S2Net: A New Service Description Model based on Heterogeneous Service Network

Peng Li, Junhao Wen, Junping Dong
College of Computer Science, Chongqing University, Chongqing, China
Email: {pengli, jhwen, dongjunping}@cqu.edu.cn

Abstract— **In service computing, with the development of more and more published services, the network relationships between services, providers and requesters have become more and more complex. The traditional WSDL based service description methods only give the functional expression but ignore the potential relations between service participants. In this paper, based on the relations between services, provides and requesters in service application context, we propose a heterogeneous service network based service description model S2Net which provides service network extraction and cached considered maintenance algorithms. The experiments on real service dataset show the efficiency and effectiveness of our approach.**

Index Terms— **Service Description, Heterogeneous Service Network, Network Analysis**

I. Introduction

With the development of service computing, especially the publications of Web Services standard and related policies, more and more services based on WSDL have been published in the Internet. In services application, service request and service provide operations are happening every day, the service context is becoming more and more complex.

In the applications of service discovery, service composition or service selection, how to describe the relation links of service context is the key process to be considered carefully. The conventional WSDL based simple service description method can give a detail information on service function [1], but the context based service information, such as service relations, cannot find a suitable method to describe. With the creation of semantic Web [2], ontology based service description can give more details about service [3], but the shortage of domain ontology slows down the development of this kind of method.

In today's service application environment, interactions between services, service providers and requesters have become so frequent that every day there are large numbers of services are published and requested, the relations between service participants are changing from time to time. The network liked relations are useful for knowledge mining on service or any other advanced applications.

In this paper, focusing on the problems of nowadays service descriptions, we propose a heterogeneous service network based service description model S2Net which provides a service network extraction method and a cache based network maintenance approach. The main contributions in this paper are shown as follows:

(1) We propose a heterogeneous service network description model S2Net which can extract service network based on two processes: nodes extraction and links extraction;

(2) We propose a cache based network maintenance model which can monitor and update the mapped network when nodes or links changed;

(3) The experiments on Titan service set show the time cost of network extraction and maintenance.

The rest of paper is organized as follows: Section 2 is the Related Works, Section 3 is the Concepts and Definitions, Section 4 is our main part of S2Net Model algorithms, Section 5 is some Research Suggestions and Section 6 is our experiment, the last section is conclusion.

II. Related Works

In recent years, many researches on service network have been proposed. In this section, we will give some representative related works. In literature [4], Hui proposed social network and semantic Web based service network model which considers services as network nodes and relations between services as network links, this kind of service network plays a role of bridge between service providers and consumers. John proposed a service network model based on complex network and network analysis which consider the feature of small world network [5]. Based on the theory of discrete mathematics, a new service network model based on graph theory was proposed which can supports service composition and service discovery using network analysis technology [6] [7].

In information network related researches, the concept of heterogeneous information network has been proposed such as [8] and [9], authors give the definition of heterogeneous information network to denote the network structure which contains more than one kind of node in a network. This kind of heterogeneous network gives a new idea on service network description which contains three types of nodes: services, providers and requesters.

III. Concepts and Definitions

In this section, we will give the related concepts and definitions about our proposed S2Net service network

This work is supported by China Natural Science Foundation Project No. 61075053 and No. 71102065, supported by Natural Science Foundation Project of CQ CSTC2010BB2244 and the Fundamental Research Funds for the Central Universities Project No. CDJZR10090001.

doi:10.4304/jsw.8.2.404-409

model. We propose two kinds of service networks, simple Service Network and Heterogeneous Service Network, the definitions are shown as follows.

A. Service Network

Definition 1.**Service Network**.We define $SN =< S, E, W >$ as a service network, where S is the set of service nodes and $S = \{s_1, s_2, ..., s_n\}$ with the service number of n; E denotes the links between service nodes, if any two service s_i and s_j has a link , there is an edge e between them where $e =< s_i, s_j >\in E$; W is the weight matrix of service nodes which denotes the adjacency matrix of each two services.

Service Network defined in Definition 1 is one kind of homogeneous network which only contains one type of nodes, the service type. However, in the complex service environment, services, providers and requesters are linked together by the usage of services. In order to description the different kind of network, we define a new kind of network which we called heterogeneous service network in Definition 2.

