Social Network Analysis on the Interaction and Collaboration Behavior among Web Services

Shizhan Chen, Yuanbin Han, Zhiyong Feng
School of Computer Science and Technology, Tianjin University, Tianjin, 300072, China
shizhan@tju.edu.cn, ybhan09@gmail.com, zyfeng@tju.edu.cn

Abstract

Service-Oriented Computing (SOC) has received much interest due to its potential to tackle many adaptive system architecture issues that were previously hard to overcome by other computing paradigms. However, it has been facing great difficulty in quickly discovering and dynamically combing available Web services to satisfy given request on-demand. Most of the current researches concentrated on the semantic model for service discovery, composition, and so on. But there are few studies concerned the intrinsic pattern and law of the service interactions and relationships. To achieve the vision of SOC in heterogeneous and open environment, in our opinion, not only the semantics of individual Web service but also the interactions and relationships among Web services are needed to be considered seriously. In this paper, beginning with combining Semantic Web and social networking technology within SOC paradigm, we study associations between Web services, mine the relationships among services to design and build Service Network (SN), analyze the structural and social characteristics and complexity of SN to reveal the user interests, business requests, information and data flow and direction. In short, we would like to reassess and reconsider the SOC paradigm from the network perspective, through finding new knowledge to build new theoretical basis and approach which can be used to guide and promote the service discovery, composition, and so on, in SOC paradigm.

Introduction

Recently, Service-Oriented Computing (SOC) (Papazoglou and Georgakopoulos 2003; Papazoglou, Traverso et al. 2008) has become an emerging and promising paradigm, which utilizes functionalities provided by business applications and encapsulated within Web services, to build low-cost and flexible dynamic business processes within and across organizational boundaries and computing platforms even in highly distributed, open and heterogeneous environments.

There are great deal of activities and effects mainly concentrated on the semantic model for services discovery and dynamic service composition methods applying semantic Web, ontology and AI planning techniques (Dustdar and Schreiner 2005; Kaijun, Nong et al. 2011). However, it has been facing great difficulty in accurately discovering and automatically combing several available Web services to satisfy given goal on-the-fly. In our opinion, not only the semantics of individual Web service but also the interactions and associations among Web services are needed to be considered seriously to achieve the visionary promise of SOC. Unfortunately, there are few studies concerned the intrinsic pattern and law of the services relationships which may greatly promote and improve service computing. In this paper, beginning with combing semantic Web and social networking technology within SOC paradigm, we study the associations between Web services, and apply complex network analysis on the Web services ecosystem, named Service Network (SN) in our former work (Chen and Feng 2008; 2009; 2010), to explore the structural and social characteristics of SN.

The rest of this paper is organized as follows: Section 2 gives a brief review of service relationships and Service Network through semantic annotation and service mining. In Section 3, using graph theory and complex network analysis techniques, we continue with a detailed experiential study of the topological landscape of SN formed by real-world existing web services downloaded from public repositories, with our findings. Section 4 is involved in related works. With a conclusion and an outlook at the work to come in Section 5 we will finish this paper.

Service Network (SN)

For discovering and assembling individual Web service into more complex yet new and useful business processes, it is not enough to advertise what Web services can do based on their syntactic interface to client. Furthermore, each Web service should know the facts: who are its "friends" and who can assist or replace it to fulfill a given task. That is, all available Web services published on the Web constitute a complex social ecosystem, in which every one really understands its situation and knows each other. Several related but distinct points of SN are summarized and refined:

1) SN is a cross-linked social graph, or a complex affiliation network, in which the nodes correspond to services and the edges to semantic links (or Service Relation, SR) between services. It has the form:

$$SN = Graph_{ws} < V_{ws}, E_{ws} >, \text{ in which}$$
$$V_{ws} = \{Abstract\ WS\ node\} \cup \{Concrete\ WS\ node\}$$
$$E_{ws} = V_i \times V_j \quad V_i, V_j \in V_{ws}$$

2) Services node can be classified as abstract services shaped like an ellipse and concrete services like a ball in Fig 1. An abstract service is a functionality operation enclosed a set of interface operations, which come from different concrete services but have the common capabilities. Whereas a concrete service, corresponding to one Web service, may consist of one or many interfaces which described what it can do. Therefore, there are many-many from concrete services to abstract services, and SN is separated into two vertical sections in logic: the abstract service (functionality) layer on the top and the concrete service (implementation) layer at the bottom.

