International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 8 Issue 1 –APRIL 2014.

# Personalized Quality of Service – Aware Web Service using Road Knowledge and Route Choice

[1*]Mr.P.Segar, and [2#]Mrs.R.Kalpana

[1]*M.Tech, Pondicherry Engineering College, Pondicherry*
[2]*Associate Professor, Pondicherry Engineering College, Pondicherry*

***Abstract*— Web services are software components designed to support interoperable machine-to-machine interaction over a network. The adoption of web services as a delivery mode in business has fostered a new paradigm shift from the development of monolithic applications to the dynamic setup of business process. Service users are not knowledgement about all the different types of web services. Hence, Web Service Recommender System (WSRS) is needed to provide quality of service to the users. These web service recommendation systems use collaborative filtering algorithm to deliver the most relevant data for the given queries. In this paper, a control centre is designed to improve the quality of service to the end users and to provide the confidences of the predicted values. This can be employed effectively by the service users for better web service selection. Our proposed techniques are effective and efficient when compared to the previous approaches through our experimental and simulation analysis.**

**Keywords-Web Databases, Data Alignment, Data Filtering, Multimode Text, Service recommendation, QoS, collaborative filtering, self-organizing map, visualization**

## I. INTRODUCTION

Databases are established technologies for managing large amount of data. Web is a good way of presenting information. Alignment and annotation of data increases the efficiency of searching and updating information. Data alignment is the way of arranging data and accessing in computer memory. Data annotation is the methodology for adding information to a document, a word or phrase, paragraph or the entire document. In other words data unit annotation is the process of assigning meaningful labels. Data annotation enables fast retrieval of information in the deep web. A result page retrieved from a web database consists of several search result records (SRRs) and each result records consist of multiple data units. A data unit is defined as the values that represent real world entities. These data units are encoded dynamically into result pages for human browsing and converted into machine process able unit and assigned meaningful labels. The encoding of data units requires lot of human efforts to annotate data units manually. Thus, lack in scalability. To overcome this, automatic assigning of data units within the SRRs is required. Annotated in different aspects and aggregated to predict a final label. Finally, a wrapper is constructed. Wrappers are commonly used as translators which annotate new result pages from the same web database. This automatic annotation approach is highly effective and more scalable.