B. Heterogeneous Service Network

Definition 2.**Heterogeneous Service Network**. We define $HSN =< H, E, W >$ to denote a heterogeneous service network, where $H = S \cup P \cup R$ is the node set includes nodes belonged to Service type (S), Provider type (P) and Requester type (R). E is the set of edges which links between Service and Provider or Service and Requester, in E if there is an edge between any two nodes in H, there is an $e =< h_i, h_j >\in E$. W is the weight matrix of all the types of network nodes, in H, if there is a link between any two nodes h_i and h_j , $W(h_i, h_j) = 1$, or $W(h_i, h_j) = 0$.

For convenience, we will use HSN on behalf of the heterogeneous service network in the following statements. $H(HSN)$ denotes the nodes set of a HSN network, $E(HSN)$ is the edges set of the network HSN and $W(HSN)$ on behalf of HSN's weight matrix. We use uppercase S, P and R to represent the nodes set of Services, Provided and Requesters, we user lower case letters s, p and r to on behalf of the three types of node instance belonged to S, P and R.

Something we should note is that relationship may exist between the three kinds of nodes in a heterogeneous network of services which are "Provide-Provided" relations between providers and services, "Request-Requested" relationship between requesters and services, "Call-Called" relationship exists between services and services who call other services in their business processes, In our research we only consider the first two relationships between different types of nodes, for the service-service call , we don't consider within the scope of this paper discussion.

In order give an easy understanding of our HSN definition, we use the following example of Java service.

Example 1.Java Service Network

Assume that there are three Java coffee-related services and three Java programming language services which are released by the developers of the two different procedures within a certain time and requested by four different service requesters from this two areas(Java coffee and Java program), the Java heterogeneous service network is shown in Figure 1 .

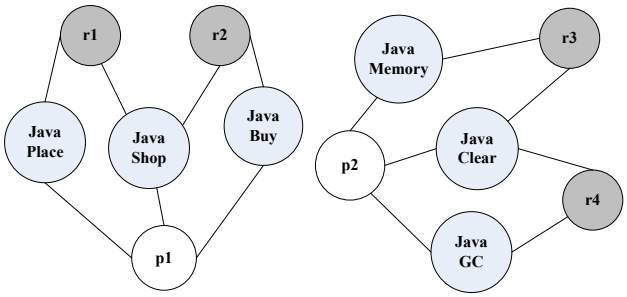

Figure 1. Java Service Network Example

As shown in Figure 1, in our Java service network example, the network is a heterogeneous service network which nodes include six services, two providers and four requesters. In the network, service provider p1 provides JavaPlace, JavaShop and JavaBuy three Java coffee related services, p2 provides Java Memory, JavaGC and JavaClear three Java language related services. From the service request records we can see the service requester r1 and r2 mainly like coffee services while service requester r3 and r4 are more interested in the Java language related services.

IV. S2Net Model: Framework and Algorithms

In this section, we will give the details of S2Net model. In our S2Net heterogeneous service network model, there are two key processes should to be considered, the first one is the service to network extraction process and the second process is the maintenance of the service network. In following subsections, we will give the related algorithms in each process.

A. Service Network Extraction

Service network extraction is the process of extracting heterogeneous service network from service application context. The extracted service network contains nodes of services, providers, requesters and the links between nodes which include two types of relations "Provide-Provided" and "Request-Requested".

In the process of service network extraction, there are three main steps: Nodes Extraction, Links Extraction and Network Creation, the main framework of network extraction is shown as Figure 2.

As shown in Figure 2, $provider1$ and $provider2$ are two service providers, $requester1$ and $requester2$ are two service requesters, $s1$, $s2$, $s3$, $s4$, $s5$ are available services. In the right-bottom of Figure 2, we get the extracted service network, in the network, we use $p1$, $p2$ to repress $provider1$ and $provider2$, $r1$ and $r2$ to denote $requester1$ and $requester2$. From the network extraction framework, we can clearly see the main three

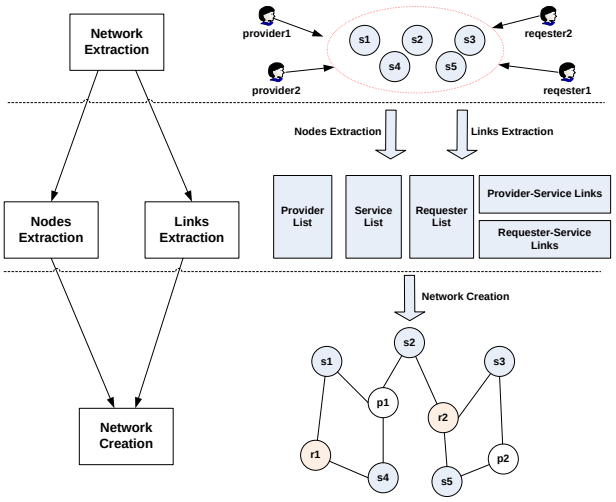

Figure 2. Service Network Extraction Framework

steps of extraction, in following parts, we will give the details for each step.