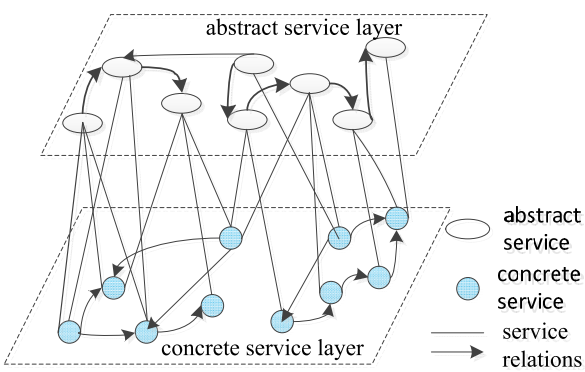

Fig 1. Schematic diagram of SN

3) Service Relation is the contact, tie and connection which does not simply denote the interface-specific matching but synthesizes syntax- and semantic-level relations, that is, the interaction patterns which present the competition, cooperation and collaboration in some business context. From the functional perspective, Web services can be modeled as a tree concept structure which consists of Service, Interface, and Parameter. That is, the Web service, ws, consists of one or many interface operations, which take one or many input parameters, and return one or many output parameters. As a consequence, Service Relations have an appropriate level of granularity consisted of Service-, Operation- and Parameter-level.

Parameter-level relationships are identified as $partOf(x,y)$, $attributeOf(x,y)$, $kindOf(x,y)$ and $equivalentOf(x,y)$, whose semantics is shown in Fig 2.

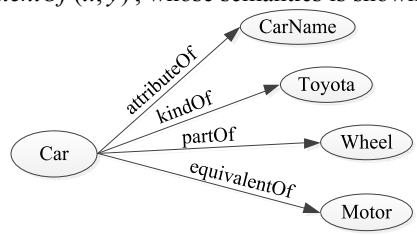

Fig 2. Semantics of Parameter-level relationships

Service-level relationships are based on Operation-level, and Operation-level on Parameter-level. For example, *fully sequential* and *partially sequential* relationships are formulated as:

$$sequTotal_o < op_1, op_2 > \Leftrightarrow out_1 \neq \phi \wedge in_2 \neq \phi \wedge in_2 \subseteq_s out_1$$
$$sequPart_o < op_1, op_2 > \Leftrightarrow out_1 \neq \phi \wedge in_2 \neq \phi \wedge out_1 \subseteq_s in_2$$
$$sequTotal_s < ws_1, ws_2 > \Leftrightarrow \exists op_1 \in ws_1 \wedge \exists op_2 \in ws_2 \wedge sequTotal_o < op_1, op_2 >$$
$$sequPart_s < ws_1, ws_2 > \Leftrightarrow \exists op_1 \in ws_1 \wedge \exists op_2 \in ws_2 \wedge sequPart_o < op_1, op_2 >$$

where, $X \subseteq_s Y$ denotes that parameter (or a set of parameters) X is included in parameter (parameters set, respectively) Y in semantics. That is to say, op_1 in Web service ws_1 can fully invoke op_2 in Web service ws_2, or respectively partially invoke op_2, meaning that in the full invocation ws_1 is a sufficient condition for ws_2 and on the contrary, only merging with another one or several output parameters in other Web service, ws_1 can invoke ws_2 in the partial mode. An edge from a Web service node ws_1 to ws_2 indicates that there is a sequential relationship (fully or partially) between an operation op_1 of ws_1 and an operation op_2 of ws_2, furthermore, the operation op_1 produces the parameter out_1 which is used as one of inputs for the operation op_2.

