International Journal of Computational Intelligence Systems, Vol. 5, No. 6 (November, 2012), 1040-1051

Constructing Service Semantic Link Network Based on the Probabilistic Graphical Model

ANPING ZHAO[*]
College of Computer and Information Science, Chongqing Normal University, College Town, Shapingba District,
Chongqing, 401331, China
E-mail: apzhao@cqnu.edu.cn

YAN MA
College of Computer and Information Science, Chongqing Normal University, College Town, Shapingba District,
Chongqing, 401331, China
E-mail: csmayan@126.com

Received 18 April 2011

Accepted 28 July 2012

Abstract

Automatic services collaboration calls for the development of semantically structured service network to maximize the utility of Web services. Service Semantic Link Network (S-SLN) is the semantic model for effectively managing Web service resources by the dependency relationship between services. We provided an effective method for constructing S-SLN based on the graphical structure representation of the dependencies embedded in a probabilistic model. A Markov network is an undirected graph whose links represents probability dependencies. We first learned Markov network structure from Web services data, and then transformed the undirected Markov network structure into a directed graph structure of S-SLN based on the same joint probability distribution. Finally, experimental results show the effectiveness of the method.

Keywords: Web service, Probabilistic graphical model, S-SLN, Service relationship

1. Introduction

The rapid increase in the amount of Web service produced in recent years on the Web has resulted in a more sophisticated service process for e-Commerce, which involving numerous interacting business objects within complex distributed processes. Because of the growing number of Web services available, Web services are being made available to increase collaboration in a distributed environment. However, discovering such resources to facilitate automatic services collaboration is a major bottleneck in this context. In order to effectively collaborate between services, Web services need to be organized and their functionalities semantically described.

This fact calls for the development of automatic methods for service relationship discovery and their dependency structure in order to maximize the utility of Web services by making them widely available to the community. One of the key aims of providing Web service resources with semantic descriptions is to improve service discovery for the purpose of collaboration. Semantically-described services can not only be searched, browsed and discovered by using some queries (for instance, via their names or task descriptions), but also on the basis of the semantic relatedness of their functionalities. Therefore, developing a model to describe the inherent

[*] Corresponding author. Email: apzhao@cqnu.edu.cn

Published by Atlantis Press

dependencies and an approach to discover semantic associated service network among domain-specified Web services is critical for their automatic service collaboration.

We use a semantic model *Service Semantic Link Network (S-SLN)* to define semantic structure among Web services. *Semantic Link Network (SLN)* is a loosely coupled semantic data model for managing Web resources. Its nodes can be any types of resources. Its edges can be any semantic relations. A semantic link network instance is a directed graph, denoted as *S (ResourceSet, LinkSet),* where *S* is the name of the semantic link network, *ResourceSet* is a set of resources, and *LinkSet* is a set of semantic links in the form of $R \xrightarrow{\alpha} R'$, where R, $R' \in$ *ResourceSet*, and α is a semantic factor representing a semantic relation between R and R' [1]. S-SLN is a special instance of SLN where nodes are Web services and arcs are service relationship links to describe inherent semantic association among Web services.

S-SLN is the underlying semantic model for effectively implementing automatic Web service search and composition by a relationship dependency network which connects services with different types of relationships. It allows us to specify the types of the inherent service relationship dependencies to describe the behavior of Web services and focuses on collaboration, availability and navigation. When manually assigned annotation relationship tags of related services are not available, we hypothesize that automated approaches could be used to improve the S-SLN discovery process. These include building networks of related services. For example, a user can search for a Web service that corresponds to a particular input, output or operation performed. If, however, the retrieved services do not fulfil the all requirements, the user may be interested in exploring related services (for example, with more generic/specific input/output, but still with a related functionality), which can be identified by browsing a Web service network S-SLN. However, S-SLN discovery is always highlighted in real applications, both on precise service search and on automatic services collaboration.

The mainly advantages of constructing S-SLN are:

(i) Aiding in analysis. The semantically structured network of service makes service easier to find and collaboration. Identifying service semantic link network would automate part of the analysis process.

(ii) Reducing network complexity. As the size of Web service grows on the web, it becomes increasingly difficult to navigate or display the service space. Abstracting semantically structured service network can reduce the size and complexity of the network representation, making it more manageable with existing navigation and visualization techniques.

This paper focuses on how to discover S-SLN and discover dependency relationships among Web services by combining several dependency representation methods of graph including undirected graph structure Markov network and directed graph structure. We propose a methodology to discover service semantic link networks, which can help to identify related service resources on the basis of their inherent service relationship dependencies. We are interested in a particular problem where the input is service datasets with their historical invocations and output is an S-SLN, namely discovering S-SLN from a large amount of Web services.

The contribution of this paper is as follows:

(i) We incorporate SLN to construct Service Semantic Link Network S-SLN based on the dependency relationship between services.

(ii) Development of a graphical model based approach for automated discovery of S-SLN from service dataset.

(iii) We conduct preliminary experiments and performance studies, which verify the feasibility and effectiveness of our methods.

The rest of this paper is organized as follows: In section 2 we review related work, Section 3 introduces the problem definition and preliminaries related to our work. Section 4 describes the approach to S-SLN construction based on Markov structure. We show the effectiveness of the presented approach by experimental result in section 5. Finally, we summarize our conclusions and the future research in section 6.