1) Nodes Extraction: The first step in Network Extraction is Nodes Extraction which is the process of transfer from services context to network nodes of services, service providers and requesters, the extracted nodes are saved in form of node lists. For example, $ProviderList$ is the nodes list for providers which stores the providers' information in services application context. The details of node lists are defined as follows:

$ProviderList = p_1, p_2, , p_(|P|)$, p_1 denotes one provider instance object, $|P|$ is the number of providers; $ServiceList = s_1, s_2, , s_(|S|)$, s_1 is one service object and $|S|$ is the total number of services; $RequesterList = r_1, r_2, , r_(|R|)$, r_1 is one of the requesters, $|R|$ denotes the numbers of requesters.

2) Links Extraction: After Nodes Extraction, the next step is Links Extraction which get the relations from services, providers and requesters application environment. In the aimed heterogeneous service network, the relations between nodes are called service links which repress the interconnection between two types of nodes in network. As shown in Figure 2, after extraction process, the relations between services and providers, services and requesters are saved as two link lists: Provider-Service Links and Requester-Service Links, the definition details are shown as follows:

$ProviderService = \{e_{ij}\}, i \in [1, |P|] and j \in [1, |S|]$

Where e_{ij} is one link between any one service provider p_i and any one service s_j, if there is a provide relation between them $e_{ij} = 1$,otherwise $e_{ij} = 0$.

Similarly, for requesters and services relations, we define another link list:

$RequesterService = \{e_{ij}\}, i \in [1, |R|] and j \in [1, |S|]$

Where e_{ij} is one link between any one service requester p_i and any one service s_j , if there is a request relation between them $e_{ij} = 1$,otherwise $e_{ij} = 0$.

3) Network Creation: After extraction of network nodes and links, in the next step, we can create the heterogeneous service network as defined in Definition 2. The output node lists of nodes extraction and link lists from links extraction are the input of network building as shown as Algorithm 1:

Algorithm 1: NetworkCreate

Input: ServiceList, ProviderList, RequesterList, ProviderServiceLinks, RequesterServiceLinks
Output: Heterogeneous Service Network $HSN =< H, E, W >$

```
1  HSN =< H, E, W > ;
2  H = S ∪ P ∪ R ;
3  for i = 0 to ServiceList.count do
4      s = ServiceList[i];
5      S.Add(s);
6      for j = 0 to ProviderList.count do
7          p = ProviderList[j];
8          P.Add(p);
9          if ProviderServiceLinks(p, s) is 1 then
10             e =< p, s >;
11             E.Add(e);
12             W(p, s) = 1;
13         end
14     end
15     for k = 0 to RequesterList.count do
16         r = ProviderList[k];
17         R.Add(r);
18         if RequesterServiceLinks(r, s) is 1 then
19             e =< r, s >;
20             E.Add(e);
21             W(r, s) = 1;
22         end
23     end
24 end
25 Return HSN;
```

In Algorithm 1, the final service network HSN are extracted from services application context which contains services, providers and requesters, the links in HSN are based on the real relations of service provide-provided and service request-requested between three types of nodes.

The time cost of our network extraction algorithm is mainly based on the nodes and links numbers. Suppose in our network, there are N1 number of services, N2 providers and N3 requesters, there are N4 times of provide operations happened and N5 for request operations, the whole time complexity is $O(N1(N2 + N4) + N1(N3 + N5))$, where indeed, the services number N1 is same to provide relation number N4, so the final complexity is $O(N1(N1 + N2 + N3 + N5))$.

B. Service Network Maintenance

In service application environment, the service context are changing from time to time, new services will be added, old services may be deleted, new operation of service provide and request will happen in any time. Faced to the multi-change service application context, service

network need to provide update or maintenance model to catch up with the real changes of services and relations.

In this section, we provide our service network maintenance model which considers the update on network nodes and network links, the algorithms of two kinds of maintenances will be given in follows.