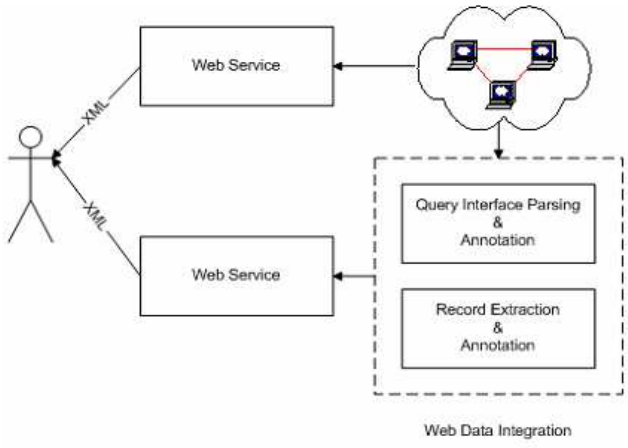


Figure 1: High level Web Services Architecture

There are many existing a number of techniques proposed for quality of services in web for personalized process. The basic idea of the existing model is that to locate users closely with each other who are more likely to have similar service experience than those who live far away from each other. Inspired by the success of Web 2.0 websites that emphasize information sharing, collaboration, and interaction, we employ the idea of user-collaboration in our web service recommender system. Different from sharing information or knowledge on blogs or wikis, users are encouraged to share their observed web service QoS performance with others in this recommender system. The more QoS information the user contributes, the more accurate service recommendations the user can obtain, since more user characteristics can be analyzed from the user contributed information. 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. The correlation between users' physical locations and QoS properties is needed to solve this problem. In this it 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. The Region as a group of users who are closely located with each other is likely to have similar QoS profiles. Each user is a member of exactly one region. Regions need to be internally coherent, but clearly different from each other. The region creation phase is designed as a three-step process. In the first step, users with similar IP addresses are categorized into a small region and extract region features. In the second step, we calculate the similarity between different regions. In the last step, we aggregate highly correlated regions to form a certain number of large regions. After the phase of region aggregation, queries are clustered into a certain number of regions based on their physical locations and historical QoS similarities. The service experience of users in a region is represented by the region center. With the compressed QoS data, searching neighbors and making predictions for an active user can be computed quickly. Traditionally, the QoS prediction methods need to search the entire data set, which is rather inefficient. In this approach, similarity between the active user and users of a region is computed by the similarity between the active user and the region center. Moreover, it is more reasonable to predict the QoS value for active users based on their regions, for users in the same region are more likely to have similar QoS experience on the same web service, especially on those region-sensitive ones. In order to overcome, the obstacles in the previous approach we implementing new concepts in this paper. A novel collaborative filtering algorithm designed for large-scale web service recommendation. This collaborative filtering algorithm uses a rule-based mechanism to determine behavior consistent information based control strategies for route guidance in a dynamic vehicular traffic system. This approach employs the characteristic of QoS and achieves considerable improvement on the recommendation accuracy.

The predicted QoS values can be employed for the Web service recommendation and selection by the following ways:

1) For functionally equivalent Web services, the one with best predicted QoS performance can be recommended to the active user.

2) Service Recommender System can recommend the top best performing Web services, which may not have equivalent functionality, to the service users to help them discover the potential Web services.

3) Service Recommender System can also recommend the top active service users, who have good predicted QoS values on a certain Web service, to the service provider to help the provider find its potential customers. Different from all other existing prediction methods, SRS not only provides the predicted QoS values for the active users, but also provides the confidences of the predicted values, which can be employed by the service users for better Web service selection. Our proposed techniques are effective and efficient when compared to the previous approaches through our experimental and simulation analysis.

The rest of the paper will be organised as follows: In section 2, we see about the related works of the paper. In section 3, we discuss about the proposed method. The algorithms and simulation are shown in the section 4 and 5. The conclusion of our paper is in section 6.

## II. Related Works

In this section, we will see the some of the related works to using different approaches:

Many web sites contain large collections of pages generated using a common template or layout. For example, Amazon lays out the author, title, comments, etc. in the same way in all the book pages. The values used to generate the pages (e.g., the author, title, etc.) typically come from a database. We have studied the problem of automatically extracting database values from such a collection of web pages automatically without any human input. Please follow this link for the paper discussing the techniques that we have developed for the above problem. This page contains the experimental results of applying our techniques to real web page collections. Some of the collections that we used in our experiments were obtained from Road Runner Project which tries to solve a similar problem. The other collections were manually crawled from well-known data-rich sites like E-bay and Netflix.

The self-organizing map (SOM) is widely used as a data visualization method in various engineering applications. It performs a non-linear mapping from a high-dimensional data space to a lower dimensional visualization space. In this paper, a simple method for visualizing the cluster structure of SOM model vectors is presented. The method may be used to produce tree-like visualizations, but the main application here is to get different colour coding that express the approximate cluster structure of the SOM model vectors. This coloring may be exploited in making false color (pseudo color) presentations of the original data. The method is especially meant for making an easily implementable, explorative cluster visualization tool.

Data extraction from web pages is performed by software modules called wrappers. Recently, some systems for the

automatic generation of wrappers have been proposed in the literature. These systems are based on unsupervised inference techniques: taking as input a small set of sample pages, they can produce a common wrapper to extract relevant data. However, due to the automatic nature of the approach, the data extracted by these wrappers have anonymous names. In the framework of our ongoing project Road Runner, we have developed a prototype, called Labeller that automatically annotates data extracted by automatically generated wrappers. Although Labeller has been developed as a companion system to our wrapper generator, its underlying approach has a general validity and therefore it can be applied together with other wrapper generator systems. We have experimental the prototype over several real-life web sites obtaining encouraging results.

