

## 基于参数相似度的语义 Web 服务组合研究

汤全武<sup>1</sup>, 毕 利<sup>2</sup>

(1 宁夏大学 物理电气信息学院, 宁夏 银川 750021; 2 宁夏大学 数学计算机学院, 宁夏 银川 750021)

**摘 要:** 本文在分析现有相关研究基础上, 提出了基于参数相似度的信息类 Web 服务组合方法, 详细描述了基于本体的方法得到词汇相似度和基于人工智能规划 (AI Planning) 的自动信息类 Web 服务组合方法. 该方法可以根据用户需求直接生成服务组合序列, 整个 Web 服务的组合和调用完全绑定在一起, 直接将组合结果返回给用户, 同时保证组合的效率和质量.

**关键词:** AI Planning; 语义 Web 服务; Web 服务组合; 参数相似度

**中图分类号:** TP311.13

**文献标识码:** A

**文章编号:** 1000-7180(2012)05-0113-06

## Research of Semantic Web Service Composition Based on Similarity of Parameter

TANG Quan-wu<sup>1</sup>, BI Li<sup>2</sup>

(1 School of Physics and Electronic Information Engineering, Ningxia University, Yinchuan 750021, China;

2 School of Mathematics and Computer, Ningxia University, Yinchuan 750021, China)

**Abstract:** Based on the analysis of the existing related research, proposes similarity of parameter information web services composition method, describe based on ontology ways to get vocabulary similarity and the artificial intelligence (AI) planning of automatic information web service composition method. This method can according to user needs to produce a composition of services sequence, the entire web service composition and call completely bound together, results returned to the user, while ensuring composition the efficiency and quality.

**Key words:** AI Planning; semantic web service; web service composition; similarity of parameter

### 1 引言

Web 服务组合问题是 Web 服务系统架构中的一个重要部分<sup>[1]</sup>. 现有的服务组合技术按照静态服务组合、动态服务组合分为两类. 目前 Web 服务的组合方法大多是静态的、手工的, 这种组合方法费时且很难适应服务的动态性. Web 服务的自动组合是十分复杂的问题, 至今没有得到很好的解决, 目前一个比较好的解决方法是采用 AI Planning 来寻找 Web 服务的组合序列从而达到预期目标.

Sheila A. McIlraith 在《Adapting Golog for Programming the Semantic Web》一文中, 提出了将语义 Web 服务分类的思想. 文献[2]进一步将 Web

服务区分为信息类 Web 服务和动作类 Web 服务<sup>[2-3]</sup>.

信息类 Web 服务, 也叫做知识服务, 其主要特点是获取知识信息, 例如获取天气信息, 获取公交车换乘线路等. 动作类 Web 服务, 其主要特征是该 Web 服务的调用可以引起外部环境的变化. 例如预订飞机票、网上电子汇款等服务. 对比这两类服务, 信息类 Web 服务具有幂等性的特点, 可以多次重复调用.

### 2 信息类 Web 服务描述

在 Web 服务的语义描述的有关介绍中可知, IOPE 是 OWL-S 语义信息的核心部分, 这部分信息

收稿日期: 2011-09-03; 修回日期: 2011-10-27

基金项目: 国家自然科学基金项目(70961007); 宁夏自然科学基金项目(NZ1026)

可以归结为服务的“输入 (Input)”、“输出 (Output)”、“前提条件 (Precondition)”、“执行效果 (Effect)”四部分,简称“IOPE”.在“Precondition”所述的条件成立的情况下,通过输入参数“Input”,可以获得“Output”的输出,并且在该服务执行后,可以保证“Effect”所述的状态成立.

目前大多数对 Web 服务的语义描述都遵循 OWL-S 规则,将 Web 服务抽象定义为三方面的信息来描述: $WS=\langle S,C,Q\rangle$ ,其中  $S$  是基本描述,即服务名称、服务的商业实体以及服务的文本描述; $C$  是服务功能描述,即对服务的行为描述,包括服务的接口参数 Parameter、前置条件 PreCondition 和后置条件 PostCondition; $Q$  是对服务非功能属性的描述<sup>[4-5]</sup>.