2. Related Work

Most of the efforts in the domain of Semantic Web for Web service have been focused on service relationship automatic discovery for service collaboration (e.g. automatic Web service search and composition), using both manual and automated approaches.

Liu et al. A service discovery method was proposed for the collaboration based on Collaborated Semantic Link Network [2]. The network is constructed using services represented by Element Fuzzy Cognitive Maps (E-FCMs) and the similar and associated relations between E-FCMs as well. An automatic semantic relationship discovering approach for constructing the semantic link network was presented [3]. The basic premise is that the semantics of a web page can be reflected by a set of keywords, and the semantic relationship between two web pages can be determined by the semantic relationship between their keyword sets. The approach adopts the data mining algorithms to discover the semantic relationships between keyword sets. Dong et al. [4], for example, used clustering-based approach in which parameters of service operations are grouped into meaningful concepts, which are then used to find similar service operations based on similar parameters. However, this method provides only a limited solution and is unable to provide comprehensive service discovery based on the underlying semantics provided by services. Employing Semantic Web approaches such as ontological annotations could improve this approach [5].

Recently, these efforts have been extended to semantically structured network description and discovery of resources that are used to analyze, visualize and explore such a network for potentially improving the resource discovery process. By studying the intrinsic relationship between semantic communities and the semantic space of SLN, approaches to discovering reasoning-constraint, rule-constraint, and classification-constraint semantic communities are proposed [6]. Approaches make use of the semantic communities and the emerging semantic relations in a dynamic complex network of learning resources to support effective learning. Z. Huang and Y. Qiu propose a Citation Semantic Link Network (C-SLN) to describe the semantic information over the literature citation networks [7]. A framework of the construction of C-SLN is represented by integrating several NLP methods. The methods of aggregating a C-SLN and the algorithms of discovering opinion communities in a C-SLN are discussed. Multi-perspective exploration on the C-SLN can effectively find articles of high importance, aggregate the function of citations and detect opinion communities among scientific documents. The approach of mining recurring structures that represents an interesting area of the semantic web was presented [8]. The recurring structures could be mined by existing or novel knowledge discovery methods. The paper [9] applies automated text mining techniques to text-based communication to identify, describe and evaluate underlying social networks among online community members for automate discovering social ties that form community between members and presents a content-based method for automated discovery of social networks from threaded discussions, referred to as 'name network'. The 'name network' method can be used to study online classes and to look for evidence of collaborative learning in online learning communities. There have been a number of recent efforts to mining semantic networks in Bioinformatics e-Resources from the literature [10]. These efforts mostly rely on manual creation and are unable to cope with the huge influx of various electronic resources, which consequently result in their unavailability to the community. They present a text mining approach that utilizes the literature to extract and semantically profile bioinformatics resources. Resources are then either clustered or linked into a network, providing the users with a possibility to explore tools, services and datasets based on their relatedness, thus potentially improving the resource discovery process. A mining socio-semantic networks method for egocentric and polycentric queries in multi-dimensional networks is proposed [11]. The method allows fast search for objects in sufficient proximity of other object(s) where the proximity is defined in terms of multiple relationships between objects. The advantages of such an approach are high performance and high scalability in terms of size of multidimensional network. Authors present an approach to mining a semantic network of Bookmarks for Web Search and Recommendation based on a semantic similarity measure for URLs that takes advantage both of the hierarchical structure of the bookmark files of individual users, and of collaborative filtering across users [12].

Even though the application area is different in this work, some of the ideas are built on a common ground. As compared with our work, a methodology to discover S-SLN, which can help to identify related service resources on the basis of their inherent service relationship dependencies embedded in probabilistic models, is to be discussed.

3. Problem Definition and Preliminaries

Graphical models (Markov networks) are an important subclass of statistical models that possess advantages that include clear semantics and a sound and widely accepted theoretical foundation (probability theory). Graphical models can be used to represent efficiently the joint probability distribution of a domain. They have been used in numerous application domains, ranging from discovering gene expression pathways in bioinformatics to computer vision. In real-world service paradigms, the inherent semantic association dependencies between Web services can be discovered by gathering and mining their distributed historical invocations implied by SOAP messages [13]. Naturally, statistical analysis is one of the widely adopted approaches; one problem that naturally arises is the construction of a service relationship network based on the theory of graphical models.

In our problem, we believe that the structure of S-SLN represents the conditional independence relationships among the Web services. S-SLN is directed acyclic graph in nature in which the nodes represent Web services (variables), the links (arcs) signify the existence of direct causal influences between the linked variables, and the strengths of these influences are expressed by forward conditional probabilities. The structure of a Markov network graphically encodes a set of conditional independences among the variables in the domain. The Markov structure is adopted as the backbone framework of S-SLN. This is a basis of our following discussion. Knowledge of these independences is invaluable in a number of fields, especially those that rely more on probability distributions.

