A method for predicting perishing services in a service ecosystem

Bofei Xia  
Department of Automation  
Tsinghua University  
Beijing 100084, China  
xiabf11@mails.tsinghua.edu.cn

Yushun Fan  
Department of Automation  
Tsinghua University  
Beijing 100084, China  
fanys@tsinghua.edu.cn

Keman Huang  
Department of Automation  
Tsinghua University  
Beijing 100084, China  
khm09@mails.tsinghua.edu.cn

Abstract—With the wide adoption of Service-oriented Architecture, we can observe a rapid increase of web services these days. These services with their compositions produced by consumers form a service ecosystem. However as time goes by, some of the services are no longer available or says perishing due to the competition among the ecosystem. Obviously, compositions invoking these perishing services are also becoming unavailable. Thus, choosing the services which are more stable can help the consumers to product more valuable composition. The goal of this paper is to find a way to separate the potential perishing services in the ecosystem so that we can give suggestions to service consumers and help them develop more durable and valuable compositions. Firstly, we study the competition relation between services by their user tags and establish a service-service competition network. Based on the network analysis, we extract the common feature of perishing services and formalize this feature as percentage ranking (PR) of services. Finally, we propose a classification algorithm to predict potential perishing services. With the good performance in recall and precision rate, our algorithm is credible for identifying potential perishing services thus we can suggest the consumers to select more durable services for their compositions.

Keywords-service competition, service user tags, complex network analysis, binary classification

I. INTRODUCTION

With the prevailing of Service-oriented Architecture (SOA), individual services on the Web together with service compositions are gradually giving rise to service ecosystem [1]. Service providers release services into the ecosystem with the expectation of high usage rate. Service consumers compose the existing services into compositions to fulfill their requirements. However due to the competitions among services in the ecosystem, some services will perish or emigrate from the ecosystem over time. Here, ‘perish’ means the services which have ever been invoked in compositions are not accessible anymore. Obviously, service compositions which invoke these perishing services are also unavailable. Thus, how to effectively and efficiently measure the competitive strength of the services is very important for the consumers so that they can construct more durable and valuable compositions with the persistent available services. Consequently, the goal of our work focus on two aspects: how to explore the common features of perishing services from the view of service competition and how to take advantage of the common features to predict potential perishing services so that we can offer suggestion for the consumers.

Social Tagging is a major characteristic of Web 2.0 [2]. Nowadays, annotating web service by user tags gains a lot of momentum among web service repositories like Seekda.com, Programmable.com. Compared to the traditional semantic Web service description methods like WSDL-S [3], WSMO, OWL-S [4], service tags neglect trivial specifics and outline a concise function of services. In advantage of this, consumers can search for their target services by particular service tags instead of analyzing the sophisticated service description files. In this way, services which share same user tags compete for specific consumers. So we can establish the competition relationship among services through their user tags which results into a complex competition network among services. Thus the complex network can be a powerful tool to study and extract the common features for those perishing services. The main contribution of this paper is providing a heuristic method for service consumers to quantify the durability of invoked services. Using services and their user tags, we establish a service-service competition network to describe the competition relationship inside the service ecosystem. Then based on the network analysis, we explore the common feature for the perishing services. In this paper, we name it as percentage ranking (PR). Finally, based on the percentage ranking we propose a classification method to separate the potential perishing services.

The rest of the paper is organized as follows: Section 2 presents our service-service competition networks in formalism way. The data set used in this paper is shown in Section 3. Section 4 shows the analysis of the service-service competition network and extract the common feature for those perishing services. Section 5 introduces and evaluates the algorithm of predicting perishing services. The related work is discussed in Section 6 and Section 7 concludes our work.

II. NETWORK MODEL

Services of competition must be, in a way, similar in capability [5]. We choose user tags to describe the services competition relation for the following two criterions:

Criterion 1
Since user tags sketch out service’s function, we can infer overlap on capabilities from overlap on user tags. Thus, there exists a competition relationship between two services if their user tags have an intersection.

Criterion 2
The more services that provide a certain function, the more intensive the competition among them is. And this can
be reflected on a user tag’s occurrence frequency in the services.