1) Nodes Maintenance: Nodes Maintenance is the process of update operations when nodes in network changed. Here, because we only express the relations between services, providers and requesters, we dont care the changes of contents. The main change types in our algorithm are node add and add delete, the detail of the algorithm is shown as Algorithm 2:

Algorithm 2: NodesMaintenance

Input: h, changed node; $ctype$: change type; $HSN =< H, E, W >$;
Output: Heterogeneous Service Network $HSN =< H, E, W >$

```
1  if ctype is deleted then
2      H.Delete(h);
3      for i = 0 to h.links.count do
4          l = h.links[i];
5          e =< l, h >;
6          E.Delete(e);
7          W(l, h) = 0;
8      end
9  end
10 if ctype is added then
11     H.Add(h);
12     p = h.Provider;
13     e =< p, h >;
14     E.Add(e);
15     W(p, h) = 1;
16 end
17 else
18 end
19 HSN.Update();
20 Return HSN;
```

In Algorithm 2, two kinds of nodes update are processed, in algorithm $h.links$ means the connected nodes linked to node h, if the node h is deleted, the links between the deleted node and linked nodes need to be deleted too.

2) Links Maintenance: In the maintenance of network links, the main change types we should consider are new publication of services or new request operation of services, the algorithm detail is shown as Algorithm 3:

In Algorithm 3, the new happened service provide or request operations can updated to the exist service network.

3) Cache based Maintenance Model: The above maintenance algorithms we proposed are real time update processes, however, when changes are happening in a high frequency, the frequent update operations will cost high computing resource, so, we provide a cache based maintenance model which first stores changes in cache list, then make the update operations in batch.

Algorithm 3: LinksMaintenance

Input: $operation$, changed operation; $ctype$: change type; $HSN =< H, E, W >$;
Output: Heterogeneous Service Network $HSN =< H, E, W >$

```
1  if ctype is provideoperation then
2      s = get service node from operation;
3      p = get provider node from operation;
4      e =< p, s >;
5      E.Add(e);
6      W(p, s) = 1;
7  end
8  if ctype is requestoperation then
9      s = get service node from operation;
10     r = get requester node from operation;
11     e =< r, s >;
12     E.Add(e);
13     W(r, s) = 1;
14 end
15 HSN.Update();
16 Return HSN;
```

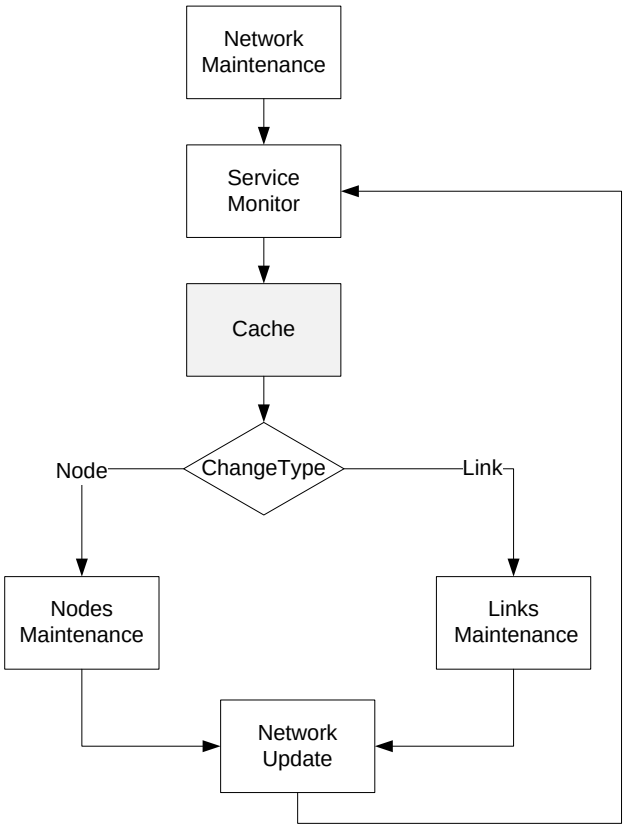

Figure 3. Cache based Network Maintenance Model

The main framework of cache based maintenance model is shown as Figure 3.

After considering the cache when maintain the changes of service network, the performance will be enhanced.

V. Research Suggestions

Network analysis based methods can help give a more clear understand of complex relationships between multiple interconnected objects.