A solution to this problem is to learn an undirected graph structure based on probabilistic dependencies from service data, then to transform the undirected graph structure into a directed graph structure S-SLN. Therefore, we seek effective graphical representation of the dependencies embedded in a probabilistic model. A Markov network is an undirected graph whose links represents probability dependencies, while S-SLN is a directed graph whose directed links represent conditional probability dependencies. Learning Markov network is easier than learning S-SLN because it doesn’t need to find the direction of an edge. In this paper, we employ a Markov network to describe the service relationship dependencies between services based on the joint probability distribution. First, we learn the graphical structure of Web service. By mining historical invocations among Web services, we adopt a dependency analysis based Markov network learning approach to discovering the structure of Markov networks. Then, we transform found Markov network into a directed graph structure with the same joint probability, at the same time, the direction of links in directed graph just shows the semantic interaction relationship between Web services. So after finding a Markov network and annotating semantic links, an equivalent S-SLN can be obtained by representing the same joint probability.

As stated above, we adopt the probability dependency approach to construct the structure of S-SLN so that the probability based S-SLN can be the semantic model of representing the dependency relationship between Web services.

Definition 1 *S-SLN*

A service semantic link network S-SLN is a directed graph, denoted as S_s*(ServiceSet, LinkSet),* where S_s is the name of the S-SLN, *ServiceSet* is a set of Web services, and *LinkSet* is a set of semantic links in form of $S \xrightarrow{\alpha} S'$, where S, $S' \in$ *ServiceSet*, and α is a semantic factor representing a semantic relation between S and S'.

Definition 2 *Markov Network*

Let $U = \{\alpha, \beta, \ldots\}$ be a finite set of variables in the domain. A Markov network is an undirected graphical model which represents the joint probability distribution over$U$.

A Markov network of a domain is an undirected graph model that can be used (i) to represent the joint probability distribution of the domain, and (ii) to represent the set of conditional independences present in the domain. Each node in the graph represents one of the random variables in the domain, while the absence of edges encodes conditional independencies among them.

Following, we introduce some basic definitions of graph model and Markov network [15], which are the basis of our later discussion.

Definition 3 *Conditionally Independence*

Let $U = \{\alpha, \beta, \ldots\}$ be a finite set of variables with discrete values, let $P(\cdot)$ be a joint probability function over the variables in U, and let X, Y and Z stand for any three disjoint subsets of variables in U. X and Y are said to be conditionally independence given Z ,denote as $I(X,Z,Y)$ iff $P(x|y,z) = P(x|z)$, Especially, if $Z = \Phi$, $I(X,\Phi,Y)$ iff $P(x|y) = P(x)$ whenever $P(y) > 0$.

Definition 4 *Dependency Model*

A dependency model M is the set of triples (X,Z,Y) such that $I(X,Z,Y)$ is true, denote as $I(X,Z,Y)_M$.

Characterizing a given dependency model M by an undirected graph $G = (V,E)$ is desired, we have independence dependencies and conditional independencies by a graphical representation of a dependency model M. We mean a direct correspondence between the elements in U (of M) and the set of vertices in V (of G), such that the topology of G reflecting some properties of M. When this correspondence is established, we will make no distinction between U and V but will write $G = (V,E)$.

Let X , Y and Z are three disjoint subsets of V , if Z intercepts all paths between the nodes of X and those of Y , denoted as $< X|Z|Y >_G$, then this interception should correspond to conditional independence between X and Y given Z , namely, $< X|Z|Y >_G \Rightarrow I(X,Z,Y)_M$, and conversely, $I(X,Z,Y)_M \Rightarrow < X|Z|Y >_G$.

But there is often no way of using vertex separation in G to display all dependencies and conditional independencies. Therefore, the definition is given for weaker condition.

Definition 5 *D-map; I- map; P-map*

An undirected graph G is a dependency map (D-map) of M if there is a one-to-one correspondence between the elements of U and the nodes V of G , such that for all disjoint subsets X , Y , Z of elements we have

$I(X,Z,Y)_M \Rightarrow < X|Z|Y >_G$.

Similarly, G is an independency map (I-map) of M if $I(X,Z,Y)_M \Leftarrow < X|Z|Y >_G$.

G is a perfect map (P-map) of M if it is a D-map and I-map， $I(X,Z,Y)_M \Leftrightarrow < X|Z|Y >_G$.

A D-map guarantees that vertices found to be connected are indeed dependent in M . It may, however, display a pair of dependent variables as a pair of separated vertices. An I-map, conversely, guarantees that vertices found to be separated correspond to independent variables but does not guarantee that all those shown to be connected are in fact dependent.

Definition 6 *Minimal I-map*

A graph G is a minimal I-map, if deleting any edge of G would make G cease to be an I-map. Such a graph is called a Markov network of M .

Definition 7 *Markov Boundary*

A Markov blanket $BL_I(\alpha)$ of an element $\alpha \in U$ is any subset S of elements for which $I(\alpha, S, U - S - a)$ and $\alpha \notin S$ The set S is called a Markov boundary of α , denoted as $B_M(\alpha)$, if it is a minimal Markov blanket of α . Similarly, the Markov boundary of α in G, denoted as $B_G(\alpha)$.

Definition 8 *Chordal Graph*

An undirected graph $G = (V,E)$ is said to be chordal if every cycle of length four or more has a chord (an edge joining two non-consecutive vertices).

