**International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)**
**ISSN: 0976-1353 Volume 9 Issue 3 –SEPTEMBER 2014.**

# Effective QoS Web Service Advice and Apparition

E. Naga Prabhakar[*1], and K.Ramesh Rao[#2]

[*]*Student (M.Tech), Dept of CSE, Ananthalakshmi Institute of Technology & Sciences,*
*Affiliated to JNTUA University, ALITS- Anantapur*

[#] *Asst Professor, Dept of CSE, Ananthalakshmi Institute of Technology & Sciences,*
*Affiliated to JNTUA University, ALITS- Anantapur*

***Abstract*— Some of the software components that are designed to support interoperable peer to peer interaction over a network are the web services. With the proliferation of web services, QoS based approach is becoming more important to service recommendation (SR). the present SR performance is not satisfactory. This is because of two reasons. First approach fails to consider the QoS variance based on users location. Second, all the recommenders are balck boxes , these provide limited information on the service candidates performance. Here we propose a algorithm called novel collaborative filtering algorithm. This algorithm is designed for large scale web service recommendations. This scheme employs the characteristic of QoS and achieves improvement on the recommendation accuracy. To provide better services to service users we need to understand the rationale of the recommendation and remove some sort of confusion, we make use of recommendation visualization technique. This technique is used to show how recommendation is grouped with other options. More than 1.5 million QoS records has been experimented. These experimental results show the effectiveness and efficiency of present approach.**

**General Terms**

**Service recommendation, QoS, collaborative filtering, self-organizing map, visualization**

## I. INTRODUCTION

Some of the software components in real applications are web services. Web services are designed to support interoperable peer to peer interaction over a network. Web services are adopted as a delivery mode in business. This process has fostered a new paradigm shift from the development of monolithic application to the dynamic setup of business process. Now days, web services have attracted more attentions from industrial point of view and academia point of view. The public web service is randomly increasing in their number.

While implementing service oriented applications, service users get a list of web services from search engines or service brokers. These should meet the specific functional requirements. Then they need to identify the particular requirement from the functionally equivalent candidates. But, it is tough to select the limited knowledge of their performance. Here best approaches to service recommendation and service selection are needed urgently.

QoS (quality of service) is majorly used to represent the nonfunctional performance of web services this is been considered as the important factor in selection of service. Quality of service is defined as a user perceived properties including reputation, availability response time etc. presently, it's not practically possible for users to achieve QoS information by evaluating all the service candidates. Because, conducting web service invocations is resource consuming and time consuming.

Some QoS properties such as reliability and reputation are very difficult to evaluate. To evaluate number of invocations and long duration observations are required. It is not practically possible to acquire QoS information from third party communities or from service users. Different users may observe different Qos performance of the same web service. The value of QoS evaluated by one user cannot be directly used by another in service recommendation and service selection.

The main objective here proposed is to make personalized QoS based web service recommendation for different users. This helps the users to select the optimal one among various functional units.

These web service recommender systems based on CF works by gathering user observed QoS records of different web services and mapping together users who ever share the same information needs or same tastes. Users, that belongs to CF system share their opinions and judgments on web services. Then the system provides useful recommendations that are personalized. Here three problems that are unsolved based on the previous work affect the performance of present service recommender systems.

First, existing approaches fail to identify the QoS variation with user's physical locations. After analyzing the real-world web service data set that contains 1.5 million service invocation results related to 100 public services evaluated by users from more than 20 countries. We identify that some QoS properties like availability and response time highly relate to the users physical locations. For example, the response time of a service observed by users who are closely located with each other usually changes suddenly around a certain value, while it sometimes varies significantly between users very far away from each other.

Second, the online time complexity of memory based CF recommender systems. If the number of web services increases then users will face a great challenge to current systems. With O(mn) time complexity.
Where

m is the number of services
n the number of users.

Third, all the current web service recommender systems are black boxes. These provide a list of ranked web services with no transparency into the reasoning behind the recommendation results. It is not likely for users to believe a recommendation when they have no information of the underlying rationale. The opaque recommendation procedures prevent the acceptance of the recommended services. Herlocker proposed that explanation capabilities is an important method of building trust in recommender systems because users are more likely to trust a recommendation when they know the reason behind it.

To represent the first two problems, we propose a CF algorithm that is innovative for QoS based web service recommendation. To represent the third problem and enable a fast understanding of the web service recommendation rationale. Here, we provide a personalized map for browsing the recommendation results. The map shows the QoS relationships explicitly. These relationships are of the recommended web services as well as the underlying structure of the QoS space by using map metaphor like 623 areas, dots and spatial arrangement.