Except these collaborative relationships between Web services, there are several competitive ones, mainly including *subsume* and *exact*. They are formulated as:

$$subsume_o < op_1, op_2 > \Leftrightarrow out_2 \neq \phi \wedge in_2 \neq \phi \wedge in_2 \subseteq_s \text{in}_1 \wedge out_2 \subseteq_s out_1$$

$$subsume_s < ws_1, ws_2 > \Leftrightarrow \exists op_1 \in ws_1 \wedge \exists op_2 \in ws_2 \wedge subsume_o < op_1, op_2 >$$

$$exact_o < op_1, op_2 > \Leftrightarrow out_1 \neq \phi \wedge in_1 \neq \phi \wedge out_2 \neq \phi \wedge in_2 \neq \phi \wedge in_2 \subseteq_s \mathrm{in}_1 \wedge in_1 \subseteq_s \mathrm{in}_2 \wedge out_1 \subseteq_s out_2 \wedge out_2 \subseteq_s out_1$$

$$exact_s < ws_1, ws_2 > \Leftrightarrow \exists op_1 \in ws_1 \wedge \exists op_2 \in ws_2 \wedge exact_o < op_1, op_2 >$$

In semantics, $exact_s < ws_1, ws_2 >$ indicates that Web service ws_1 and ws_2 are equivalent in functionality and can be replaced by each other. For $subsume_s < ws_1, ws_2 >$, it indicates that Web service ws_1 embraces ws_2 and then ws_2 can be replaced by ws_1, but the reverse is not true.

Both at Operation- and Service-level, there are:

$$plugin_o = \neg subsume_o \qquad plugin_s = \neg subsume_s$$

$$precPart_o = \neg sequPart_o \qquad precPart_s = \neg sequPart_s$$

$$precTotal_o = \neg sequTotal_o \qquad precTotal_s = \neg sequTotal_s$$

These three granularity Service Relations and their relevance are shown in Fig 3. But only Service-level relationships are represented in SN to show the flow of control or data and the competitiveness among Web services.

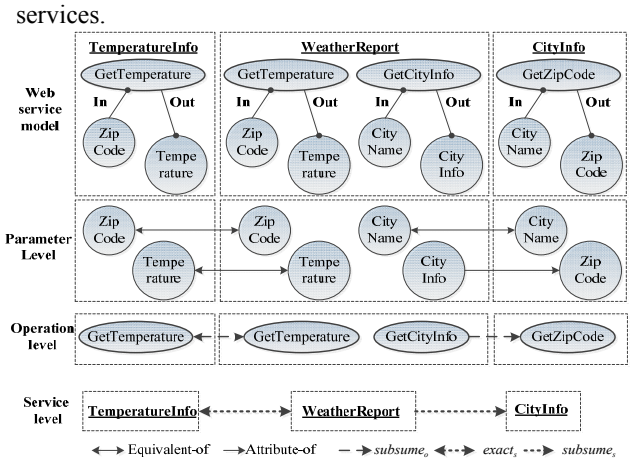

Fig 3. Sample of Service Relation

4) Annotation-based Services Mining (Wang, Feng et al. 2010) was presented to discover knowledge about semantic relationships and associations among Web services and to construct SN.

Network Analysis

In this section, we use analytical methodology offered through the study of complex networks to explore the structural and social characteristics of SN formed by real-world data set.

Research Method

First, 300 available web services (as WSDL files) from travel domains were taken from the public repositories, Seekda (http://webservices.seekda.com). We referred to these data sets as TEST. Second, we downloaded the OWLS-TC version 3 from http://www.semwebcentral.org, denoted by OWLS-TC3, that semantic Web services data set written in OWL-S from seven different domains, such as education, medical care, food, travel, communication, economy, and weapons. And 867 semantic Web services were parsed. Then we had two snapshots of the SN based on real-world which direct the further analysis as to what aspects of the network to study.

To examine relationships between nodes of the same type, we projected SN to a network of abstract services which cluster Web services according to functionality, and a network of concrete services. Thus, the abstract services layer had been split away from the SN by cutting the relationships between two layers simplistically and formed abstract services network. On the other hand, concrete services that are linked to the same abstract services node were added an *equivalentOf* link in concrete services network, a projection of the SN to Web services in the real-world.

Similar to many studies on the complex network, we restricted our attention to a variety of network properties to characterize SN and Web services ecosystem. That is, we analyzed abstract services network and concrete services network, which are constructed from the TEST and OWLS-TC3 data set, to see whether they exhibit characteristics of the Small-World effect and Scale-Free property, and to explore the topological property and evolution mechanism of SN to reveal SOC paradigm. For visualizing and analyzing specific network properties, we used Pajek (http://pajek.imfm.si/doku.php).

