. In Fig.7 (a), the density decreases as $\lambda$ increases, since smaller density means better diversity. And in Fig.7 (b), expansion ratio increases as $\lambda$ increases, since larger expansion ratio means better diversity. Therefore, diversity generally increases as $\lambda$ increases. This is because a larger $\lambda$ means more weights are assigned to the diversity feature. In contrast, according to Fig.7(c), the total score de-creases slightly as $\lambda$ increases, but the decrease is trivial. From the above observations, improving the diversity will oppositely reduce the score, while the change of $\lambda$ is not much sensitive to the diversified ranking measure as can be seen from Fig.7 (d). Hence, in practice, users can set a reasonable tradeoff between the score and the diversity according to their preferences.

V. CONCLUSION

In this paper author bestowed net an internet an online Service recommendation approach with diversity to seek out desired web services for content users. Author’s gives useful interest, QoS preference, and variety feature for recommending top-k heterogeneous net services. The top-k heterogeneous top-k services stratified list supported their useful connection and potential user interest connection, non useful connection like QoS quality and variety feature. Here shows that the projected approach improves the online service recommendation performance in terms of diversity, the mix of QoS utility, useful connection and therefore the heterogeneous ranking analysis. Web Service choice supported Hybrid cooperative Filtering and QoS-Trust analysis addresses numerous Collaborative Filtering issues specifically cold-start downside, grey sheep downside, word downside, ramp-up downside, shillings attack, information exiguity and measurability. Additionally it achieves improved precision-recall price, result accuracy and promising retrieval time. Author introduced a new QoS-based web service selection and ranking algorithm with trust and reputation management support. Author shown that our selection and ranking solution yields very good results in most cases as the proposed reputation management
mechanism is robust against various cheating behaviors, the results are generally of good quality even in hostile situations in which many different types of cheaters make up a high percentage of the overall users and report values with remarkable variances.

VI. REFERENCES
[1] Guosheng Kang, Student Member, IEEE, Mingdong Tang, Jianxun Liu, Member, IEEE, Xiaoqing(Frank) Liu, Member, IEEE, and Buqing Cao, Member, IEEE, “Diversifying Web Service Recommendation Results via Exploring Service Usage History”, IEEE Transactions on Services Computing.