In this paper, we propose meta-learning as a general technique to combine the results of multiple learning algorithms, each applied to a set of training data. We detail several Meta learning strategies for combining independently learned classifiers, each computed by different algorithms, to improve overall prediction accuracy. The overall resulting classifier is composed of the classifiers generated by the different learning algorithms and a meta-classifier generated by a meta-learning strategy. The strategies described here are independent of the learning algorithms used. Preliminary experiments using different strategies and learning algorithms on two molecular biology sequence analysis data sets demonstrate encouraging results. Machine learning techniques are central to automated knowledge discovery systems and hence our approach can enhance the effectiveness of such systems.

The combination of different text representations and search strategies has become a standard technique for improving the effectiveness of information retrieval. combination, for example, has been studied extensively in the TREC evaluations and is the basis of the “meta-search” engines used on the Web. This paper examines the development of this technique, including both experimental results and the retrieval models that have been proposed as formal frameworks for combination. We show that combining approaches for information retrieval can be modeled as combining the outputs of multiple classifiers based on one or more representations, and that this simple model can provide explanations for many of the experimental results. We also show that this view of combination is very similar to the inference net model, and that a new approach to retrieval based on language models supports combination and can be integrated with the inference net model.

This paper describes Seeker, a platform for large-scale text analytics, and SemTag, an application written on the platform to perform automated semantic tagging of large corpora. We apply SemTag to a collection of approximately 264 million web pages, and generate approximately 434 million automatically disambiguated semantic tags, published to the web as a label bureau providing metadata regarding the 434 million annotations. To our knowledge, this is the largest scale semantic tagging effort to date. We describe the Seeker platform, discuss the architecture of the SemTag application, describe a new disambiguation algorithm specialized to support ontological disambiguation of large-scale data, evaluate the algorithm, and present our final results with information about acquiring and making use of the semantic tags. We argue that automated large scale semantic tagging of ambiguous content can bootstrap and accelerate the creation of the semantic web.

For many KDD applications, such as detecting criminal activities in E-commerce, finding the rare instances or the outliers, can be more interesting than finding the common patterns. Existing work in outlier detection regards being an outlier as a binary property. In this paper, we contend that for many scenarios, it is more meaningful to assign to each object a degree of being an outlier. This degree is called the local outlier factor (LOF) of an object. It is local in that the degree depends on how isolated the object is with respect to the surrounding neighborhood. We give a detailed formal analysis showing that LOF enjoys many desirable properties. Using real world datasets, we demonstrate that LOF can be used to find outliers which appear to be meaningful, but can otherwise not be identified with existing approaches. Finally, a careful performance evaluation of our algorithm confirms we show that our approach of finding local outliers can be practical.

CF-based web service recommender systems employ the predicted QoS mainly in two ways. 1) When users query a service with specific functionality, the one with the best predicted QoS is recommended to them. 2) Top-k best-performing services are recommended to help users discover potential services. While this kind of recommendation is useful, it is not obvious to users why certain services are recommended. More than a service list ranked by predicted QoS as recommendation, we need to develop an exploratory recommendation tool that provides valuable insight into the QoS space and enables an improved understanding of the overall performance of web services. The QoS space visualization of all web services on a map will reveal the rationale behind QoS-based service recommendations. QoS space visualization is more than a picture or method of computing. It transforms the information of high dimensional QoS data into a visual form enabling service users to observe, browse, and understand the information. To draw the QoS map by dimension reduction step and map creation step. Create a 2D representation of the high-dimensional QoS space by using self-organizing map (SOM), and each web service is mapped to a unique 2D coordinates.

## III. PROPOSED WORK

This paper is intended to provide an insight of the annotation techniques and application of few techniques to provide the required results with the above stated advantages. A novel collaborative filtering algorithm designed for large-scale web service recommendation. This collaborative filtering algorithm uses a rule-based mechanism to determine behavior consistent information based control strategies for route guidance in a dynamic vehicular traffic system. This approach employs the characteristic of QoS and achieves considerable improvement on the recommendation accuracy.