Definition 9 *Decomposable Relative*

A probability model P is said to be decomposable if it has a minimal I-map that is chordal. P is said to be decomposable relative to a graph G if the following two conditions are met:

(i) G is I-map of P .

(ii) G is chordal

4. Service Semantic Link Network Construction

4.1. *Service Markov Network*

In this subsection we address the problem of learning the structure of the service Markov network from data. Markov network is graphical statistical models whose structure can be represented by an undirected graph. A Markov network consists of two parts: an undirected graph (the model structure), and a set of parameters. An example Markov network is shown in Fig.1, for a domain containing eight variables. Learning such

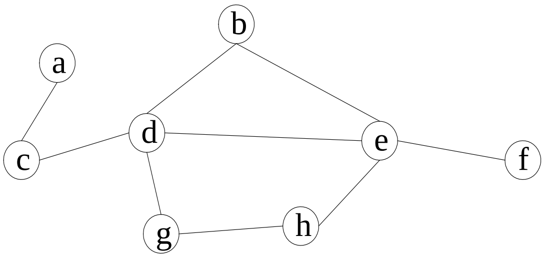

Fig. 1. An example Markov network

models from data consists of two interdependent problems: learning the structure of the network, and, given the learned structure, learning the parameters. In this work we focus on structure learning of the Markov network from Web service data, which is frequently the most challenging of the two tasks.

As stated in Section 3, in order to construct S-SLN from Web services, we will first learn a service Markov network structure described the inherent dependency relationships between services. In our work, Markov structure is adopted as the backbone framework of S-SLN for its effective representation service relationship.

In the following, we give some definitions of the semantic model for Web services.

Definition 10 *Service Cluster*

A service cluster S_C can be defined as follows:

$S_C = \{S_1, S_2, \ldots, S_n\}, 1 \le n \le M$

Where $S_1, \ldots, S_M$ represent M ordered services in a given domain. $S_n (1 \le n \le M)$ is a separate service represented as an operation in the corresponding WSDL document.

Definition 11 *Service invocation links*

A service invocation link

$S_l = (ID, S_1, S_2, \ldots, S_n), 1 \le n \le M$,

Where ID is thea service invocation linkervice invocation link, $(S_1, S_2, \ldots, S_n)$ represents an invocation trace of Web services in a request execution.

Definition 12 *Relationship Matrix of Service Invocation*

The relationship matrix of service invocation is a matrix

$$M = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

Such that:

- Each row in M corresponds to a Web service S_i of S_C ;
- each column in M corresponds to a service invocation link S_l ; and
- the element $a_{ij} = 1$ if S_i is invoked in a S_l , $a_{ij} = 0$ otherwise.

