Recommending a Credible Web Service

Jiaxiang Ge, Bin Xu, Qi Li, Yongbin Liu
Department of Computer Science and Technology
Tsinghua National Laboratory for Information Science and Technology
Tsinghua University
Beijing, China
gerjiaxiang@gmail.com, xubin@tsinghua.edu.cn, zhongguoliqi@163.com, yongbinliu03@gmail.com

***Abstract*—The explosive growth of Internet Web services has intensified the study of Service-oriented Computing (SOC). The recommendation of a Web service has become extremely common. Current approaches utilize the Quality of Service (QoS), which is based on the performance of the Web service. However, acquiring average QoS values is impractical because these values are susceptible to the uncertain Internet environment. Additionally, the performance information of a Web service does not provide the discriminative power to the user when selecting identical service candidates. In this paper, we propose Creditability of a Web service (CoS) to recommend a credible Web service. A credible Web service satisfies three features: *popularity*, *influence* and *authority*. We measure CoS by aggregating Web services together to form a Web service society. Then, we capture the social information for the Web service and calculate the behavior of the Web services. We use about 30,000 Web services from ProgrammableWeb, WSDream and Seekda in the experiments. Our approach effectively recommends credible Web services and displays a better discriminative power.**

***Keywords*-Web services recommendation; creditability of Web services;**

I. Introduction

The purpose of recommendation system is to predict Web services that may interest a user. The Quality of Service (QoS) is widely employed to represent the non-functional characteristics of a Web service and has been considered as a key factor in recommending a Web service [1]. However, values of some properties (e.g., response time) must be measured on the client-side [2]. Acquiring average QoS values is impractical because these values are susceptible to the uncertain Internet environment and to conduct real world Web service invocations is time-consuming and resource-consuming.

We investigated the basis of Web service selection in previous work [3]. The Creditability of Web Services (CoS) represents the reputation of a service within a service society. In this paper, we propose the CoS to recommend a credible Web service. We measure the CoS by aggregating Web services together to form a Web service society. Then, we capture the social information of the Web services and calculate the behaviors of the Web services. The main contributions are:

- We propose the CoS for Web service recommendation. This process maintains a discriminative power, allowing the user to make a decision when selecting a service from identical service candidates.
- We conduct comprehensive experiments to evaluate our approach by employing a real-world data set of Web services. These data set included more than 30,000 Web services including WSDLs, APIs and mashups type Web services.

The remainder of this paper is organized as follows: Section 2 presents the definition of the Web service society and the CoS. Section 3 presents our approach. Section 4 describes the application of the recommendation system in a service search engine. Section 5 shows our experiment and Section 6 concludes the paper.

II. Service Society and Creditability of a Web Service

A service society is a group of services involved in persistent service relationships or a large social grouping sharing identical functional or non-functional details. Service societies are characterized by patterns of service social relations between individuals sharing distinctive properties. In this paper, we construct two types of service societies: Linked-Data and API. The CoS represents the reputation of a service within a service society. A good creditability is a commendation or honor given to a service that provides a trustworthy basis for the user when selecting a service.

III. The Aggregation Techniques for a Service Society

A. Semantic-based Approach in a Linked-Data Society

1) Word Similarity: By preprocessing the data, we found that several character sets are terminology words or acronyms. These character sets are referred to as terms, and the others are referred to as words. Terms and words are two disjointed sets. We adopted WordNet and Edit Distance to calculate the similarity of words. Because the terms are absent from the lexical database of WordNet, we address the calculation of word similarity using Edit Distance. When calculating the similarity between a word and term, we

978-1-4673-7272-5/15 $31.00 © 2015 IEEE
DOI 10.1109/ICWS.2015.101

IEEE computer society

Authorized licensed use limited to: Tsinghua University. Downloaded on October 09,2026 at 01:56:08 UTC from IEEE Xplore. Restrictions apply.

assume the similarity is zero. We define the function of calculating the Word Similarity, $wSim$ as

$$wSim(x,y) = if \begin{cases} WordNet & x \in word \& y \in word \\ 0 & x \in word \& y \in term \\ EditDistance & x \in term \& y \in term \end{cases}$$

2) Operation Match Degree: Let V_1 and V_2 be the word vector of parameters of operations op_1 and op_2, respectively. Formula (1) defines the Operation Match Degree $opMatch$.