Research Method

After validation and Services Mining, the details of some statistics of TEST and OWLS-TC3 are summarized in Table 1. It shows how complex SN is by measuring how many abstract services nodes, concrete services nodes and relationships are involved in the data set.

Another variety of network properties can be obtained from this data that allow us to characterize SN, identify significant component and discover the law behind the structure of SN. These include the average shortest path length L, and the clustering coefficient C, and the small world network index defined as: $\mathrm{Index}_{SN} = \frac{|C_{actual} - C_{random}|}{|L_{actual} - L_{random}|}$, compared to random graphs with the same number of nodes and same average number of edges per node.

It shows that, in each aspect of SN (i.e., *fully sequential, partially sequential, exact, and subsume*) generated from TEST and OWL-S TC3 data set, the small-world effect is not obvious, whose C and L shows: $C_{actual} \gg C_{random}$ but $L_{actual} \leq L_{random}$ except for *partially sequential* in

abstract services network of OWLS-TC3(C_{actual}=0.0006, C_{random} =0.0014) and *fully sequential* in abstract services network of TEST (L_{actual} =2.0969 , L_{random}=1.8850). And there are a great number of isolated nodes.

Table 1. Statistics of TEST and OWLS-TC3

* N: the number of nodes in each aspect of network, E: the number of edges, $\overline{K}$: the average degree, and I: the number of isolated nodes. There is no exact relationship between abstract services, because if exist, the two nodes will merge.

(a) abstract services network of OWLS-TC3

scheme	*N*	*E*	$\overline{K}$	L_{actual}	C_{actual}	L_{random}	C_{random}	$Index_{SN}$	*I*
fully sequential	693	6033	17.4113	2.3131	0.0168	3.2653	0.0123	0.0047	159
partially sequential	693	686	1.9798	3.2571	0.0006	7.7350	0.0014	0.0002	458
subsume	693	1985	5.7287	1.0000	0.2748	6.1354	0.0037	0.0528	58
fully+partially sequential	693	6719	19.3911	3.8408	0.0217	3.1240	0.0139	0.0109	134
exact+subsume	-	-	-	-	-	-	-	-	-
all	693	8704	25.1198	4.0560	0.2572	2.8423	0.0180	0.1971	7

(b) concrete services network of OWLS-TC3

scheme	*N*	*E*	$\overline{K}$	L_{actual}	C_{actual}	L_{random}	C_{random}	$Index_{SN}$	*I*
fully sequential	867	6033	13.9170	2.3131	0.0134	3.7020	0.0075	0.0042	333
partially sequential	867	686	1.5825	3.2571	0.0005	3.5362	0.0000	0.0018	632
exact	867	268	0.6182	1.0000	0.0911	1.4621	0.0000	0.1972	708
subsume	867	1985	4.5790	1.0000	0.2197	7.8296	0.0017	0.0319	232
fully+partially sequential	867	6719	15.4994	3.8408	0.0173	3.5353	0.0093	0.0263	333
exact+subsume	867	2253	5.1972	1.3185	0.2796	7.0254	0.0028	0.0485	135
all	867	8954	20.6551	3.9738	0.2658	3.1507	0.0119	0.3085	86

(c) abstract services network of TEST

scheme	*N*	*E*	$\overline{K}$	L_{actual}	C_{actual}	L_{random}	C_{random}	$Index_{SN}$	*I*
fully sequential	1897	413878	436.3500	2.0969	0.6000	1.8850	0.1151	2.2878	97
partially sequential	1897	2368953	2497.5783	1.2027	0.8370	1.3414	0.6587	1.2861	4
subsume	-	-	-	-	-	-	-	-	
fully+partially sequential	1897	2782831	2933.9283	1.2172	0.9488	1.2263	0.7737	19.2573	0
exact+subsume	-	-	-	-	-	-	-	-	
all	1897	2782831	2933.9283	1.2172	0.9488	1.2263	0.7737	19.2573	0