Three different types of contributions are proposed for our work:

Firstly, we design a visuallyrich interface to browse the recommended web services, which enables a better understanding of the service performance.

Secondly, we combine the memory based CF algorithm and model based CF algorithm for web service recommendations, this improves the recommendation accuracy and time complexity. These are compared with previous service recommendation algorithms.

Thirdly, we conduct comprehensive experiments to evaluate our approach by deploying real world web service QoS data set. More than 1.5 million web service QoS records from more than 20 countries are used in our experiments.

## II. Related Works

In motivating scenario we present an service searching scenario that is online, to show the research problem of this scenario. For example, alice is a software engineer who is working in india. She needs an service for email validation, that filters emails. Then she searches an service registry that is located in US, she gets a list of recommended services in some order based on the service average response time. Then Alice tries the top most 2 services which are provided by a Canadian company and finds that response time is more than her expectation. Then she finds that the service ranking is done based on the evaluation performed by the US registry and response time may vary greatly due to the different user context, such as user network conditions, user locations etc. based on suggestions from her colleagues, she tried service k provided by some local company through ranked lower in recommendation list. After trying, alice thinks that service k has very good performance and meets the requirements.

The problem faced by alice is to find a service that meets non functional requirements and functional requirements. The present way of identifying a suitable web service is inefficient, so alice needs to try the recommended services one by one. To specify this challenge, we propose a accurate approach to service recommendation by considering region factor. Here we try to provide more users friendly and more informative interface for browsing the recommendation results rather than list that is ranked. Using this procedure user are able to know more about overall performance of the assumed services, and thus trust the recommendations.

The basic procedure of our approach is that users located closely with each other will have same service experience when compared with users who live far away from each other. Based on web 2.0 websites that emphasize information sharing, interaction, and collaboration. We deploy the idea of user collaboration in our webservice recommender system. Different from sharing knowledge or sharing information on wikis or blogs, users are encouraged to share their observed quality of service performance with others in the system that is recommended. If the user contributes more QoS information automatically, the more accurate service recommendations the user can obtain. Here more user characteristics can be analysed from the information contributed by user.

Based on quality of records that have been collected our recommendation approach is designed as a 2 phase process

We divide the basic users into different regions based on

- Physical locations on we services
- Historical quality of service experience on web services.

We identity similar users for present users and make QoS prediction for unused services.

For the current user, services with the best predicted QoS will be recommended.

***Phase 1: Region Creation***

In web service recommender system, users usually provide QoS values on a small number of web services. Traditional memory-based CF algorithms suffer from the sparse user contributed data set, since it's hard to find similar users without enough knowledge of their service experience. Different from existing methods, we employ the correlation between users' physical locations and QoS properties to solve this problem. In this paper, we focus on the QoS properties that are prone to change and can be easily obtained and objectively measured by individual users, such as response time and availability. To simplify the description of our approach, we use response time (also called round-trip time (RTT)) to describe our approach. We assume that there are n users and m services. The relationship between users and services is denoted by an n _ m matrix R. Each entry Ri;j of the matrix represents the RTT of service j observed by user i and ? is the symbol of no RTT value. Each user i (i 2 f1; 2; . . . ; ng) is associated with a row vector Ri representing his/her observed RTT values on different web services. The user að a 2 f1; 2; . . . ; ngÞ is called the active user or current user if he/she has provided some RTT records and needs service recommendations.

## III. RECOMMENDATION VISUALIZATION

CF based web service recommender systems deploy the QoS that is predicted, mainly in 2 ways

- While users questions a service with specific functionality, this functionality is the one with the best predicated QoS is recommended to them.
- To help users top-k best performing services are recommended to discover potential services.

A service list ranked by predicated quality of service (QoS) as recommendation, we should develop an exploratory recommendation tool that provides valuable insight into the quality of service (QoS) space. This enables an improved understanding of the total performance of web services.

The quality of service space visualization of almost all web services on a map will evaluate the rationale back of QoS based service recommendations. QoS space visualization is more than method of computing or a picture. Qos space visualization transforms the high dimensional QoS data information into a visual form. This enables service users to browse, observe, and understand the information.

QoS map is drawn by using 2 steps

1) Dimension reduction step

2) Map creation step

### *1) Dimension reduction step:*

In this step we create a two dimensional representation of high dimensional QoS space by using SOM(self organized map). Here each web service is coordinated to a unique 2D coordinates. SOM is a popular unsupervised artificial neural network. SOM is successfully applied to many areas. Some of the areas are medical engineering, speech recognition and document organization. SOM is used for information visualization. This can be viewed as a mapping of high dimensional input space to a lower dimensional output space. The SOM output space is a network of neurons. These neurons are located on a regular 2D grid. Each neuron is equated with a prototype vector which has the equal dimension of the input apace. All the neurons are connected to adjacent neurons. This relation indicates the structure of the map. The structure of the map can be rectangular or hexagonal lattice. When training phase is completed automatically data points close to each other in input space are matched to the nearby neurons. This technique is called topology preserving projection property. With this property, SOM is regularly employed in data survey applications. This helps us to visualize the structure of high dimensional complex data.