$$opMatch(op_1, op_2) = \frac{wSim(V_1, V_2)}{|V_1||V_2|} \quad (1)$$

B. Social Network based Approach in an API Society

A combination of mapping APIs represents two or more APIs cooperating in a mashup. We define the APIs mapped in the mashup as a *composition-service network (CSN)*, and formalize the CSN as a $m \times n$ matrix, where m is the number of compositions and n is the number of services [4]. CSN is formalized as

$$CSN = [CS_{ij}], 0 \leq i \leq m, 0 \leq j \leq n \quad (2)$$

where $CS_{ij} = 1$ when S_i invokes service S_j. Because only 14% of APIs were invoked in the mashup ecosystem (Table I), we enrich the density of a service society through exploiting tag information. Therefore, the condition of $CS_{ij} = 1$ is formed when the tag information of S_i is similar to S_j. We define an additional network: a *service-service network (SSN)*, in which two services are connected when they appear in the identical composition. The SSN is formalized as

$$SSN = [SS_{ij}] = CSN^T \cdot CSN, 0 \leq i, j \leq n \quad (3)$$

where SS_{ij} is the number of compositions in which service S_i and service S_j are invoked, and SS_{ii} is the number of composition in which service S_i is invoked. We calculate the SSN using the Perfect Match model [3] in our data set (Table I).

C. Measuring the Creditability of a Web Service

Measuring the CoS using service activities and relationship factors is inspired by a human society, in which service activities represent social actions. Service relationships represent social interactions including any relationship between two or more individuals. Additional statements of service activities and relationship have been presented in previous work [3]. In this paper, we present the *popularity*, *influence* and *authority* of a service in a certain service society.

1) Popularity: The *popularity* of a Web service is a social phenomenon that dictates the service that is best liked. The more the service is invoked or composed with an additional service, the more popular the service will then be considered. This index will assist users in identifying the famous services quickly (e.g., $GoogleMaps$, $Facebook$ and $del.icio.us$). In the network of a service society, the in-degree is a count of the number of ties directed to the node, and the out-degree is the number of ties that the node directs to others. We use the sum of the in-degree and out-degree as the $popularity$ of a service in direct or indirect networks. Formula (4) defines the $popularity$ of a service. We represent the in-degree and out-degree as $auth$ (authority) and hub, respectively.

$$popularity = auth + hub \quad (4)$$

We formalize the calculation of the authority/hub score [5] of each node as

$$auth(S_i) = \sum_{j=1}^{n} hub(S_j) \quad (5)$$

where n is the total number of nodes connected to service S_i, and S_j is a node connected to service S_i. Formula (5) describes the authority score of a node. This score is the sum hub scores of all nodes that point to the initial node.

$$hub(S_i) = \sum_{j=1}^{n} auth(S_j) \quad (6)$$

where n is the total number of nodes connected to service S_i, and S_j is a node connected to service S_i. Formula (6) describes the hub score as the summation of the authority scores of all its linking nodes.

2) Influence: The *influence* of a Web service indicates the importance of the role of a service in a service society. In a service society, each service has activities and relationships that result in the position of a service. The service relationship measures the $influence$ of a service in a service society. The eigenvector centrality is a measure of the influence of a node in a society [6]. We use the eigenvector centrality to measure the most influential service(s) within a service society. Furthermore, calculated service relationships are also used to indicate the number of times a service acts as a bridge along the shortest path between two other services, referred to as the betweenness. Formula (7) defines the $influence$ of a service. We represent the eigenvector centrality and betweenness as eig and $betw$, respectively.

$$influence = eig + betw \quad (7)$$

The eigenvector centrality is a relative score that is recursively defined as a function of the number and strength of connections to its neighbors and the centralities of those neighbors. Let A denote the adjacency matrix of a service society, and let $\Gamma(S_i)$ denote the neighborhood of S_i and the

Authorized licensed use limited to: Tsinghua University. Downloaded on October 09,2026 at 01:56:08 UTC from IEEE Xplore. Restrictions apply.

set of nodes that S_i is connected to directly. The eigenvector centrality of S_i is calculated as

$$\begin{aligned} eig(S_i) &= \frac{1}{\lambda}\sum_{j\in\Gamma(S_i)} S_j \\ &= \frac{1}{\lambda}\sum_{j=1}^{n} A_{i,j}S_j \end{aligned} \tag{8}$$

where λ is a constant. Formula (8) can be rewritten as the vector formula (9) where $S_i = S_1, S_2, S_3, ..., S_n$ is the vector for the eigenvector centrality scores of all nodes. The vectorized representation is the following:

$$\begin{aligned} S_i &= \frac{1}{\lambda}AS_i \\ \lambda S_i &= AS_i \end{aligned} \tag{9}$$

where λ is an eigenvalue and S_i is the corresponding eigenvector of matrix A. Therefore, several eigenvector pairs are noted for the adjacency matrix A.