After given the basic concepts and extraction model about heterogeneous service network, in this section we will give some advices on network based service computing research. We can use service network to describe service relations and then do some more higher levels of researches as follows:

(1) Service ranking based on heterogeneous service network analysis;

(2) Service recommendation based on service network ranking which consider heterogeneous feature of service network;

(3) Service discovery and composition research based on network analysis;

(4) Service clustering research based on service network partitioning.

Besides what we suggest above, service network based description can help on more research directions of service computing such as service selection, service matching or service retrieve.

VI. Experiment and Evaluation

In order to validate our service network model S2Net, we design experiments to give an evaluation on network extraction and maintenance processes.

A. Service Dataset

We use Titan dataset [10] [11] as the service set in experiments which collected real services records from Seekda web services search engine (http://webservices.seekda.com/). In order to construct the heterogeneous service network, we create random access to the service dataset by simulating 3,800 service requesters to request the existing services. After preprocessing on the dataset, we create our service dataset which includes 15,957 services, 4,428 service providers and 3,800 service requesters. All of our experiments are running in the Microsoft .Net Framework v4.0 version.

B. Service Network Extraction

In order to evaluate our network extraction algorithm, we design experiments to validate. In the first experiment, we run our S2Net network extraction process on the whole service set to create a HSN, one small part of the HSN is shown as Figure 4.

In the mapped service network, three service nodes StockQeote, RealTimeStock and HistoticalStock are provided by service providers strikeiron and btolcom, 138.238.250.155 is one of the seven service requesters which we denote it by IP address. As shown in Figure 4, the nodes of services, providers and requesters are linked together in form of heterogeneous service network.

In order to show the relations of time cost and service numbers, we design ten groups of experiments with

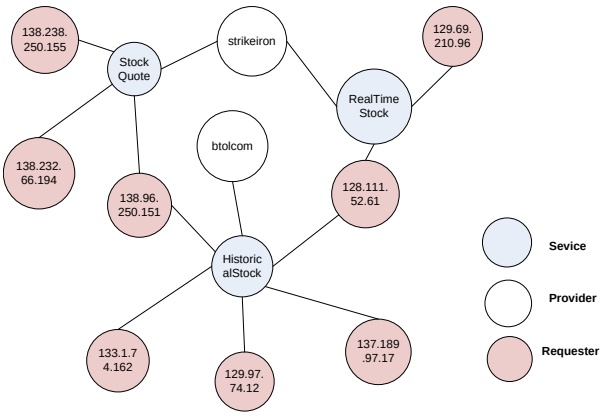

Figure 4. Part of HSN Network

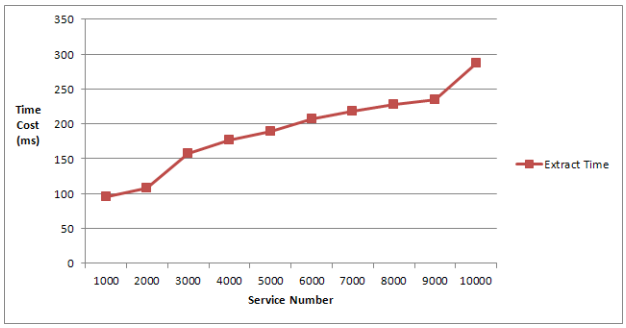

Figure 5. Time Cost of Network Extraction

different service numbers from 1000 to 10000, the time cost changes are shown as Figure 5.

As shown in Figure 5, with the increase of service numbers, the extraction process will cost more time, the change tendency is a near linear curved line.

C. Service Network Maintenance

In our service network maintenance experiment, we design four sub-experiments, the first one is design for nodes update without cache, the second one is for nodes update with cache, the third one is links update without cache and the last one is links update consider cache, the experiments are shown as Figure 6.

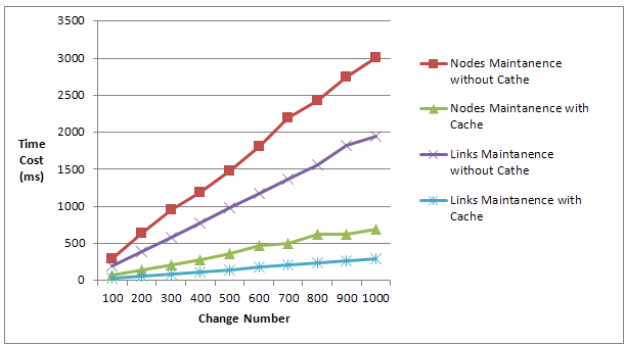

Figure 6. Time Cost of Network Maintenance

As shown in Figure 6, all the modes and links maintenance processes will cost more time with the increase of

service numbers, however, nodes maintenance and links maintenance process without cache will cost more time than the cache based process, the cache based service network maintenance model has a higher performance.