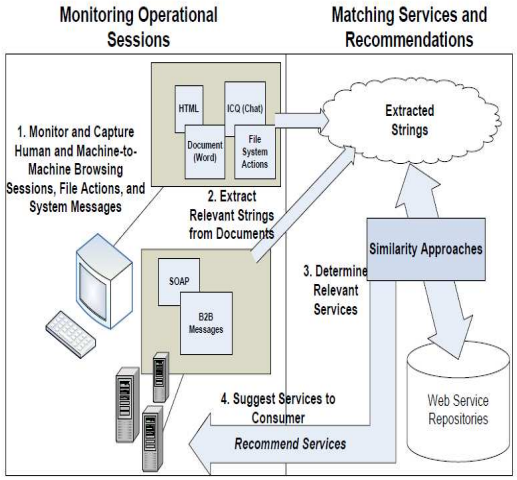


Figure 2: Web service recommendation

In the presence of multiple Web services with identical or similar functionalities, Quality of Service (QoS) provides non-functional Web service characteristics for the optimal Web service selection. Since the service providers may not deliver the QoS it declared, and some QoS properties (e.g., network latency, invocation failure-rate, etc.) are highly related to the locations and network conditions of the service users, Web service evaluation by the service users can obtain more accurate results on whether the demanded. The service recommender system component is answering the global user query. The latter has to be split local queries (i.e., sub-queries) and has to determine which peer is able to solve a local query. Each sub-query is expressed in SQL. SRS handles a Local Query Processing Engine component. Then, it carries out all the interactions between the composed services and generates a set of composition plans to provide the requested data.

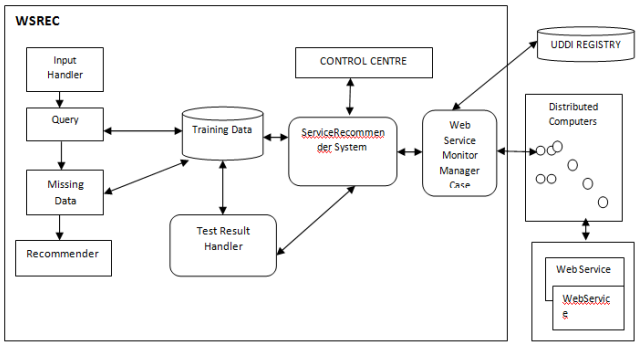


Figure 3- Proposed System Architecture

The basic function of this architecture is to obtain sufficient Web service QoS information from different service users is crucial for making accurate Web service recommendations. The idea is that by contributing the individually observed Web service QoS information to Web Service Recommender System, the service users can obtain accurate Web service recommendation service. Apart from the user contribution mechanism, Web Service Recommender System also controls a number of distributed computers for monitoring the publicly available Web services.

The system architecture of Web Service Recommender System, which includes the following procedures:

1) An active service user provides the individually obtained Web service QoS information to the Web Service Recommender System;

2) The Input Handler in the Web Service Recommender System processes the input data;

3) The Find Similar Users finds similar users from the training data of Web Service Recommender System.

4) The Predict Missing Data predicts the missing QoS values for the active user using collaborative filtering algorithm and saves the predicted values.

5) The Recommender employs the predicted QoS values to recommend optimal Web services to the active user.

## IV. SIMULATION WORKS/RESULTS

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.

QoS Value Prediction:

After the phase of region aggregation, thousands of users are clustered into a certain number of regions based on their physical locations and historical QoS similarities. The service experience of users in a region is represented by the region center. With the compressed QoS data, searching neighbors and making predictions for an active user can be computed quickly. Traditionally, the QoS prediction meth-ods need to search the entire data set, which is rather inefficient. In our approach, similarity between the active user and users of a region is computed by the similarity between the active user and the region center. Moreover, it is more reasonable to predict the QoS value for active users based on their regions, for users in the same region are more likely to have similar QoS experience on the same web service, especially on those region-sensitive ones.

User-collaboration Idea:

The basic idea of our approach is that users closely located with each other are more likely to have similar service experience than those who live far away from each other. Inspired by the success of Web 2.0 websites that emphasize information sharing, collaboration, and interaction, we employ the idea of user-collaboration in our web service recommender system. Different from sharing in-formation or knowledge on blogs or wikis, users are encouraged to share their observed web service QoS performance with others in our recommender system. The more QoS information the user contributes, the more accurate service recommendations the user can obtain, since more user characteristics can be analyzed from the user contributed information

Time Complexity Analysis:

The time complexity is calculating the median and MAD of each service. Form services, the time complexity. With MAD and median, we identify the region-sensitive services from the service perspective. Since there are most of ‘n’ records for each service, the time complexity of each service. Therefore, the total time complexity of the region-sensitive service identification.