We formalize the calculation of the betweenness score [7] of each node as

$$betw(S_k) = \sum_{S_i \neq S_k \neq S_j \in S_n} \frac{\sigma_{S_iS_j}(S_k)}{\sigma_{S_iS_j}} \tag{10}$$

where $\sigma_{S_iS_j}$ and $\sigma_{S_jS_i}$ are the number of the shortest paths from S_i to S_j, and $\sigma_{S_iS_j}(S_k)$ is the number of shortest paths from S_i to S_j on which S_k resides.

3) Authority: The *authority* is the power of a service in the composition of a Web service. These services are either functionally powerful or a universally useful service such as $GoogleMaps$, $GeoSeearch$, and $zipcode$. In graph theory, the closeness centrality has been used to measure how long the spread information will take from one node to all other nodes sequentially. We use closeness to measure the capability of a service in the composition path length. We define the $authority$ of a service by combining the closeness and Formula (5). Formula (11) defines the $authority$ of a service. We represent the closeness as clo.

$$authority = clo + auth \tag{11}$$

We formalize the calculation of closeness score [7] of each node as

$$col(S_i) = \frac{1}{\sum_{S_j \in S_n} d_{SN}(S_i, S_j)} \tag{12}$$

where $S_{SN}(S_i, S_j)$ is the distance between service S_i and S_j in the service network SN, and S_n is the set of service.

IV. A Search Engine with Recommendation System

In our previous work [8], we proposed a multimodal-query search and mining service system—SMiner. To improve the customer experience, we investigate the recommender system and attempt to embed the system in SMiner. Figure 1 displays the system overview for the service search engine with a recommender system, which includes the following procedures:

- Web service users log on to the SMiner system through a Web browser and search for a Web service through the functions provided by SMiner.
- The recommender system aggregates the service through aggregation techniques for a service society (details will be introduced in Section 3.A and 3.B).
- The recommender system measures the CoS in three aspects (details will be introduced in Section 3.C).
- The recommender system develops a result list from the service search engine. The services in the result list satisfy the requirement of the users provided in the query.
- The user receives the result list and the CoS score for three aspects: $popularity$, $influence$ and $authority$.

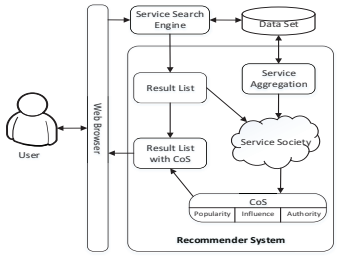

Figure 1. System Overview

V. Experiment

A. Data Set

Table I
Data Set Parameters

Number of WSDL services	12624
Number of service operations	118695
Number of mashup	7568
Number of APIs	11182
Number of Validated APIs	9654
Number of APIs invoked in the mashup	1366 (=14%)
Average number of service operations per WSDL	9.4
Average number of APIs per mashup	2.12

We obtain the service data set from ProgrammableWeb, Seekda and WSDream (http://www.wsdream.net). The data set includes 7568 mashups, 11182 APIs and 1,2624 WSDL services (Table I).

B. Experimental Results

Table II shows the ranking of the popular APIs provided by ProgrammbleWeb. We show the scores for the CoS, $popularity$, $influence$ and $authority$ of the APIs to observe the relationships between these scores. The rankings allow users to quickly evaluate complex information and are applicable in Web services search engines. However, when a user selects a service they should compare the

Authorized licensed use limited to: Tsinghua University. Downloaded on October 09,2026 at 01:56:08 UTC from IEEE Xplore. Restrictions apply.

Table II
RANKING POPULAR APIS

APIs Name	Ranking					CoS	Popularity	Influence	Authority
	PW	CoS	Popularity	Influence	Authority				
Google Maps	1	5	10	5	2	0.0003664410	0.0001919975	0.0000441357	0.0010776512
Twitter	2	6	4	6	4	0.0002736660	0.0006143922	0.0000322423	0.0009787678
YouTube	3	7	9	10	1	0.0000379253	0.0002303970	0.0000000067	0.0012380594
Flickr	4	1	2	1	9	0.0446924853	0.0019583749	0.0059117783	0.0008508304
Facebook	5	4	3	4	8	0.0003670772	0.0007679902	0.0000448586	0.0009141754
Amazon Product Adverstising	6	8	4	8	4	0.0000319747	0.0006143922	0.0000000886	0.0009787678
Twilio	7	10	8	9	3	0.0000305149	0.0002687965	0.0000000642	0.0009790428
Last.fm	8	3	6	3	7	0.0010639253	0.0005375931	0.0001370280	0.0009319038
eBay	9	2	1	2	10	0.0105246074	0.0026495662	0.0013894883	0.0008486227
Twilio SMS	10	9	7	7	6	0.0000314237	0.0003839951	0.0000003427	0.0009393019