信息类 Web 服务组合主要是对服务之间的输入和输出进行匹配,这里定义“Input,Output”来描述信息获取类 Web 服务.

**定义 1** 信息类 Web 服务(IWS):信息类 Web 服务定义为一个二元组  $IWS=\langle I,O\rangle$ ,IWS 是 Web 服务的名字; $I=\langle i_1,i_2,\cdots,i_n\rangle$  是服务输入参数的集合; $O=\langle o_1,o_2,\cdots,o_n\rangle$  是服务输出参数的集合.

### 3 基于本体的词汇相似度

**定义 2** 本体是如下形式的三元组  $\text{Ontology}=\langle C,P,SR\rangle$ ,其中  $C$  是某领域的概念集合, $P$  是领域内的属性集合, $SR$  是概念之间关系的集合, $SR=\{IsA,Partof,SameAs\}$ .

**定义 3** 参数相似度(Sim):设  $C_1$  与  $C_2$  分别是某个本体中定义的概念,且  $C_1,C_2\in C$ .若存在  $\langle C_1,C_2\rangle\in IsA$ ,则

$$\text{Sim}_{\text{concept}}(C_1,C_2)=\frac{1}{d+1}$$

若  $\langle C_1,C_2\rangle\in Parof$ ,则

$$\text{Sim}_{\text{concept}}(C_1,C_2)=\frac{\alpha\times(l_1+l_2)}{(d+\alpha)\times\max(|l_1-l_2|,1)}$$

其中  $d$  是 Ontology 中  $C_1$  到  $C_2$  的路径长度, $l_1$  和  $l_2$  是  $C_1$  和  $C_2$  分别所处的层次, $\alpha$  是相似度为 0.5 时  $C_1$  和  $C_2$  之间的距离,是个可调节的参数,一般  $\alpha>0$ .

若  $\langle C_1,C_2\rangle\in SameAs$ ,则

$$\text{Sim}_{\text{concept}}(C_1,C_2)=1$$

否则

$$\text{Sim}_{\text{concept}}(C_1,C_2)=0$$

计算参数匹配度实际就是计算服务请求与输入参数之间的参数相似度,这里我们假定  $IsA$  关系和  $SameAs$  关系的示例.设当前服务需求信息中包含的参数变量为  $\{P_1,P_5\}$ ,服务  $S_1$  的输入参数为  $\{P_1,P_2,P_3,P_4,P_5\}$ ,这些参数所对应的本体概念之间的关系如图 1 所示.以定义 1.3 为基础,得到匹配度如表 1 所示.

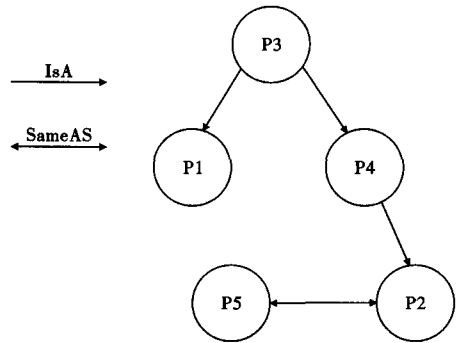


图 1 参数所对应的概念之间的关系示例

表 1 参数匹配度示例

|       | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $P_5$ |
|-------|-------|-------|-------|-------|-------|
| $P_1$ | 1     | 0     | 1/2   | 0     | 0     |
| $P_5$ | 0     | 1     | 1/3   | 1/2   | 1     |

由上面的定义 2 和 3 可知,服务参数都对应到本体中某个概念上,这个概念代表了参数的语义,而本体中的  $IsA$  关系、 $SameAs$  关系、 $Partof$  等关系描述了概念之间的继承性,因此由概念的语义就可以判断两个参数之间是否具有相似关系.以此为基础,参数匹配中所包含的核心问题可归结为对参数之间具有继承关系的相似度的定义与计算.

### 4 基于服务参数相似度的组合

#### 4.1 服务参数匹配

两个集合中的相似度计算模型,必须满足集合相似度计算的一些直观要求.一般来说,都给出两个限定:(1)一个集合和它本身的相似度为 1;(2)假设两个集合都有  $n$  个元素,其中  $m(m<n)$  个相同,又假设两个元素的相似度只能是 0(不同)或 1(相同),那么这两个集合的相似度应该是  $m/n$ <sup>[6]</sup>.

**定义 4** 服务参数语义相似度(Sim):假定服务  $S_1$  的输入或输出  $IOS_1$  有  $m$  个参数, $C_{11},C_{12},\cdots,C_{1m}$ ,服务  $S_2$  的输入或输出  $IOS_2$  有  $n$  个参数, $C_{21},C_{22},\cdots,C_{2n}$ ,若参数  $C_1$  有  $m$  个概念来描述  $S_{11},S_{12},\cdots,S_{1m}$ ,概念  $C_2$  有  $n$  个概念来描述  $S_{21},S_{22},\cdots,S_{2n}$ ,

则服务参数  $IOS_1$  和  $IOS_2$  的相似度定义如下:

$$\text{Sim}_{\text{parameter}}(IOS_1, IOS_2) = \begin{cases} \text{Sim}_{\text{concept}}(C_1, C_2), & IOS_1 \text{ 和 } IOS_2 \text{ 只包含一个参数} \\ \max \sum_{i=1}^m \sum_{j=1}^n \text{Sim}_{\text{concept}}(C_{1i}, C_{2j}) x_{ij} / \cdot \\ \max(|IOS_1|, |IOS_2|), & \text{其他情况} \end{cases}$$

$$x_{ij} = \begin{cases} 1, & C_{1i} \text{ 与 } C_{2j} \text{ 配对} \\ 0, & \text{其他情况} \end{cases}$$

$$\sum_{j=1}^n x_{ij} = 1, i = 1, 2, \dots, m$$

$$\sum_{i=1}^m x_{ij} = 1, j = 1, 2, \dots, n$$

要计算两个集合的相似度,通常的方法是先计算两个集合的所有元素两两之间的相似度,然后进行加权平均。

两个服务的输入和输出集合分别由一个或多个概念构成,计算相似度即将两个服务参数集合中的概念两两配对计算相似度,即得  $\text{Sim}(IOS_1, IOS_2)$ ,如果两个参数集合包含概念项不等,则会出现某些概念未能找到匹配的对项,可以忽略计算结果。

**定义 5** 服务参数匹配 (Match): 设  $IOS_1$  和  $IOS_2$  是服务  $S_1$  和服务  $S_2$  的对数集合,如果  $\text{Sim}_{\text{parameter}}(IOS_1, IOS_2) \geq \xi$ , 其中  $\xi$  是参数匹配的阈值,则称  $IOS_1$  与  $IOS_2$  是可匹配的,记为  $\text{Match}(IOS_1, IOS_2)$ 。

## 4.2 服务组合

首先,给出原子服务和组合服务的定义。*Web* 服务组合的过程实际上是一系列用户目标的实现过程。原子服务是指与用户只交互一次的服务;组合服务是为了满足特定的目标遵循一定的数据逻辑关系和控制逻辑关系,将原子服务或者其他组合服务组合起来,相互协作的服务,如图 2 所示。

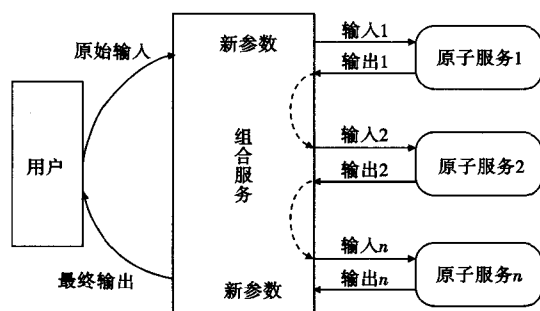


图 2 组合服务定义图式

**定义 6** Web 服务请求 (WSR): 用户提交的 Web 服务请求可以认为是提交给组合系统的一个

服务组合任务,这里形式化定义为一个三元组:  $WSR_K = \langle In_K, Out_K \rangle$ , 其中  $WSR_K$  是服务请求的名字;  $In_K$  是服务请求的输入集合;  $Out_K$  是服务请求的输出集合。

**定义 7** 服务组合序列 (WSC): 一个服务组合序列是指能满足某个服务请求  $WSR$  的一个服务序列  $WSC, IWS_1, IWS_2, \dots, IWS_n$ , 且该序列满足下列条件:

- (1)  $\text{Match}(WSR. Input, IWS_1. Input)$ ;
- (2) 该序列中任意的两个相邻的服务  $\text{Match}(IWS_i. Output, IWS_{i+1}. Input)$ ;
- (3)  $\text{Match}(IWS_n. Output, WSR. Output)$ 。

**定义 8** 组合满意度 (Sat): 组合满意度是指该服务组合序列  $WSC$  满足用户请求的程度,对于服务请求  $WSR$ , 一个服务组合序列  $WSC, IWS_1, IWS_2, \dots, IWS_n$  的组合满意度  $Sat$ :

$$\text{Sim}_{\text{parameter}}(WSR. Input, IWS_1. Input) \times \prod_{i=1}^{n-1} \text{Sim}_{\text{parameter}}(IWS_i. Output, IWS_{i+1}. Input) \times \text{Sim}_{\text{parameter}}(IWS_n. Output, WSR. Output)$$

根据以上的定义,服务组合过程就是由用户输入的请求,然后根据服务参数相似度的定义,首先找到服务请求的后继服务,以此类推,直至找到某个服务的后继服务的输出结果满足用户需求。所得到的序列即为服务组合的结果。由于一个服务可能具有多个后继服务,所以这些服务构成一个 Web 服务组合图 (WSC-Graph)<sup>[6-7]</sup>。

对于一个服务请求,在其对应的 WSC-Graph 中,从起点到终点的每条路径都是一个满足该服务请求的组合序列。最优的服务组合序列就是服务组合满意度最大的路径。那么如何找到服务组合满意度最大的路径,是解决服务组合问题的关键。

一般来说,一个 Web 服务组合图 WSC-Graph 中,所有服务之间的组合逻辑都可以由六个基本的控制结构而得到,这六种分别是顺序 (Sequence)、与分叉 (And-split)、与合并 (And-join)、或分叉 (Or-split)、或合并 (Or-join) 以及循环 (Iteration),如图 3 所示。

在信息类 Web 服务的组合过程中,每个服务都由输入参数和输出参数表示,输出参数是不变的,并且信息类 Web 服务不会改变外部世界的状态,这里不考虑循环关系。因此,在信息类 Web 服务组合的 WSC-Graph 中,由五种结构可以构成三种执行关系,包括顺序关系、并行执行关系和选择执行关系。

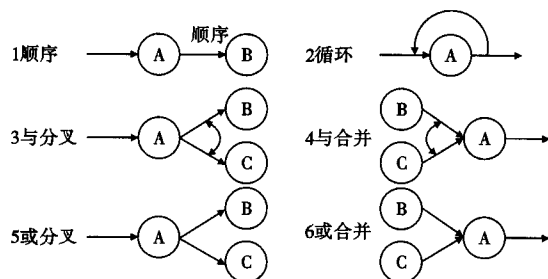


图3 服务组合逻辑图

(1) 顺序关系: 这种关系由图中的顺序结构构成, 表达了两个服务执行的时序关系, 即一个服务完成后, 下一个服务执行. 这种情况下的服务参数相似度计算为:

$$\text{Sim}_{\text{parameter}}(\text{SA. input}, \text{SB. input})$$

(2) 并行执行关系: 这种关系由一个与分叉 And-Split 和一个与合并 And-Join 组成, 表示两个或两个以上的服务在某个时刻后可独立的并行执行, 不需要其执行顺序, 但是需要这些服务要在某个时刻同步执行完. 这种情况下的服务参数相似度计算为:

$$\min(\text{Sim}_{\text{parameter}}(\text{SA. output}, \text{SB. input}), \text{Sim}_{\text{parameter}}(\text{SA. output}, \text{SC. input}))$$

(3) 选择执行关系: 这种关系由一个或分叉 Or-Split 和或合并 Or-Join 构成, 在或分叉节点, 选择其中一个服务执行. 这种情况下的服务参数相似度计算为:

$$\max(\text{Sim}_{\text{parameter}}(\text{SA. output}, \text{SB. input}), \text{Sim}_{\text{parameter}}(\text{SA. output}, \text{SC. input}))$$

在遍历 WSC-Graph 时, 就可以根据上面定义的三种关系计算服务参数度的方法, 来计算组合路径中最大满意度. 在 WSC-Graph 中, 如果某个服务有多个后继服务, 满足选择执行关系, 那么可以根据其后继服务参数相似度的大小, 将后继服务排序, 使服务参数相似度最大的排在前面, 从而形成优化的 WSC-Graph.

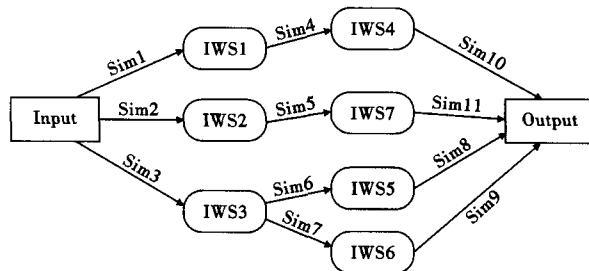
图 4(a) 显示了 WSC-Graph, 图中标有 *Input* 的方框表示服务请求 WSR 提供的输入参数 *Input*, 标有 *Output* 的方框表示 WSR 期待的输出参数 *Output*. 椭圆形表示一个服务, 有向边相连的两个服务是语义关联的, 其参数相似度为有向边的权值. 例如服务 *IWS1* 与服务请求的输入参数 *Input* 的参数相似度为:

$$\text{Sim1} = \text{Sim}_{\text{parameter}}(\text{WSR. input}, \text{IWS1. input})$$

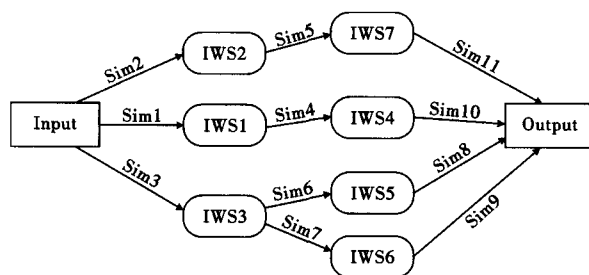
一般的连线表示了服务执行的顺序关系和选择

执行关系, 服务 *IWS3* 到 *IWS5* 和 *IWS6* 的并行执行是由弧线关联的两条连线表示的:

$$\text{Sim}_{\text{parameter}}(\text{IWS3. output}, \text{WSR. output}) = \min\{(\text{Sim6} + \text{Sim8}), (\text{Sim7} + \text{Sim9})\}$$



(a)



(b)

图4 WSC-Graph 及优化的 WSC-Graph

在 WSC-Graph 中的服务请求和已有的服务中, 如果有组合路径 *IWS1* → *IWS4* 的参数相似度  $\text{Sim2} + \text{Sim5} + \text{Sim11}$  大于 *IWS2* → *IWS7* 的参数相似度  $\text{Sim1} + \text{Sim4} + \text{Sim10}$ , 那么优化的 WSC-Graph 如图 4(b) 所示.

本节采用基于 AI Planning 的方法, 动态产生 WSC-Graph, 并且找到服务组合满意度符合用户设定的最小满意度阈值的那条路径. 这样既可以保证服务组合效率, 又可以兼顾用户满意程度.

## 5 核心算法

### 5.1 算法描述

根据上面的分析和定义, 本文将信息类 Web 服务组合看成是一个四元组  $\langle D, S_0, S_g, W \rangle$ , 其中:  $D$  是词汇集合, 包括 Web 服务组合所用到的词汇概念;  $S_0 \subset D$  表示初始条件集合, 包含若干个用户能够提供给系统的初始条件;  $S_g \subset D$  定义用户需求, 表示期望的目标状态;  $W$  是所有可调用的信息类 Web 服务的集合. 其中的 Web 服务都如定义所示. 即对于  $W_i \in W$ ,  $W_i = \langle \text{Input}, \text{Output} \rangle$ , 其中  $\text{Input} \subset D$ ,  $\text{Output} \subset D$ .

服务组合过程就是一系列目标获取的过程, 即

调用服务  $WS_1, WS_2, \dots, WS_n$  后,可以达到目标状态. 设服务组合初始集合为  $S_0$ , 服务组合算法如下:

算法 信息类服务组合算法

```
输入: 服务请求  $WSR(I_k, O_k)$ , 最小服务组合满意度
输出: 服务组合序列 WSC, 且 Sat
初始化  $GoalServices = \Phi$ ;  $Sat = 1$ ;  $Pre \subseteq I_k \cup S_0$ 
While  $O_k \neq \epsilon$  and  $Sat \geq \xi$ 
do
{
    select subGoal from  $O_k$ ;
    restGoal =  $O_k - subGoal$ ;
    if the subgoal has already been processed by subServices then
    {
        restServices to reach restGoal;
        GoalServices = subServices + restServices;
        Return GoalServices;
    } else if subGoal  $\subseteq$  Pre then
    {
        GoalServices = restServices;
        Return GoalServices;
    } else
    {
        There exists a set of Services can satisfy the SubGoal
    }
}
```

```
loop
forall Match( $W_i, Output, subGoal$ )
if Match( $W_i, Input, Pre$ ) then
{
     $W_i \leftarrow \text{MaxSim}(W_i, Output, O_k)$ 
    GoalServices = GoalServices  $\cup$   $W_i$ ;
     $Sat = Sat * \text{Sim}(W_i, Output, O_k)$ ;
     $O_k = O_k - subGoal$ ;
}
```

5.2 组合实例

本文给出一个“影讯的查询实例”应用实例,来说明上面的规划算法. 该服务组合的目的是帮助用户在北京市寻找一家电影院,提供正在公映的某类影片,用户需要输入“想看的电影及当前所在地”,经过服务组合过程,输出“电影院的名称,确切的播出时间,影院的位置,以及从当前位置到达该电影院的电子地图”.

分析用户需求,需要的服务功能包括:查询放映该影片的影院,查询到该影院的地理位置;公交换乘线路查询等. 根据上面的规划算法描述,  $\langle D, S_0, S_g, W \rangle$  中,  $S_0 = \{CustomerLocation; FilmType\}$ ;  $S_g = \{CinemaName; ProjectTime; Bitmap; FilmInfo\}$ ;  $W$  是现实中已经存在可供用户使用的相关功能的语义 Web 服务的集合,表 2 给出了这些服务的功能定义.

表 2 示例服务功能说明

| Web 服务功能描述                          | 服务的相关信息                                                                                            |
|-------------------------------------|----------------------------------------------------------------------------------------------------|
| IWS <sub>1</sub><br>影院查询 CinemaInfo | 提供基于电影名称的影讯信息数据查询<br>输入: 地理位置 Input; Location<br>输出: 附近电影院的名称 Output; CinemaName                   |
| IWS <sub>2</sub><br>影讯查询 FilmInfo   | 提供基于放映时间的影讯信息数据查询<br>输入: 电影院名称 Input; CinemaName<br>输出: 该影院近期播放电影的名称, 场次安排等 Output; FileName, Time |
| IWS <sub>3</sub><br>电子地图查询 GetMap   | 提供基于地理名称的电子地图的查询<br>输入: 电影院名称 Input; CinemaName<br>输出: 该电影院的电子地图, 行车路线 Output; Bitmap              |
| IWS <sub>4</sub><br>电影内容查询 FilmInfo | 提供基于电影名称的电影信息查询<br>输入: 电影名称 Input; FileName<br>输出: 电影信息, 如主要演员, 剧情介绍, 电影分类等 Output; FileInfo       |

假设用户 Input 输入为 {CustomerLocation = 海淀区牡丹园; FilmType = 战争片}, 则根据已有服务, 可以确定 IWS1  $\rightarrow$  IWS2  $\rightarrow$  IWS4  $\rightarrow$  IWS3 作为执行序列:

服务 IWS1 (CustomerLocation  $\rightarrow$  CinemaName): 输入“海淀区五道口”, 服务执行 IWS1 执行后, 输出“五道口电影院; 牡丹园电影院; 亚运村电影院; 朝阳区电影院等一系列电影院名称”. 如果在词

汇本体中如图 5 所示. 由于用户的输入“海淀区牡丹园”, 根据词汇相似度的计算方法, 其匹配程度最高的是“牡丹园电影院”, 其次是“亚运村电影院”和“五道口电影院”, 而“朝阳区电影院”匹配程度最小, 则根据匹配度的阈值设定, 选择前三个服务.

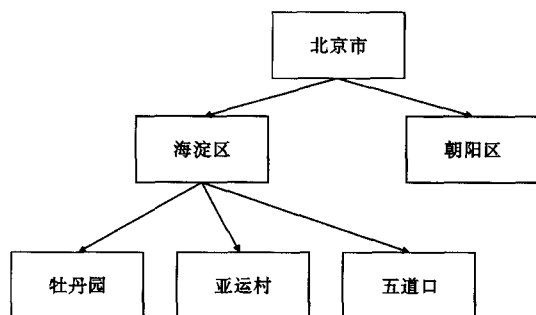


图 5 服务本体概念定义图示

服务 IWS2 (CinemaName  $\rightarrow$  FilmName, Time): 分别输入“牡丹园电影院”、“亚运村电影院”、“五道口电影院”, 得到电影名称和放映时间.

服务 IWS3 (CinemaName  $\rightarrow$  Bitmap): 分别输出三个电影院的电子地图及行车路线等信息.

服务 IWS4 (FilmName  $\rightarrow$  FilmInfo): 输入电影名称, 得到该电影的简介信息.

## 6 结束语

根据 Web 服务是否作用于外部世界, 本文将 Web 服务分为信息类 Web 服务和动作类 Web 服务, 提出了基于 AI Planning 的规划算法, 用于信息类 Web 服务的组合. 并提出了基于本体的概念相似度计算方法, 来计算参数的相似度, 在进行服务功能

匹配时, 考虑到了领域特征在服务组合中的影响, 根据语义信息来进行服务的动态选择.

## 参考文献:

- [1] 岳昆, 王晓玲, 周傲英. Web 服务核心支撑技术: 研究综述[J]. 软件学报, 2004, 15(3): 428-440.
- [2] 邱莉榕. 语义 Web 服务组合技术[M]. 北京: 中央民族大学出版社, 2009.
- [3] Klein M, Bernstein A. Searching services on the semantic Web using process ontologies[C]//Proceedings of the International Semantic Web Working Symposium (SWWS). Amsterdam: IOS Press, 2001: 159-172.
- [4] 毕利. 电子商务中 Web 的客户访问模式算法的分析与实现[J]. 微电子学与计算机, 2005, 22(10): 143-146.
- [5] 朱正东, 蓝荣贵, 陈彦萍, 等. 基于 MDA 的语义 Web 服务的组合与验证[J]. 微电子学与计算机, 2010, 27(11): 4-9.
- [6] 温嘉佳, 陈俊亮, 彭泳. 基于目标距离评估的启发式 Web Services 组合算法[J]. 软件学报, 2007, 18(1): 85-92.
- [7] 钟福金, 辜丽川, 张友华. 基于语义 Web 服务的电子政务模型研究[J]. 微电子学与计算机, 2010, 27(3): 144-147.

## 作者简介:

汤全武 男, (1965—), 硕士, 副教授. 研究方向为信息处理与计算机仿真技术.

毕利 女, (1968—), 硕士, 教授. 研究方向为数据库与信息系统集成技术.