(d) concrete services network of TEST

scheme	*N*	*E*	$\overline{K}$	L_{actual}	C_{actual}	L_{random}	C_{random}	$Index_{SN}$	*I*
fully sequential	300	24947	166.3133	1.6331	0.5359	1.7219	0.2783	2.9031	8
partially sequential	300	68569	457.1267	1.2014	0.8739	1.2356	0.7644	3.2048	0
exact	-	-	-	-	-	-	-	-	
subsume	-	-	-	-	-	-	-	-	
fully+partially sequential	300	71607	477.3800	1.1968	0.8879	1.2017	0.7983	18.3978	0
exact+subsume	-	-	-	-	-	-	-	-	
all	300	71607	477.3800	1.1968	0.8879	1.2017	0.7983	18.3978	0

In fact, Web services are the components of SOC, and as such provide added value to users when they are composed or allow them to be reused in new contexts. The more complementary (i.e., the fully sequential and partially sequential relationships in our opinion) Web services SN has, the more attractive it becomes to users, and hence it will be more useful for SOC. So we focus our attention more on the composed issue. For example, it shows that all complementary networks generated from OWL-S TC3 data set are small-world networks. For considering all four type relationships, the same phenomenon also is founded.

Unfortunately, the Small-World effect is not so much obvious in TEST data set which is not like the finding by (Kil et al. 2009). One explanation for this result is that although the number of Web services in TEST is far more than that in (Kil et al. 2009), but it is from only one domain. In this situation, as a result of considerable amount of *partially sequential* relationships (i.e., 2368953 for abstract services layer and 68569 for concrete services layer), the SN has evolved into a great community whose local density is very high and global measure of separation is very low.

However, $Index_{SN}$, which represents how strong small word property a network has, illustrates that each aspect of

SN has a different level of small word properties. In the case of TEST, Index_{SN} is generally bigger than one of OWL-S TC3. This result implies that the collaboration within one domain makes the network denser with the increased number of edges.

After all, the small-world effect suggests that available Web services have short-cuts, which is a valuable hint to heuristic web services composition algorithm. At the same time, the fact that most pairs of vertices in SN seem to be connected by a short path through the network (the biggest L_{actual} =4.0560) also is an underlying assumption to improve the efficiency of web services discovery and to estimate how many component Web services sequentially involved in a business process.

The degree k of a node representing a Web service is a measure of the number of Web services interaction (both collaboration and competition) with the Web service. And again, as the table shows, the values of average degree $\overline{K}$ are in OWLS-TC3 at the same level as (Kil et al. 2009), but the values in all cases of TEST are quite big—much bigger than the published networks in (Newman 2003), for instance.

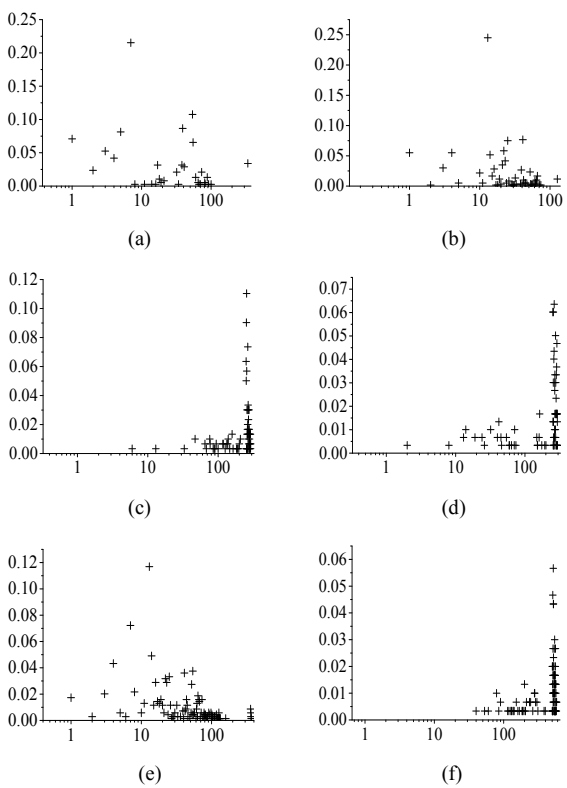

Fig 4. Degree distribution of concrete services network: (a) In-degree distribution of OWLS-TC3 over sequential (including fully and partially sequential), (b) Out-degree of OWLS-TC3 over sequential, (c) In-degree of TEST over sequential, (d) Out-degree of TEST over sequential, (e) undirected degree distribution of OWLS-TC3, and (f) undirected degree distribution of TEST. X-axis is the degree *k* of web services, and Y-axis is the *P*(*k*).