Accordingly, to describe the competition relationship among services, we first construct a service-tag network (ST network), an undirected weighted network in which a node presents either a service or a tag and edges represent the inclusive relation between them. Then extract a service-service competition network, an undirected weighted network in which a node presents a service and edges represent the competition relationship between them, as illustrated in Fig. 1.

**Figure 1.** Service-tag bipartite graph and service-service competition network

Enlightened by [6], we formalize the service-tag network as a \( r \times c \) matrix, where \( r \) is the number of services and \( c \) is the number of distinct user tags.

\[
ST = [s_{ij}], 0 \leq i \leq r, 0 \leq j \leq c
\]

(1)

Where \( s_{ij} = 1 \) if service \( i \) is annotated with tag \( j \).

Then, we extract a service-service network in which two services are connected if they share some tags.

\[
SS = [s_{ij}] = ST \cdot ST^T, 0 \leq i, j \leq r
\]

(2)

Where \( s_{ij} \) is the number of intersection tags of service \( i \) and service \( j \).

The SS network only counts the number of intersection tags between two services, but ignore the difference between tags. In order to reveal a more reasonable competition relationship between services we should also take tag occurrence frequency into account.

For one thing, we define a \( c \times c \) matrix \( TF \),

\[
TF = [t_{ij}], 0 \leq i, j \leq c
\]

(3)

Where \( t_{ij} = \begin{cases} \sum_{k=1}^{c} s_{ik}, i = j \\ 0, i \neq j \end{cases} \)

\( TF \) is a diagonal matrix in which the diagonal element \( \sum_{k=1}^{c} s_{ik} \) equals the sum of column \( j \) of \( ST \) matrix that represents the number of services annotated with tag \( k \).

Furthermore, we obtain a service-service competition network in consideration of tags frequency,

\[
SSTF = [s_{uj}'] = ST \cdot TF \cdot ST^T, 0 \leq i, j \leq r
\]

(4)

Where \( s_{uj} \) is the sum of tag frequency of intersection parts between service \( i \) and service \( j \), denoting the influence degree between service \( i \) and service \( j \) in their competition relationship.

Finally, we obtain two useful networks: \( SS \) network follows the criterion 1 but doesn’t take tag frequency into consideration; \( SSTF \) network follows both criterions and demonstrates a relatively integrated competition relation between services. The analysis of \( SSTF \) network is our major task in the following section and \( SS \) network is used as a contrast to demonstrate the rationality of tag frequency.

### III. DATA SET ACQUISITION

To our best knowledge, ProgrammableWeb.com\(^1\) is by far the largest online repository of Web APIs and mashups. Web APIs are Web accessible endpoints for users to invoke which can be considered as a form of web service. Mashup is a way to compose Web APIs to create new applications [7]. We should note that we do not distinguish between Web API and service or mashup and composition in this paper. From the directory of Web APIs and mashups, we get a list of Web APIs that have ever been invoked in mashups. Some of these APIs are available all the time which are referred to as persistent available APIs. However some are not, which we name them as perishing APIs. The perishing APIs definitely cause the mashups which invoked them to become unavailable. We obtain the ProgrammableWeb.com data from June 2005 to June 2012. Each service’s information contains its name, provider, category, emerging date and a set of tags.

Finally, we retain a collection of 1149 services and 570 distinct user tags. The detail information is illustrated in Table 1.

<table>
<thead>
<tr>
<th>TABLE I. DATA SET OF THE PROGRAMMABLEWEB.COM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of services</td>
</tr>
<tr>
<td>Total number of service provider</td>
</tr>
<tr>
<td>Number of perishing services</td>
</tr>
<tr>
<td>Total number of user tags</td>
</tr>
<tr>
<td>Number of distinct user tags</td>
</tr>
<tr>
<td>Average number of tags per services</td>
</tr>
<tr>
<td>Average number of services per tag</td>
</tr>
</tbody>
</table>

### IV. NETWORK ANALYSIS AND COMMON FEATURE EXTRACTION

Applying the complex network analysis approaches, we study \( SSTF \) network. Based on the observation of the network analysis, we extract a common feature of perishing services.