Historical service invocations are necessary to discover the inherent semantic association dependencies among Web services. Under the basic architecture of Web services, historical service invocations can be recorded by parsing the corresponding SOAP messages [13]. And in our previous work [14], we use a semantic soap header SSH for gathering necessary semantic interaction information between Web services, which is a collection of management attributes that is padded to every message flowing between Web services. The SOAP message with SSH is shown as Fig. 2. From the SOAP message, we can see the service invocation relationship between services.

```
<SOAP-ENV:Envelope
  Xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap envelope/"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Header>
    <SSH>
      <parent_SSH></parent_SSH>
      <message_id>1122-3551-5721</message_id>
      <message_name>GetZipcode</message_name>
      <source>zipcode.com</source>
      <target>officeplace.com</target>
    </SSH>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
   <GetPostofficeByCity>
      <City xsi:type=“xsd:string”>Beijing</City>
   </GetPostofficeByCity>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Fig. 2. A sample of SOAP message fragment with SSH

From the definition 6, we can know that a minimal I-map can represent the most of dependency relationship in the dependency model, so we desire to discover a service Markov network which is a minimal I-map. Every dependency model M satisfying symmetry, decomposition, and intersection has a unique minimal I-map $G_0 = (U, E_0)$ produced by connecting only those pairs (α, β) for which $I(\alpha, U - \alpha - \beta, \beta)_M$ is FALSE. $(\alpha, \beta) \notin E_0$ iff $I(\alpha, U - \alpha - \beta, \beta)_M$ [15].This theorem guarantees that if the graph meets the condition $I(\alpha, U - \alpha - \beta, \beta)_M$, then the graph is a minimal I-map. We use the existing *Edge-deleting algorithm* [16] which based on the above theorem to find Markov boundaries according to the given historical invocations of services, and further obtain the Markov network of services. Clearly, this Markov network is a minimal I-map and represents most of the dependencies of services in sample data.

During the process of Markov network discovery, we know that the most challenging and time-consuming is how to perform tests of conditional independencies. That is, we need to learn the Markov network structure by identifying the conditional independence relationships among the nodes. In our work, we take advantage of an important conclusion in information theory *conditional mutual information* [17] to test conditional independency. The mutual information between variables A and B measures the expected information gained about B, after observing the value of the variable A. While in Markov network, if two nodes are dependent, knowing the value of the one node will give us some information about the value of the other node. Hence, the mutual information between two nodes can tell us if the two nodes are dependent and if so, how close their relationship is. We use following equation to test $I(X,Z,Y)_M$:

$$I(X,Y|Z) = \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} P(x,y,z)\log\frac{P(x,y|z)}{P(x|z)P(y|z)}, (X,Y,Z)\in S_C \qquad (1)$$

Where S_C is service cluster, $P(x,y,z)$ is marginal probability, $P(x,y|z)$, $P(x|z)$, $P(y|z)$ are conditional probability. Marginal probability and conditional probability can be obtained by computing the corresponding frequencies for sample data (the matrix of service invocations) respectively.

$$P(x,y,z) = \frac{N_{x,y,z}}{N} \qquad (2)$$

where

$N_{x,y,z}$ is the number of columns in matrix M which x, y, z are nonzero.

N is the total number of columns in matrix M.

$$P(x,y|z) = \frac{P(x,y,z)}{P(z=1)} \qquad (3)$$

Similarly, we can compute $P(x|z)$ and $P(y|z)$.

$$P(x|z) = \frac{P(x,z)}{P(z=1)} \qquad (4)$$

$$P(y|z) = \frac{P(y,z)}{P(z=1)} \qquad (5)$$

Given threshold value ε , if $I(X,Y|Z) \le \varepsilon$, then $I(X,Z,Y)_M$ holds.

4.2. *Directed Clique Tree*

Our final goal is to find "what is connected to what" – that is, which nodes should be joined by directed links to represent the dependency invocation relationship between Web services. The semantics of S-SLN depends on its network structure and semantic relationship over the structure. Thus, the semantics of S-SLN demands a clear correspondence between the topology of a directed graph and the dependence relationships portrayed by it. This suggests that the directed S-SLN structure also can be learned by identifying the conditional independence relationships among the services. With Markov networks this correspondence was based on a simple separation criterion: If the removal of some subset Z of nodes from the network rendered nodes X and Y disconnected, then X and Y were proclaimed to be independent given Z . Directed graph use a slightly more complex separability criterion, called *d-separation*, which takes into consideration the directionality of the arrows in the graph. A path between nodes X and Y is closed, given some evidence C , if X and Y are conditionally independent given C . Based on this concept, all the valid conditional independence relations in a directed graph S-SLN probability distribution can be directly derived from the topology of the corresponding S-SLN.

We are to transform undirected graph into a directed graph based on the same joint probability distribution. That is, the joint probability distribution representation of Markov network is represented by the product form of conditional probabilities of directed graph. We can see that the directed graph representation captures a larger set of probabilistic independencies. From the chain rule of basic probability theory, we know that every distribution function can be represented as a product

$$P(x_1,\ldots,x_n) = P(x_1)P(x_2|x_1)\ldots P(x_n|x_1,\ldots,x_{n-1}) \qquad (6)$$

Based on this, we let the right part of the conditional probability be the parents and the left be the children, and the semantic links (directed edges) just represent some type of relationship between services, i.e. $S_j \xrightarrow{r} S_i$. So a directed graph S-SLN with conditional dependency can be discovered. For example, given an ordered service set $(S_1, S_2, \ldots, S_n)$ according to their invocation relationship, if we expand P in the order dictated by the service link, and use the conditional independencies encoded in the chain, we obtain

$$P(S_1,\dots,S_n)=P(S_1)P(S_2|S_1)\dots P(S_n|S_{n-1}) \qquad (7)$$

There is a way to find product form of joint probability distribution of Markov network by product division. This method for expressing the joint distribution is to divide the product of the marginal distributions on the edges (cliques) by the product of the distributions of the intermediate nodes (the intersections of the cliques).