The degree distribution of Web services over *fully sequential* and *partially sequential* relationships measures the probability $P(k)$ of finding a Web services with this degree and is an indicator of the popularity of an available Web service in SOC paradigm. It is strange that the In- and Out-degree distributions of different aspects of two data sets are far from neither uniform nor power-low (see Fig 4 a-d). Although each of relationships is directed, considering *subsume* with *plugin*, *sequPart* with *precPart*, and *sequTotal* with *precTotal*, SN is a undirected network. The degree distributions are shown in Fig 4 e-f). Compared with other networks human made, by and large, the degree distributions show fewer power-law distributions.

One probable explanation for this result is the follow. As we have designed, SN follows the classical scenario of a growing network. It continues to expand one Web services node by one Web services node, through Services Mining algorithm, which is the first and necessary condition for the emergence of a scale-free topology or power-law distribution. The second and more important condition, preferential attachment, does not attract enough attention, however.

In SOC paradigm each Web service node also has a certain *fitness* (Bianconi and Barbasi 2001). We assumed that a node's *attractiveness* was determined solely by its number of operations and parameters. It is not enough to characterize *fitness* with functionality, but without non-functionality. For example, Web services with higher QoS and better user experience are invoked more frequently.

For Web services and SN, things are a bit more complicated. First, Web services do not appear randomly across the Web. They are published by provider, and what they can do depends on the provider's will. Thus there is a strong correlation between link density and the functionality and parameters involved in Web services. Second, in a competitive environment collaboration never has equal opportunity. Instead, popularity based higher QoS and user rating is attractive. The inherent quality of Web services, such as functionality, interoperability semantics, and so on, is not the only decisive factor of fitness. User experience also plays a significant role: Web services with higher popularity are more likely to be discovered and invoked again. Third, nodes and links can disappear. Indeed, many Web services go out of availability, taking with them from several to hundreds of links. Links can also be rewired, as when Web services are upgraded or the provider decides to pay more attention to a Web services published by his business partner. At the same time, User experience also has an effect on link: a lot of lower user rating could weaken the vitality of link, even make it disappear. Therefore, when trying to model the SN, we must simultaneously consider the interplay of growth, preferential attachment based on fitness, functionality dependence, and an underlying factor of human.

Nevertheless, it is evident to observe the existence of hub web services for concrete layer, with huge number of In- or Out-degrees while majority has only a few links (see Fig 5). That is, some Web services enjoy significantly greater popularity than others. It implies that some Web services play as the keystones and provide the basis for the majority SOC, and the other Web services are only selected in certain application niches. This is consistent with the finding of Weiss and Gangadharan (2010) in their study of Mashup ecosystem.

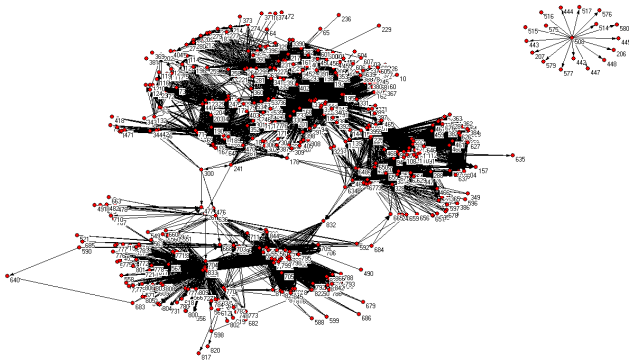

Fig 5. Snapshot of SN using OWLS-TC3 data set (numbered nodes represent Web services, a link between two Web services indicates that the former can fully or partially invoke the later).