---

\(^{1}\) www.programmableweb.com
A. Weighted degree and Percentage ranking measure

The weighted degree of vertex in SSTF network represents the interaction among services that provide similar functions and may compete for the opportunity of being invoked by compositions. After a coarse observing of the weighted degree distribution, we notice that large number of perishing services tend to have relative high weighted degree. Thus, we introduce a processing method to extract and formalize the common feature of perishing services by weighted degree.

Let \( V = \{ v_{(1)}, \ldots, v_{(n)} \} \) denotes the set of all vertexes in SSTF network. \( WD(v_{(i)}) \) represents \( v_{(i)} \)'s weighted degree, \( WD(v_{(1)}) \geq \ldots \geq WD(v_{(i)}) \ldots \geq WD(v_{(n)}) \), where \( 1 \leq i \leq n \) and \( n \) is the total number of services. Besides, we can get two subsets of \( V : V_{PA} \) and \( VP \) which denote all persistent available services and perishing services, respectively. Then we define \( PR = \{ p_1, \ldots, p_n \} \) to formalize the percentage ranking of each vertex in \( V \), where \( p_i = \frac{i}{100} \) and \( i \) is in accordance with \( v_{(i)} \)'s subscript. Subsequently, we define an indicator function \( I_{\alpha}(pr) \):

\[
I_{\alpha}(pr) = \begin{cases} 1, & p_i \in PRP \\ 0, & p_i \in PRPA \end{cases} \\
1 \leq i \leq n
\]  

(5)

Where \( PRP = \) the maximum subset of \( PR \), \( \forall p_i \in PRP, v_{(i)} \in VP, PRPA = \) the maximum subset of \( PR \), \( \forall p_i \in PRPA, v_{(i)} \in VPA \). To summarize, \( I_{\alpha}(pr) \) is actually an indicator of perishing services, that is, \( I_{\alpha}(pr) \) equals 1 indicates service with percentage ranking \( pr \) is perishing.

![Indicator of perishing services](image)

Figure 2. \( I_{\alpha}(pr) \) of SS and SSTF network

Fig. 2 gives the bar diagram of \( I_{\alpha}(pr) \) in SS network and SSTF network. Intuitively, most perishing services distribute in the relative high percentage ranking section. Thus, a high percentage ranking (PR) of services weighted degree can be considered as a common feature of perishing services. In order to quantify the distinctive ability of PR, we define a Partition Degree of \( I_{\alpha}(pr) \) as follows:

\[
PD = \frac{\sum_{i=1}^{N_{vp}} I_{\alpha}(pr)}{N_{vp}}, 1 \leq i \leq N_{vp}
\]

(6)

Where \( N_{vp} = \) the number of elements of \( VP \) representing the total number of perishing services. The value of PD ranges from 0 to 1. For example, if the total number of perishing services is 100 and the sum of \( I(p_{r1}), I(p_{r2}), \ldots, I(p_{r_{100}}) \) is 90, then PD equals to 0.9 which means that \( I_{\alpha}(pr) \) can partition 90% perishing services into relative high percentage ranking section; if the total number of perishing services is 100 and the sum of \( I(p_{r1}), I(p_{r2}), \ldots, I(p_{r_{100}}) \) is 10, then PD equals 0.1 which means that \( I_{\alpha}(pr) \) can partition only 10% perishing services into relative high ranking section.

\( PD \) of SS network is 0.637 and PD in SSTF network is 0.887. Since \( PD \) in SSTF network is larger than that in SS network, we claim that service’s percentage ranking (PR) in SSTF network is a more distinctive feature.

B. Trace of Partition Degree of PR

Since the dataset covers all the services that have been involved in compositions from June 2005 to June 2012, we divide time interval as one year and thus get seven time endpoints. By collecting arrival time of each services and perishing time of those perishing ones, we have an explicit knowledge of which services have emerged or perished till each time endpoint, respectively.

From the analysis of static structure part, we know that majority of perishing services are mainly distributed in high percentage ranking section. And we measure this property by partition degree of \( I_{\alpha}(pr) \). In this subsection, we trace the variation of partition degree from a chronological view in order to explain the formation of the common feature of perishing services.