Theorem 1 *Joint distribution of the chordal Markov network G_M can be written as a product of the distribution of the cliques of G_M divided by a product of the distributions of their intersections.*

Proof:

Let G_M be the discovered service Markov network.

G_M is I-map and chordal, we obtain the tree T of the cliques of G_M , and the ordered cliques $\{c_1,c_2,\dots,c_i,\dots\}$ that is consistent with T . For every $i>j$ we have a unique predecessors $j(i)<i$ such that $c_{j(i)}$ is adjacent to c_i in T . Namely, $c_{j(i)}$ separates c_i from $\{c_1,c_2,\dots,c_i,\dots\}$ in any such ordering. Based on the chain rule of probability theory, joint distribution of G_M can be represented as

$$\begin{aligned}P(x_1,x_2,\dots,x_n)&=\prod_i P(c_i|c_1,\dots,c_{i-1})\\&=\prod_i P(c_i|c_{j(i)})\\&=\prod_i P(c_i|c_i\cap c_{j(i)})\\&=\prod_i \frac{P(c_i)}{P(c_i\cap c_{j(i)})}\end{aligned} \qquad (8)$$

Based on the theorem 1, we use *a Graph Triangulation algorithm* [15] to obtain the cliques $\{c_1,c_2,\dots,c_i,\dots\}$ of service Markov network, thus, the joint distribution of the service Markov network can be represented as a product of the conditional probability as equation (6). Where

$$P(c_i\cap c_{j(i)})=P((c_1\cup c_2\dots\cup c_{i-1})\cap c_i) \qquad (9)$$

Let $c_1=\{x_1,x_2,\dots,x_i\}$, the $P(c_1)$ can be further represented as product of conditional probability based on the chain rule of probability theory,

$$P(c_1)=P(x_1)P(x_2|x_1)\dots P(x_i|x_1,\dots,x_{i-1}) \qquad (10)$$

At this point, we transform the joint probability of discovered Markov network into an equivalent product of conditional probability. Therefore, the joint probability P of discovered Markov network is expressible in terms of a product of conditional probability. As a result, we get a product of edge probabilities. The only requirement is that as we order the nodes from left to right, every node except the first should have at least one of its graph neighbors to its left. The service Markov network can be transformed into a directed clique tree.

4.3. *Semantic Links Annotation*

The directed clique tree transformed from the service Markov network is a directed graph, where its node is a set of Web services, and directed edge between Web services represents some kinds of service dependency relationship. But the specific type of service semantic relationship is not explicit, we need annotate semantic links to edge of directed clique tree such that it can represent specific service semantic relationships as a semantic link. A service semantic link is an ordered relationship between two Web services. It can be represented as a pointer with a type directed from one service (predecessor) to another service (successor). Types of semantic links can be defined according to the specific Web service application domain. Six types of Semantic Service Links are defined as follows:

(i) *SimilarTo link*

If two service S_i and S_j are similar in semantics, then there is a *SimilarTo* link $S_i \xrightarrow{sim} S_j$ between S_i and S_j , denoted as *sim*.

(ii) *Reference link*

A higher-granularity service $S_i=\{S_1,S_2,\dots,S_n\}$, in which S_i is a Web service that consists of more than one different component services S_j , $1\le j\le n$, then there is a *Reference* link $S_i \xrightarrow{ref} S_j$ between S_i and S_j , denoted as *ref*.

(iii) *Invocation link*

If the predecessor service S_i invocate the successor service S_j in a service composition process, then there is a *Invocation* link $S_i \xrightarrow{inv} S_j$ between S_i and S_j , denoted as *inv*.

(iv) *EqualTo link*

If two service s_i and s_j are equal in semantics, then there is a *EqualTo* link $s_i \xrightarrow{equ} s_j$ between s_i and s_j, denoted as *equ*.

(v) *OrthogonalWith link*

If two service s_i and s_j are independent of each other in semantics, then there is an *OrthogonalWith* link $s_i \xrightarrow{ort} s_j$ between s_i and s_j, denoted as *ort*.

(vi) *MemberShip link*

If two service s_i and s_j belong to the same class of services, then there is a *MemberShip* link $s_i \xrightarrow{mem} s_j$ between s_i and s_j, denoted as *mem*.

We use a service relationship mining method [18] for obtaining specific service semantic links among Web service. And a service semantic relationship matrix can be used to annotate semantic links in a direct clique tree.

Definition 13 *Service Semantic Matrix*

The service semantic matrix of *n* Web services $s_1, s_2, \ldots, s_n$ can be represented as:

$$M_s = (r_{ij})_{n\times n} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{bmatrix}$$

satisfying:

(i) $r_{ii} = \{equ\}$;

(ii) If $i \neq j$ and there exist $s_i \xrightarrow{sim} s_j$, $s_i \xrightarrow{ref} s_j$, $s_i \xrightarrow{inv} s_j$, $s_i \xrightarrow{mem} s_j$, then $r_{ij} = \{sim, ref, inv, mem\}$;

(iii) Otherwise, $r_{ij} = \{ort\}$;

where r_{ij} represents the specific semantic relationship between service s_i and s_j. The semantic relation from a resource to itself can be regarded as *equ*, so for any Web service s_i, $r_{ii} = equ$. If there does not exist dependent semantic links between two services s_i and s_j, $r_{ij} = ort$.

For a given S-SLN, the corresponding service semantic matrix is unique and vice versa. Assume a consistent S-SLN consists of *n* Web services, we can derive the reliable semantic relations of any two services from a service semantic matrix of *n* services. Annotating service semantic links in a directed clique tree is to add the semantic factor on the directed edges between two Web services. Therefore, we annotate semantic links to directed clique tree by scanning service semantic matrix and then we can obtain a final service semantic link network S-SLN.

As stated above, the main steps of S-SLN construction based on probabilistic dependency of services can be summarized as follows:

(i) A completely connected graph G_M for all Web services is constructed. Based on the conditional mutual information from historical invocations as sample data, some edges will be removed from G_M such that G_M is a service Markov network and represents most of the dependencies of services in sample data.

(ii) Triangulating the obtained service Markov network G_M for representing the joint distribution of G_M as the product form of conditional probabilities.

(iii) Transforming G_M into a directed clique tree based on the same joint probability distribution, and annotating semantic links in directed clique tree to obtain the final service semantic link network S-SLN.

Based on the above steps, the whole algorithm for S-SLN construction is described as follows.