VII. Conclusion

In this paper, focus on the complex relations between services, service providers and requesters in service application context, we propose a new heterogeneous service network based description model S2Net which tries to find the potential relations between service participants with the help of network extraction algorithms and cache based maintenance model. In this paper, we open a new dimension of network based service description method which can enhance the development of higher service applications of service ranking, service composition and service discovery.

Acknowledgment

We would like to thank DKE group of the University of Queensland for help on this paper. This work is supported by China Natural Science Foundation Project No. 61075053 and No. 71102065, supported by Natural Science Foundation Project of CQ CSTC2010BB2244 and the Fundamental Research Funds for the Central Universities Project No. CDJZR10090001.

References

[1] F. Curbera, M. J. Duftler, R. Khalaf, W. Nagy, N. Mukhi, and S. Weerawarana, "Unraveling the web services web: An introduction to soap, wsdl, and uddi," *IEEE Internet Computing*, pp. 86–93, 2002.

[2] C.-H. Liu and J. J.-Y. Chen, "Using ontology-based bdi agent to dynamically customize workflow and bind semantic web service," *JSW*, pp. 884–894, 2012.

[3] J. Kopeck, T. Vitvar, C. Bournez, and J. Farrell, "Sawsdl: Semantic annotations for wsdl and xml schema," *IEEE Internet Computing*, pp. 60–67, 2007.

[4] H. Wang, Z. Feng, Y. Sui, and S. Chen, "Service network: An infrastructure of web services," in *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, vol. 3, nov. 2009, pp. 303 –308.

[5] J. Gekas and M. Fasli, "Service network structure analysis for web service discovery and composition," in *BASEWEB06*, may 2006.

[6] H.-h. E., M.-n. Song, J.-d. Song, Y. Li, and Z.-j. Ren, "The research of service network based on complex network," in *Service Sciences (ICSS), 2010 International Conference on*, may 2010, pp. 203 –207.

[7] J. Gekas, "Web service ranking in service networks," in *European Semantic Web Conference(ESWC 2006)*, June 2006.

[8] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu, "Rankclus: integrating clustering with ranking for heterogeneous information network analysis." in *EDBT'09*, 2009, pp. 565–576.

[9] Y. Sun, Y. Yu, and J. Han, "Ranking-based clustering of heterogeneous information networks with star network schema." in *KDD'09*, 2009, pp. 797–806.

[10] J. Wu, L. Chen, Y. Xie, and Z. Zheng, "Titan: a system for effective web service discovery," in *Proceedings of the 21st international conference companion on World Wide Web*, ser. WWW '12 Companion. New York, NY, USA: ACM, 2012, pp. 441–444.

[11] L. Chen, L. Hu, Z. Zheng, J. Wu, J. Yin, Y. Li, and S. Deng, "Wtcluster: Utilizing tags for web services clustering," in *Service-Oriented Computing*, ser. Lecture Notes in Computer Science, G. Kappel, Z. Maamar, and H. Motahari-Nezhad, Eds. Springer Berlin / Heidelberg, 2011, vol. 7084, pp. 204–218.

Peng Li now is a Ph.D. student of Chongqing University in college of computer science. He received his BS and MS degrees in computer science and technology from Chongqing University of Chongqing City in China, in 2007 and 2009, respectively.

He was a joint Ph.D. research student of University of Queensland in Queensland, Australia, from 2011 to 2012. The main research areas include service computing, data mining and network analysis.

Junhao Wen received his doctor degree in computer science from Chongqing University, China, in 2008.

He is currently professor and research leader of college of computer science and software engineering in Chongqing University. In recent years, Prof. Wen published more than twenty high level papers on service computing, data mining and software engineering.

Prof. Wen is the member of IEEE, chair of CCF YOCSEF of Chongqing.

Junping Dong is a postgraduate student of Chongqing University. She received her BS degree in software engineering from Chongqing University in 2012, her main research interests are service computing and data mining.