![Trace of partition degree](image)

Figure 3. Trace of partition degree of \( I_{\alpha}(pr) \) in SSTF network

Shown in Fig. 3, the partition degree of \( I_{\alpha}(pr) \) is an increasing series from June 2006 to June 2012. In the inaugural phase of system, \( PD \) of \( I_{\alpha}(pr) \) is at a relative small value because the competition between services is not intense. Though some services with high percentage ranking
positions, they keep temporary available. That explains why the partition degree is low in the former years. As a large number of services are newly released into the system, services with high percentage ranking position are gradually subjected to competition from more services sharing similar function. For example, let’s focus on a perishing service Gowalla with user-tags \{mobile, social, mapping, location, places\}. Assuming there is a requirement of ‘mapping’, Gowalla need to compete for the invoking opportunity with 18 other services in June 2006 but with 98 other services in June 2012. Those ‘other services’ include some most popular services like Google Maps, Microsoft Bing Maps. Unfortunately, similar tough situation occurs when the requirement are ‘mobile’, ‘social’, ‘location’, ‘places’. The number of service that provides similar function with Gowalla is demonstrated in Table II.

<table>
<thead>
<tr>
<th>Date</th>
<th>mobile</th>
<th>social</th>
<th>mapping</th>
<th>location</th>
<th>places</th>
</tr>
</thead>
<tbody>
<tr>
<td>June-2006</td>
<td>2</td>
<td>7</td>
<td>18</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>June-2007</td>
<td>6</td>
<td>11</td>
<td>34</td>
<td>8</td>
<td>11</td>
</tr>
<tr>
<td>June-2008</td>
<td>7</td>
<td>38</td>
<td>44</td>
<td>13</td>
<td>15</td>
</tr>
<tr>
<td>June-2009</td>
<td>18</td>
<td>86</td>
<td>57</td>
<td>16</td>
<td>22</td>
</tr>
<tr>
<td>June-2010</td>
<td>24</td>
<td>111</td>
<td>70</td>
<td>21</td>
<td>28</td>
</tr>
<tr>
<td>June-2011</td>
<td>33</td>
<td>151</td>
<td>85</td>
<td>28</td>
<td>34</td>
</tr>
<tr>
<td>June-2012</td>
<td>41</td>
<td>168</td>
<td>98</td>
<td>31</td>
<td>40</td>
</tr>
</tbody>
</table>

Not only Gowalla but also other services with high percentage ranking positions face the same problem: though these services can provide several popular functions, but most functions provided by them are similar with too many other services. When these services are not competent to gain more opportunity to be invoked by compositions, their providers are unwilling to supply them anymore. Therefore, more and more services with high percentage ranking are perishing over time. Thus, the partition degree (PD) is increasing and the percentage ranking (PR) is becoming a credible feature to identify perishing services.

V. CLASSIFICATION

Our work shows that high percentage ranking (PR) is a common feature of perishing services. Therefore, this feature can be used to predict whether a service is potential perishing. Then, we can transform the problem into a binary classification: classifying a newly invoked service into two alternative classes in consideration of its percentage ranking.

A. Classifying algorithm

Based on Naive Bayesian Test, our complete classifying algorithm is shown as follows:

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Classifying the newly invoked service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>The newly invoked service and its user-tag</td>
</tr>
<tr>
<td>Step1</td>
<td>Obtain the proportion of perishing service from historical record as ( P )</td>
</tr>
<tr>
<td>Step2</td>
<td>Add the newly invoked service into service ecosystem, renewing ST matrix</td>
</tr>
</tbody>
</table>

| TABLE II. THE NUMBER OF SERVICES PROVIDE SIMILAR FUNCTION WITH GOWALLA |
|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
| Date     | mobile | social | mapping | location | places |
| June-2006 | 2     | 7      | 18      | 6        | 7      |
| June-2007 | 6     | 11     | 34      | 8        | 11     |
| June-2008 | 7     | 38     | 44      | 13       | 15     |
| June-2009 | 18    | 86     | 57      | 16       | 22     |
| June-2010 | 24    | 111    | 70      | 21       | 28     |
| June-2011 | 33    | 151    | 85      | 28       | 34     |
| June-2012 | 41    | 168    | 98      | 31       | 40     |

In classification context, we measure true positive (TP), false positive (FP), false negative (FN) and true negative (TN), which is demonstrated in the confusion matrix illustrated in Table V.