```
Algorithm: ConS-SLN
Input: sample dataset of services
Output: an S-SLN
Construct a complete connected graph $G_0$ for Web services in $S_C$;
For each edge $(\alpha,\beta)$ in $G_0$
    If $I(\alpha, S_C-\alpha-\beta, \beta)$ Then delete $(\alpha,\beta)$;
End For
Triangulate $G_0$ by Graph Triangulation algorithm;
Obtain ordered cliques $C_1, C_2, \ldots C_m$;
Reorder services in $S_C$ as $\{S_1, S_2 \ldots, S_n\}$;
Transform the joint distribution of $G_0$ into the product form of conditional probabilities;
$k=1$;
While $(S_C \neq \phi)$
    For $i=k$ to $n$ Do
        Let $S_i$ be a root of S-SLN;
        Assign a marginal probability $P(S_1,\ldots,S_n)$ to $S_i$;
        For $j=i+1$ to $n$ Do
```

```
            If $S_j$ is dependent on $S_i$ Then annotating
              semantic links $S_j \xrightarrow{r} S_i$;
        End For
    End For
    $S_C = S_C - \{S_i\}$;
    $k = k + 1$;
Return an S-SLN
```

It is well known that the most time-consuming of the algorithm is conditional independencies tests. We mainly focus on analysis the time complexity of conditional independencies test.

As for a complete connected undirected graph G_0, there are $\frac{n(n-1)}{2}$ edges. We only need to do $\frac{n(n-1)}{2}$ times conditional independencies test, and the time complexity of conditional independencies test depends on size of joint probability table of sample Web services data. If there are N triples of the joint probability table of sample data, the time complexity of tests is $O(N^2)$. So as for the whole algorithm, the time complexity is exponential, the efficiency of the algorithm also depends on the size of sample data.

5. Experimental Results

In this section, we will present the preliminary experiments to test the performance of our proposed S-SLN construction approaches.

What our mainly focus is that whether the resulting S-SLN could represent the true dependency relationship between services accurately. So the accuracy of the resulting S-SLN is used to assess the quality of the outcome network. Because of the artificial data set has the disadvantage of a random network topology. Real data on the other hand, may come from non-random topologies and allow a more realistic assessment of the performance. Thus, assessment of the accuracy of the resulting S-SLN makes more sense for real-world data sets.

We conducted experiments using a publicly available test set QWS Dataset Version 2.0 [20, 21, 22]. The main goal of QWS Dataset is to offer a basis for Web Service researchers. The updated QWS Dataset Version 2.0 includes a set of 2507 real Web services that exist on the Web today. The majority of Web services was obtained from public sources on the Web including Universal Description, Discovery, and Integration (UDDI) registries, search engines, and service portals. We manually created a total of 200 service invocation links by human experiences from QWS Dataset. The part of the resulting S-SLN is shown in Fig 3.

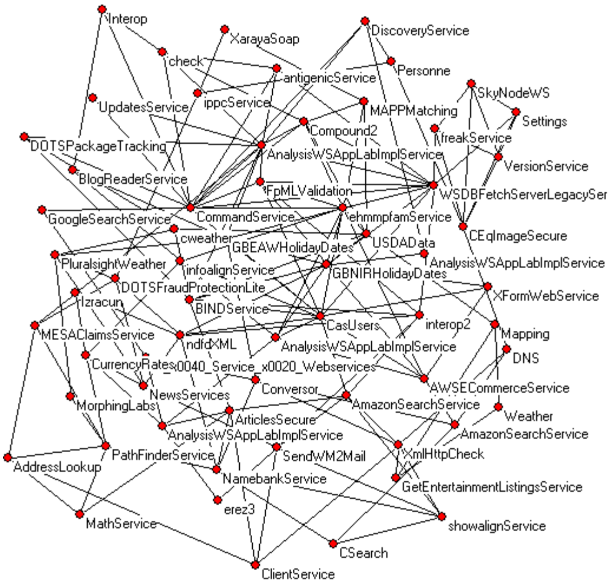

Fig. 3. The part of the resulting S-SLN

When we generated the test data, a relationship matrix of service invocation and a certain probability of invocations among given services was defined. Throughout the experiments, we tested the accuracy to evaluate the quality of the discovered S-SLN by searching the composite Web service which composed by several associated component services. And the composite service is predefined by expert experiences. The accuracy is defined as

$$acc = \frac{|R_s|}{|Q_s|} \tag{11}$$

Where $|Q_s|$ is the number of associated component services of requested composite Web service Q_s contains, $|R_s|$ is the number of returned service set R_s based on resulting S-SLN which are also appear in the Q_s. We searched 10 composite Web services for testing the accuracy of resulting S-SLN, and the number of component service that the composite service contains from at least of 5 to at most of 10. The experimental results are shown in Fig. 4.

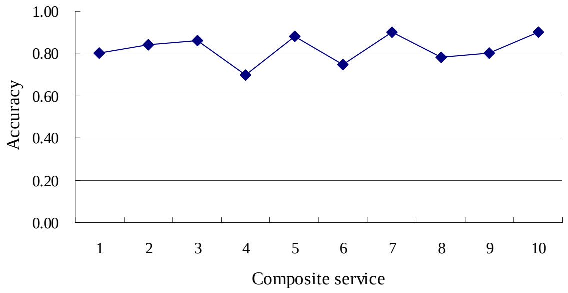

Fig. 4. Accuracy on service searching by the S - SLN

It can be seen that the accuracy value is high and basically stable. This means that the discovered S-SLN is much effective by our proposed method.

These experimental results verify the effectiveness of the discovered S-SLN. As a result, the S-SLN is an effective semantic model for representing the true dependency relationship between Web services. Generally, above experimental results and analysis show our proposed method for constructing the S-SLN from sample Web service data is efficient, feasible and practical.

6. Conclusions

In this paper we proposed and explored a graphical model based methodology for constructing a service semantic link network of semantically and functionally related Web services. The main motivation is to facilitate the service discovery and automatic collaboration process, which would improve the availability and utility of these services to the community.