The main goal of using SOM in our context is to translate CF employed QoS data into a 2D discrete map in topologically ordered fashion. Here the QoS map is to show the RTT differences of different web services. Input of SOM is QoS matrix. QoS matrix contains QoS values provided by all users and all the services.

Here, the data set is a sparse. The number of QoS values differs from service to service. The real data set cannot reveal the internal structure of the QoS space to solve this problem, a QoS set is derived from region centers, this QoS is employed to train the SOM. Let l denote the input space dimension (QoS data) and $q = [q_1, q_2, \ldots q_l]^T$ denote an input pattern (QoS vector of a service). The prototype of neuron j is denoted by

$$\omega_j = [\omega_{j1}, \omega_{j2}, \ldots, \omega_{jl}]^T, j = 1, 2, \ldots, k,$$

where total number of neurons in the network is represented with k. The SOM is trained repeatedly. In each training step, a QoS vector called q is randomly chosen from the input data set. The distance between q and all prototypes is calculated. The neuron with the closest prototype to q is chosen as the BMU (best-matching unit). Let i(X) be the index of BMU, we determine i(x) by applying…….

### *2) Map creation step:*

One approach to QoS map for web service is to assign a unique portion to each web service of the 2D display area. Then put these services adjacent to each other based on same QoS performance. Based on training result, for each service we assign unique coordinates by randomly distributing them in cell boundary based on corresponding neuron. Then the base map is formed by using voronoi diagram. Here each service corresponds to a unique polygon. But base map alone is not sufficient, and becomes too complex to reveal the internal data relationships, when applied to large set of services. A generalized map is needed. Here, hierarchical clustering method is used to cluster web services based on QoS similarity. A generalized map is formed by merging service polygons belonging to the same cluster. The feature that is topological preserving of SOM guarantees that services belong to the same cluster. These usually be neighbouring polygons on the map.

We deploy web service recommendations on the map, using the QoS values that are predicated. The one with the best predicated QoS will be marked on the map. We highlight the top k best performing services. This is done to help users to find potential service.

## IV. RECOMMENDED MAP DISPLAY

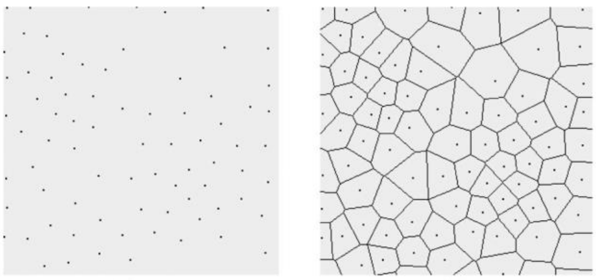
We show how to create a map that shows the similarity of RTT variance of web services. We demonstrate how to put personalized web service recommendations for active users on the map. We assume 2,700 training users and set given = 10, density = 0:5. After completing the region aggregation phase, 17 regions are formed. Here the input of SOM is a 100 X 17 RTT matrix containing 100 services (rows) and their related performance on 17 regions (columns). Each web service QoS is an inout vector. We train an SOM that has neurons which are arranged in a 60X80 hexogonal lattice. The prototypes of som are randomly initialized. The Gaussian function is taken as the neighborhood function. We train SOM in 2 phases:

Rough training phase

Fine tuning phase

After completing training process, each service is mapped on to a neuron. Unique coordinates are assigned to each service by randomly distributing them within the boundaries of the related neuron cell

But to create a geographic map, each known point is assigned to a unique portion of the map display by using a voronoi figure



(a) (b)

Fig. 6. Map creation. (a) 2D locations of services derived from SOM training. (b) Voronoi diagram of services based on the 2D locations.

We adopt the hierarchical clustering to the services depending on their similar Qos,s and obtained 42 clusters. We simplify the map by combining the neighboring polygons if they are in the same cluster. Here, we form a generalized map that shows the internal structure of the QoS space. Pointing individual service is an integral part of the map creation. The moto is to identify the potential services with optimal QoS values. Then these values are shown to users. Different label styles are used to mark services showing how strongly we evaluate them. For example, the 10 top most, best performing services are marked with 12-point boldface; good predicted QoS services are labeled with 8-point non bold face; here dots indicate those services with poor predicted QoS values. Fig. 7 shows the final map for web service recommendations.