Control centre:

Extending the previous work a control centre is designed for dynamic vehicular route choice system. It helps user to choose a trustworthy services and also provides the details about the services. It ensures the user by updating the database and guiding the user to select the Quality-Aware services when difficulties arise. It reduces the time complexity by providing an optimal value for QoS-Aware Services. Control center helps the database administrator updates the service conditions in regular intervals.

## V. CONCLUSION

An innovative approach for web service recommendation is designed, which significantly improves the prediction accuracy to provide Quality-Aware Service. Also a novel collaborative filtering algorithm designed for large-scale web service recommendation. Web service recommender systems work by collecting user observed QoS records of different web services and matching together users this approach employs the characteristic of QoS and achieves considerable improvement on the recommendation accuracy. Control Centre helps the user by updating the database and guiding the user to select the Quality-Aware services when difficulties arise. It also reduces the time complexity by providing an optimal value which ensures QoS-Aware Services. Our experimental result showed that our proposed novel technique works efficiently when compared to previous methods.

## VI. 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, ://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.

[8] T. Kohonen, “The Self-Organizing Map,” Proc. IEEE, vol. 78, no. 9, pp. 1464-1480, Sept. 1990.

[9] S. Kaski, J. Venna, and T. Kohonen, “Coloring that Reveals High-Dimensional Structures in Data,” Proc. Sixth Int’l Conf. Neural Information Processing, vol. 2, pp. 729-734, 1999.

[10] J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordan, and J. Riedl, “GroupLens: Applying Collaborative Filtering to Usenet News,” Comm. ACM, vol. 40, no. 3, pp. 77-87, 1997.

[11] G. Linden, B. Smith, and J. York, “Amazon.com Recommendations: Item-to-Item Collaborative Filtering,” IEEE Internet Computing, vol. 7, no. 1, pp. 76-80, Jan./Feb. 2003.
[12] Z. Maamar, S.K. Mostefaoui, and Q.H. Mahmoud, “Context for Personalized Web Services,” Proc. 38th Ann. Hawaii Int’l Conf., pp. 166b-166b, 2005.
[13] M.R. McLaughlin and J.L. Herlocker, “A Collaborative Filtering Algorithm and Evaluation Metric That Accurately Model the User Experience,” Proc. Ann. Int’l ACM SIGIR Conf., pp. 329-336, 2004.
[14] B. Mehta, C. Niederee, A. Stewart, C. Muscogiuri, and E.J. Neuhold, “An Architecture for Recommendation Based Service Mediation,” Semantics of a Networked World, vol. 3226, pp. 250-262, 2004.
[15] B.N. Miller, I. Albert, S.K. Lam, J.A. Konstan, and J. Riedl, “MovieLens Unplugged: Experiences with an Occasionally Connected Recommender System,” Proc. ACM Int’l Conf. Intelligent User Interfaces, pp. 263-266, 2003.
[16] C.D. Mining, P. Raghavan, and H. Schu¨ tze, An Introduction to Information Retrieval. Cambridge Univ., 2009.
[17] C. Zhao, C. Ma, J. Zhang, J. Zhang, L. Yi, and X. Mao, “HyperService: Linking and Exploring Services on the Web,” Proc. Int’l Conf. Web Services, pp. 17-24, 2010.
[18] E. Rich, “User Modeling via Stereotypes,” Cognitive Science, vol. 3, no. 4, pp. 329-354, 1979.
[19] W. Rong, K. Liu, and L. Liang, “Personalized Web Service Ranking via User Group Combining Association Rule,” Proc. Int’l Conf. Web Services, pp. 445-452, 2009.
[20] L. Shao, J. Zhang, Y. Wei, J. Zhao, B. Xie, and H. Mei, “Personalized QoS Prediction for Web Services via Collaborative Filtering,” Proc. Int’l Conf. Web Services, pp. 439-446, 2007.