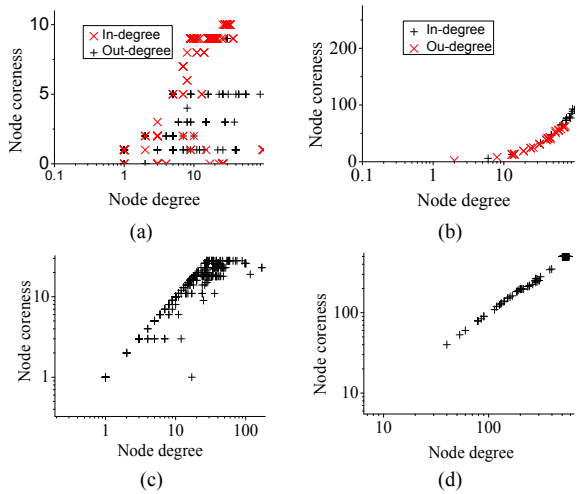

Fig 6. Degree coreness correlation:
(a) In-degree coreness correlation and Out-degree coreness correlation of OWLS-TC3, (b) In-degree coreness correlation and Out-degree coreness correlation of TEST, (c) degree coreness correlation of OWLS-TC3, and (d) degree coreness correlation of OWLS-TC3.

The correlation between the degree and the coreness value, which corresponds to the fact that the central nodes are most likely high-degree hubs of the network, is shown in Fig 6. In OWLS-TC3, when the degree is small, the distribution follows a power 1aw, while the degree is bigger, whose coreness value remains essentially unchanged. Although, the degree and the coreness of TEST are generally higher than that of OWLS-TC3, the power law also exists.

In summary, we see that while there are some different characteristics between SN and published networks, and the distribution of SN over collaboration and competition is not uniform, nor follows a classic power law, SN has better navigability for its smaller separation and denser connectivity. It is helpful to guide and promote SOC paradigm, where web services discovery and composition algorithm can navigate along Service Relations and follow a link in both directions.

Related Works

Social Networking is built on the idea that there is a determinable structure, notions as six degrees of separation, to how people know each other, whether directly or indirectly. Therefore, it was considerable interdisciplinary interest along the last decades that graph theory started to be systematically applied to represent and model natural and human-made networks, more specifically social relations, such as trade networks, transportation networks, biomolecular networks, Internet, WWW, personal relations, citation networks, etc., spurred by the work in (Watts and Strogatz 1998) and (Barabasi and Albert 1999).

In SOC area, an early contribution to the Web services relation was that by (Zhang et al. 2002; 2007), in which the authors developed a business relationship description language, WSRL, to facilitate services discovery and integration. Combining Social Networking and Semantic Web technology into Web Services repository, (Chen and Feng 2008; 2009; 2010) proposed an infrastructure for Service-oriented Computing, named Service Network (SN), which introduces relation reasoning and computing into SOC paradigm and plays an important role to facilitate flexible and automatic service discovery, composition and management. (Al-Masri and Mahmoud 2008) investigated and determined the Web service statistics and distribution based on object sizes, types of technologies employed, and the number of functioning services to provide insights on improving the service retrieval process. Using graph theory, (Kil et al. 2009) studied topological aspects of various web service networks, and found that all of them show small world properties well and power-law-like distribution to some extent. From a social networking perspective, (Maaradji et al. 2010) proposed Social Composer to handle services discovery and selection. Using Web Services Resource Framework, (Cai 2007) extended WSDL into Scale-Free Web Services.

Conclusion and the Future Work

In this paper, we examined the structure of the web services network generated by the real-world data set. The

main contribution of our paper is a research method for seeking a global web services map on the real-word data. Its novelty lies in the nature of interconnection (either collaboration or competition) in SOC based on Web services, and the use of network analysis to obtain key characteristics of the SN. We observed that SN is a complex social network with smaller average separation and denser connectivity. There exists the Small-world phenomenon, but no less significant the power law distribution. However, some Web services enjoy significantly greater popularity than others and play as the keystones for SOC. In our opinion, with knowing the SN's topology, it is possible that to design better and higher efficient algorithms and tools to quickly discover and cleverly choreograph several Web services to satisfy the given request on-demand.