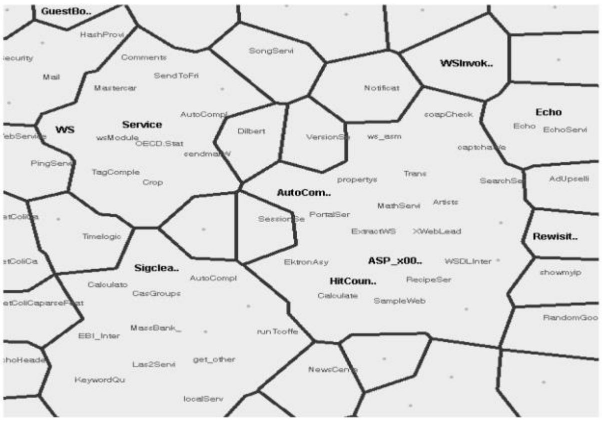


Fig. 7. Final map with service recommendations

The map can be created for one region to show the similarity of services based on a set of quality of service properties. 3 QoS properties

Reputation

RTT

Cost

Reputation is a qualitative property. RTT and cost are quantitative properties. Each region cnter is a matrix. This matrix is composed of 3 columns. They are RTT cost Reputation

| | RTT | Cost | Reputation |
|---|---|---|---|
| $WS_1$ | 1200 | 10 | 1 |
| $WS_2$ | 1000 | 20 | 2 |
| … | … | … | … |
| $WS_{100}$ | 800 | 50 | 5 |

Fig. 8. Region center matrix

As different properties have different ranges, we first minimize each of them to [0, 1] by using following steps:

First, find the maximum Rmax and minimum Rmin value of the property.
Second, for each original value (R) submitted by the user, the normalized value R’ is calculated by using the following formulae

$$R' = \frac{R - R_{\min}}{R_{\max} - R_{\min}}$$

By using these normalized values, each property will have equal weights in the SOM training. The creation of map process is the same, and we can obtain a map reflecting the web service QoS similarity of a region.

## V. CONCLUSION

Here we have proposed an innovative approach to web service visualization and recommendation. The proposed algorithm employs the characteristic of QoS by clustering users into various regions, when compared with previous work. Depending on region feature, a filtered nearest- neighbor algorithm is proposed to generate QoS that is predicated. The last service recommendations are kept on a map to reveal the internal structure of qos space. This helps users accept the recommendations. A previous result shows that our approach significantly improves the accuracy of prediction than the existing methods. These are independent of sparseness of training atrix. The online time complexity of our approach is better than the traditional CF algorithms.

Our recommendation approach considers the coordination between user’s physical locations and QoS records by using IP addresses. This process achieved good prediction performance. In some cases, users in same locations may observe various QoS performance of the same went service. Other than the user physical location, we will find more contextual information that enforces the client side QoS performance. Here contextual information is network conditions, workload of the servers, and activities that are arrived out by users with web services. More investigations on the distribution of RTT and coordination between various QoS properties will be conducted in web services. The key indicator of the effectiveness of recommender system is the user acceptance rate of the recommendation

## REFERENCES

[1] M.B. Blake and M.F. Nowlan, “A Web Service Recommender System Using Enhanced Syntactical Matching,” Proc. Int’l Conf. Web Services, pp. 575-582, 2007.

[2] J.S. Breese, D. Heckerman, and C. Kadie, “Empirical Analysis of Predictive Algorithms for Collaborative Filtering,” Proc. 14th Conf. Uncertainty in Artificial Intelligence (UAI ’98), pp. 43-52, 1998.

[3] Y.H. Chen and E.I. George, “A Bayesian Model for Collaborative Filtering,” Proc. Seventh Int’l Workshop Artificial Intelligence and Statistics, http://www.stat.wharton.upenn.edu/~edgeorge/ Research_papers/Bcollab.pdf, 1999.

[4] S. Haykin, Neural Networks: A Comprehensive Foundation, second ed. Prentice-Hall, 1999.

[5] J.L. Herlocker, J.A. Konstan, and J. Riedl, “Explaining Collaborative Filtering Recommendations,” Proc. ACM Conf. Computer Supported Cooperative Work, pp. 241-250, 2000.

[6] J. Himberg, “A SOM Based Cluster Visualization and Its Application for False Coloring,” Proc. IEEE-INNS-ENNS Int’l Joint Conf. Neural Networks, pp. 587-592, 2000, vol. 3, doi:10.1109/ IJCNN.2000.861379.

[7] Hsu, and S.K. Halgamuge, “Class Structure Visualization with Semi-Supervised Growing Self-Organizing Maps,” Neurocomputing, vol. 71, pp. 3124-3130, 2008.