scores in the identical category. For example, the CoS, $popularity$ and $influence$ of Facebook is higher than Twitter, but the $authority$ of Facebook is lower than Twitter in the social society service category in our data set. In conclusion, the CoS considered various factors and displays a better discriminative power. The $popularity$, $influence$ and $authority$ of a service provides various perspective to judge a service. The judgment mostly considers the service social information within a service society.

VI. RELATED WORK

The aim of Semantic-based recommendation approach is to determine a Web service which suits the requirement of users and displays the highest matching degree via semantic similarity computation [9]. The main challenge of this approach is to clean and retain useful information from noisy descriptions. Social network based approaches provide a clear way of analyzing the structure of entire social entities [10]. The main challenge of this approach is defining, managing and analyzing the social network model. Furthermore, multiple studies utilize QoS to investigate Web service recommendation because QoS information represents non-functional characteristics of Web services [11]. However, acquiring and analyzing QoS information remains difficult because the data is susceptible to the provider, user and Internet environment of a Web service. In this paper, we utilized semantic and social network based approaches to aggregate services together and form a service society to analyze the social information of Web services.

VII. CONCLUSION

This paper presents a Web service recommendation based on the Creditability of a Web Service (CoS). We have constructed two types of service societies: Linked-Data integration and a collaboration of APIs in a mashup ecosystem. We find that our approach effectively recommend credible services in two types of service societies and facilitate the implementation of Web service recommendation in search engines.

ACKNOWLEDGMENT

This work is supported by the Chinese National Science Foundation under grant No.61170212 and the Chinese National High-Tech Project (863) under grant No. 2013AA01A607. The Beijing Key Lab of Networked Multimedia also supports our research.

REFERENCES

[1] Yan Hu, Qimin Peng, and Xiaohui Hu. A time-aware and data sparsity tolerant approach for web service recommendation. In *Web Services (ICWS), 2014 IEEE International Conference on*, pages 33–40. IEEE, 2014.

[2] Niko Thio and Shanika Karunasekera. Automatic measurement of a qos metric for web service recommendation. In *Software Engineering Conference, 2005. Proceedings. 2005 Australian*, pages 202–211. IEEE, 2005.

[3] Bin Xu, Jiaxiang Ge, and Tonglee Chung. Measuring credit of web service. In *Web Services (ICWS), 2014 IEEE International Conference on*, pages 423–430. IEEE, 2014.

[4] Wei Tan, Jia Zhang, and Ian T. Foster. Network analysis of scientific workflows: A gateway to reuse. *IEEE Computer*, pages 54–61, 2010.

[5] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *JOURNAL OF THE ACM*, 46(5):604–632, 1999.

[6] Phillip Bonacich and Paulette Lloyd. Eigenvector-like measures of centrality for asymmetric relations. *Social Networks*, pages 191–201, 2001.

[7] Ulrik Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25:163–177, 2001.

[8] Bin Xu, Sen Luo, and Kewu Sun. Towards multimodal query in web service search. In *Web Services (ICWS), 2012 IEEE 19th International Conference on*, pages 272–279, June 2012.

[9] Mustapha Aznag, Mohamed Quafafou, Nicolas Durand, and Zahi Jarir. Web services discovery and recommendation based on information extraction and symbolic reputation. *arXiv preprint arXiv:1304.3268*, 2013.

[10] Buqing Cao, Jianxun Liu, Mingdong Tang, Zibin Zheng, and Guangrong Wang. Mashup service recommendation based on user interest and social network. In *Web Services (ICWS), 2013 IEEE 20th International Conference on*, pages 99–106. IEEE, 2013.

[11] Xi Chen, Zibin Zheng, Qi Yu, and M Lyu. Web service recommendation via exploiting location and qos information. 2013.

Authorized licensed use limited to: Tsinghua University. Downloaded on October 09,2026 at 01:56:08 UTC from IEEE Xplore. Restrictions apply.