Several open questions remain including what laws underlie the growth of SN and how SN shapes the SOC. One question is that the selection and binding of Web service for a given purpose is not only driven by what functionality the Web service provides in semantics but also implied by the QoS and user experience. Distinctly the topological structure of SN can be expected to be influenced and shaped by the history selection and non-functionality. It is mainly about more sophisticated fitness metrics for Web services. This question then becomes whether the so-called “rich gets richer" phenomenon (Barabási et al. 2002) applies to SN, and in what form further.

Another direction of our future work is to extend our analysis to the whole Web services data from Seekda to evaluate our work further.

Acknowledgements

The authors wish to thank Qixuan Liang for his valuable work and the members of service computing group from Institute of Knowledge Science and Engineering, Tianjin University. This work was supported by the National Natural Science Foundation of China under Grant No. 61173155, the National High-Tech Research and Development Program of China under Grant No. 2007AA01Z130, and the Innovation Foundation of Tianjin University under Grant No. 2010XG-0009.

References

Al-Masri, E. and Mahmoud, Q. H. (2008). Investigating web services on the world wide web. Proceeding of the 17th international conference on World Wide Web(WWW2008). Beijing, China, ACM: 795-804.

Barabási, A. L., Jeong, H. et al. (2002). Evolution of the social network of scientific collaborations. Physica A: Statistical Mechanics and its Applications 311(3–4): 590-614.

Barabasi, A. L. and Albert, R. (1999). Emergence of scaling in random networks. Science 286(5439): 509-512.

Bianconi, G. and Barbasi, A. L. (2001). Competition and multiscaling in evolving networks. Europhysics Letters (epl) 54(4): 436-442.

Cai, H. (2007). Scale-free web services. In: Internatioal Conference on Web Services (ICWS):288- 295.

Chen, S. and Feng, Z. (2008). Services Network：The New Anchor for Web Services Composition. Journal of Application Research of Computers 25(5): 1378-1382.

Chen, S., Feng, Z. et al. (2010). Service Relations and its Application in Services-Oriented Computing. Chinese Journal of Computers 33(11): 2068-2083.

Chen, S., Feng, Z. et al. (2009). Building the semantic relations-based web services registry through services mining. 8th IEEE/ACIS International Conference on Computer and Information Science(ICIS 2009). Shanghai, China, IEEE Computer Society: 736-743.

Dustdar, S. and Schreiner, W. (2005). A survey on web services composition. International Journal of Web and Grid Services 1(1): 1-30.

Kaijun, R., X. Nong, et al. (2011). Building Quick Service Query List Using WordNet and Multiple Heterogeneous Ontologies toward More Realistic Service Composition. IEEE Transactions on Services Computing, 4(3): 216-229.

Kil, H., Oh, S.-C. et al. (2009). Graph Theoretic Topological Analysis of Web Service Networks. World Wide Web 12(3): 321-343.

Maaradji, A., Hacid, H. et al. (2010). Towards a Social Network Based Approach for Services Composition. 2010 IEEE International Conference on Communications (ICC). Cape Town:1-5.

Newman, M. E. J. (2003). The Structure and Function of Complex Networks. SIAM Review 45(2): 167-256.

Papazoglou, M. P. and D. Georgakopoulos (2003). Service-oriented computing. Communications of the ACM 46(10): 24-28.

Papazoglou, M. P., Traverso, P. et al. (2008). Service-Oriented Computing: A research roadmap. International Journal of Cooperative Information Systems 17(2): 223-255.

Wang, H., Feng, Z. et al. (2010). Constructing Service Network via Classification and Annotation. 5th IEEE International Workshop on Service-Oriented System Engineering (SOSE2010). Nanjing, China:69-73.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. Nature 393(6684): 440-442.

Weiss, M. and Gangadharan, G. R. (2010). Modeling the mashup ecosystem: structure and growth. R&D Management 40(1): 40-49.

Zhang, L.-J., Chang, H. et al. (2002). Web Services Relationships Binding for Dynamic e-Business Integration. International Conference on Internet Computing (IC'02). Las Vegas, NV: 561-570.

Zhang, L.-J., Zhang, J. et al. (2007). Services Relationship Modeling. Services Computing Springer Berlin Heidelberg: 114-133.