We have presented a simple definition of semantically structured network S-SLN based on SLN and discussed an approach of S-SLN construction that are used to analyze, and explore such a network for potentially improving the service discovery process and service collaboration. The methodology revolves around the dependency relationship among services and the topology dependency structure of S-SLN in nature, which are the determinative aspects of the semantics of S-SLN. The work presented here demonstrates the potential of simple graphical method that constructing S-SLN based on Markov network. While because of the complex interrelationships between Web services, the structure of S-SLN will dynamically vary from time to time. A service should join and leave an S-SLN freely. This issue is at least worth investigating and will be part of our future work.

Our analysis has led us to advocate discovering for semantically structured network as a fruitful area of research, the problems should be amenable to straightforward knowledge discovery methods. Furthermore, even unsophisticated techniques could be beneficial, as relatively some existing results still offer some assistance where there currently is none.

Acknowledgements

This work was partially supported by the Doctoral Research Fund of Chongqing Normal University (No.11XWB021) and the Humanity and Social Science Foundation for Youth Scholars of Ministry of Education of China (No.11YJC72040003).

References

1. H. Zhuge, Y. Sun, The Schema Theory for Semantic Link Network, *Future Generation Computer Systems*.26 (3) (2010)408-420.
2. F. Liu et al, Discovery of Web services based on Collaborated Semantic Link Network, in Proc. *IEEE International Workshop on Semantic Computing and Systems*. 2008.
3. H. Zhuge et al, An automatic semantic relationship discovery approach, *in Proc. WWW 2004, (2004)* p. 278-279.
4. X. Dong et al, Similarity search for Web services, *In Proc. The Thirtieth international conference on Very large data bases*, (2004) pp. 372-383.
5. Wolstencroft et al, The myGrid Ontology: Bioinformatics Service Discovery, *International Journal of Bioinformatics Research and Applications*, 3(2007)pp. 326–340.
6. H. Zhuge, Communities and Emerging Semantics in Semantic Link Network: Discovery and Learning, *IEEE Transactions on Knowledge and Data Engineering*. 21 (6) (2009)785-799.
7. Z. Huang and Y. Qiu, A multiple-perspective approach to constructing and aggregating Citation Semantic Link Network, *Future Generation Computer Systems*. 26 (3) (2009)400-407.
8. S.R. Wolfe and R.M. Keller, Exploiting Recurring Structure in a Semantic Network, *In Proc. Workshop on*

Mining for and from the Semantic Web, Knowledge Directed Discovery (KDD) Conference, Seattle, WA, USA, 2004.

9. A. Gruzd, Automated discovery of social networks in online learning communities, *D. Eng. dissertation, Graduate College of the University of Illinois at Urbana-Champaign*, 2009.
10. H. Afzal et al, Mining Semantic Networks of Bioinformatics e-Resources from the Literature, *in Proc. Workshop on Semantic Web Applications and Tools for Life Sciences*, 2009.
11. A Troussov et al, Mining Socio-Semantic Networks Using Spreading Activation Technique, *in Proc. International Workshop on Knowledge Acquisition from the Social Web KASW08*, 2008.
12. L Stoilova et al, GiveALink: Mining a Semantic Network of Bookmarks for Web Search and Recommendation, *in Proc. LinkKDD'05*, August 21, 2005.
13. Rouached, M. et al. Web service mining and verification of properties: An approach based on event calculus, *In Proc. CooPIS, LNCS* , 4275(2006)408–425.
14. A. Zhao et al, "Semantic message link based service set mining for service composition," *in Proc. 5th International Conference on Semantics, Knowledge, and Grid*. (2009) pp. 338-341.
15. Pearl, J., Probabilistic reasoning in intelligent systems: Networks of plausible inference, *San Mateo CA: Morgan Kaufmann Publishers,* INC. 1988.
16. Y. He, W. Liu, Learning Bayesian network by first learning Markov network, *Computer Research and Development*, 39 (1) (2002)87-90 .
17. Cheng, J. Et al, Learning Bayesian networks from data: An information-theory based approach, *Artificial Intelligence*. 137 (1-2) (2002)43-90.
18. Shizhan Chen, et al, Building the Semantic Relations-Based Web Services Registry through Services Mining. *In 8th IEEE/ACIS International Conference on Computer and Information Science, ICIS* (2009: Shanghai, China) p. 736-743.
19. Y. Kun et al, Discovering semantic associations among Web services based on the qualitative probabilistic network, *Expert Systems with Applications*. 36 (5) (2009)9082-9094.
20. E. Al-Masri and Q. H. Mahmoud, The qws dataset. Web page. [Online]. Available: http://www.uoguelph.ca/~qmahmoud/qws/index.html.
21. E. Al-Masri and Q. H. Mahmoud, Qos-based discovery and ranking of web services, *in Proc. the IEEE International Conference on Computer Communications and Networks*, 2007.
22. E. Al-Masri and Q. H. Mahmoud, Investigating web services on the world wide web, *in Proc. the International World Wide Web Conference,* 2008.