<table>
<thead>
<tr>
<th>Binary Classification</th>
<th>Truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision</td>
<td>perishing</td>
</tr>
<tr>
<td>Claim perishing</td>
<td>TP</td>
</tr>
<tr>
<td>Claim persistent</td>
<td>FN</td>
</tr>
</tbody>
</table>

Then, we can obtain the recall and precision [8] of the testing result of each year, shown in Fig. 4.

![Figure 4. Recall and Precision of testing result](image)

The six groups of testing data cover all the newly invoked services from 2006 to 2012. Fig.4 shows that the algorithm has a sustaining good performance, a relative high recall and precision rate indicating that the algorithm is effective in predicting potential perishing services.

<table>
<thead>
<tr>
<th>TABLE III. CLASSIFYING ALGORITHM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm : Classifying the newly invoked service</td>
</tr>
<tr>
<td>Input: The newly invoked service and its user-tag</td>
</tr>
<tr>
<td>Step1: Obtain the proportion of perishing service from historical record as ( P )</td>
</tr>
<tr>
<td>Step2: Add the newly invoked service into service ecosystem, renewing ST matrix</td>
</tr>
</tbody>
</table>

**Step3**: Using the ST matrix obtained in Step 2, renewing TF matrix

**Step4**: Using the ST matrix obtained in Step 2 and the TF matrix obtained in Step 3, renewing SSTF network

**Step5**: Using SSTF network obtained in Step 4, getting the percentage ranking of newly invoked service as \( pr \). Output: If \( pr < P \) then the newly invoked service is claimed to be potential perishing; otherwise, the newly invoked service is claimed to be persistent available.
VI. RELATED WORK

In recent years, tags of web services are gained large popularity among researchers. [9] puts forward an approach to analyze the WSDL file and annotate web service with tags automatically. [10] provides a multi-dimensional social tagging method for semantic of web services. [11] proposes an effective service discovery method based on tags. [12] illustrates a collaborative tagging-based environment for Web service discovery, allowing users to tag or annotate a Web service using keyword or free-text. Associating with mining techniques, [13] makes an effort to discover relationships between APIs and mashups based on their annotated tags. From the above studies, we know that service tags outline a service function and are suitable for services discovery or service composition. Since the competition between services is due to their similarity on functions, service tags are reasonable for extracting services competition relationship.

The complex network is widely accepted as a powerful tool to study the large scale complex system, such as the Internet [14], public transport [15]. Some static structure properties like network connectivity, clustering coefficient, the shortest path distance are investigated. What’s more, complex network is also used to trace the dynamic metrics of the system in order to study the evolution of the network [16].

The Web service ecosystem is a logical collection of Web services whose exposure and access are subject to constraints characteristic of business service deliver [17]. With the competition among the service ecosystem, a number of services become no longer available which indeed influence the compositions that invoke them. But few work pay attention to this phenomenon. Therefore, we make an effort to study these perishing services from the competition view.

VII. CONCLUSION

As an increasing number of web service invoked in a service ecosystem, some of these invoked services are perishing over time due to the competition.

In this paper, we deem that similarity on service tags reflects the similarity on service functions which induce the competition relation among services. Using services, service-tags and tag occurrence frequency, we can extract a service-service competition network to describe the competition relationship. And we find a common feature of these perishing services: most of their user tags have high occurrence frequency which indicates most functions they provide are similar with too many other services. But none of these several functions is a unique characteristic.

Then we introduce a processing method to formalize the feature as percentage ranking (PR) of each service. Based on Naive Bayesian test, we propose a classifying algorithm to separate the potential perishing services by the percentage ranking. The testing result shows that the algorithm performs well and provides a credible proposal for service consumers.

ACKNOWLEDGMENT

This work is partially supported by the National Key Technology R&D Program of China (No. 2012BAF15G00) and the China Engineering Academy Member Workstation Project At Zhenjiang Zhongmei Co (No. BM2010551).

REFERENCES: