

# Web 服务语义标注方法

许 斌, 李涓子, 王克宏

(清华大学 计算机科学与技术系, 北京 100084)

**摘要:** 为了对 Web 服务语义信息进行描述, 提出了一种 Web 服务语义标注方法。以现有的 Web 服务描述语言 (WSDL) 文件为基础, 充分利用 WSDL 中 XML schema 的语义信息与本体实体的相似性, 从 WSDL 文件中抽取语义信息, 并利用实体之间的名称相似性与结构相似性进行语义标注, 生成基于 OWL-S 的 Web 服务语义描述。用该方法对多个领域的 Web 服务进行了语义标注, 标注正确率在 92.5% 以上, 较好地解决了 Web 服务的语义标注问题。

**关键词:** Web 服务; 语义标注; WSDL

中图分类号: TP 311

文献标识码: A

文章编号: 1000-0054(2006)10-1784-04

## Web service semantic annotation

XU Bin, LI Juanzi, WANG Kehong

(Department of Computer Science and Technology,  
Tsinghua University, Beijing 100084, China)

**Abstract:** The current Web service description standard, Web service description language (WSDL), lacks semantic descriptions even though semantic Web services can improve the discovery and composition accuracy of Web services. Web service semantic annotations add semantic information to the service description. An annotation method was developed to first extract semantic information from the WSDL file according to the similarities between the XML schema and the ontology entity. Then, the extracted information is annotated using the name similarities and structural similarities between ontology entities. Finally, a semantic description of the Web service is generated in OWL-S format. Tests annotating dozens of Web services in various domains had an accurate rate of 92.5%.

**Key words:** Web services; semantic annotation; Web service description language

现有的 Web<sup>[1]</sup> 服务的描述标准采用基于 XML 的 WSDL 语言, 由于能够方便地描述 Web 服务的调用接口而得到广泛的应用; 但是, WSDL 语言着重描述的是 Web 服务运行的接口细节, 缺乏语义信息, 致使 Web 服务的发现和组装只能在 UDDI 中基于关键字匹配的方式进行, 查准率和查全率都不高。

语义 Web 致力于将语义引入到 Web 中, 使得 Web 中的每一个信息都具有明确的含义<sup>[2]</sup>, 目的是让计算机能够自动处理和集成 Web 上的信息。本体是语义 Web 的基础, 它的目标是给出领域的概念模型, 提供对该领域知识的共同理解。一些研究工作<sup>[3-5]</sup> 将本体用于描述 Web 服务, 提高了 Web 服务发现和组装的精度。

目前存在两种方法将本体用于 Web 服务的描述。第一种办法是直接定义一套描述 Web 服务的本体, 比如 WSMO<sup>[6]</sup> 或 OWL-S<sup>[7]</sup>, 其不足之处是它仅仅是 Web 服务描述的上层本体, 并不关注特定的应用领域, 而在 Web 服务的描述过程中需要再增加特定领域本体的概念标注。第二种办法是直接利用现有的 Web 服务标准 (WSDL 和 UDDI) 上增加语义信息<sup>[3]</sup>, 采用领域本体直接对 WSDL 文件进行标注; 文<sup>[5]</sup> 的办法是通过利用一个名为 schemaGraph 的中间结构来将 XML schema 和领域本体进行匹配, 达到将 WSDL 文件中的元素与本体的概念进行映射的目的; 文<sup>[8]</sup> 则通过制订规则来达到 XML schema 与 OWL 的转换目标, 但它在多个 XML schema 的情况下将带来多个本体的统一问题, 该方法并不是针对 Web 服务中的语义标注; 文<sup>[9]</sup> 通过利用 WSDL 中的 XML schema 隐含的语义信息, 制定转换规则来完成 WSDL 的领域本体概念标注, 但并未解决如何获得 OWL-S 描述的问题。

本文为了对 WSDL 文件进行自动语义标注并提高标注的准确度, 采用 OWL 描述的领域本体来增强 WSDL 文件的语义信息, 利用 XML schema 的格式信息与本体概念之间的相似性, 实现对 Web 服

收稿日期: 2005-10-05

基金项目: 国家自然科学基金资助项目 (60443002)

作者简介: 许斌 (1973-), 男 (汉), 海南, 讲师。

通讯联系人: 李涓子, 副教授,

E-mail: ljz@keg.cs.tsinghua.edu.cn

务的语义标注;并制定 WSDL 到 OWL-S 的转换规则,生成 OWL-S 格式的 Web 服务语义描述。

## 1 WSDL 与 OWL-S

WSDL 是互联网上现有 Web 服务描述的标准语言,它是一种规范,定义了如何用 XML 语法来描述 Web 服务。WSDL 描述了 4 种关键的数据: 1) 所有公共函数的接口信息; 2) 所有消息请求和消息响应的数据类型信息; 3) 所使用的传输协议的绑定信息; 4) 用来定位指定服务的地址信息。但是它仅仅从功能和语法层面来描述 Web 服务。

OWL-S 是针对 Web 服务的语义描述本体,它定义了一套基本的类和属性来声明和描述 Web 服务,其目标是实现 Web 服务的自动发现、自动调用、自动组装和自动互操作。OWL-S 用一个上层本体(upper ontology)来描述服务,把对一个服务的全面本体划分成 3 部分:对服务的描述 ServiceProfile,对服务与 WSDL 的绑定说明 ServiceGrounding,服务内部过程的描述 ServiceModel。

目前互联网上所有的 Web 服务都提供 WSDL 描述,但并未提供 OWL-S 描述。原因是 WSDL 描述可以方便地从 Web 服务的实现程序中自动生成,而 OWL-S 由于是语义描述,需要人工生成。因此提取 WSDL 中的语义信息,并进行语义标注,是自动生成 OWL-S 描述的关键步骤。

## 2 WSDL 中语义信息的抽取

Web 服务语义标注的目标是选择恰当的领域本体来对 WSDL 文件进行标注,也就是对 WSDL 文件中的每一个元素,选择最合适的领域本体概念来与之对应。而 WSDL 文件的 XML schema 中隐含了大量的语义信息,例如 WSDL 文件中的标记“complexType”和它的子元素的关系就非常类似于本体中“class”和它的“property”之间的关系,XML schema 的数据类型和 OWL 中的概念具有结构上的相似性。因此,从 WSDL 中抽取语义信息是本体标注的首要工作,其过程是:分析 WSDL 文件中数据类型 XML schema 信息,得到用 OWL 描述的本体概念。下面为了便于描述,WSDL 文件中 XML schema 的直接子结点称为“全局 element”,其他元素称为“局部 element”。语义信息抽取的主要步骤如下:

万方数据

成 OWL 的命名空间;该步骤保证 XML schema 中的所有元素的名称都可以在 OWL 本体中直接使用。如果出现命名冲突,即某两个元素的名称一样,则在名称后面分别加上不同数字加以区分。

步骤 2 XML schema 的复杂类型(complexType)转换为 OWL 本体中的类(owl: class): XML schema 和 OWL 本体的层次结构,决定了 XML schema 中的复杂类型和 OWL 本体中类的概念是极其相似的。

步骤 3 XML schema 中的简单类型(singleType)转换成 OWL 本体的类(owl: class): XML schema 中的简单类型具有多种形式,比如 list 类型、union 类型、enumeration 类型等等。这些形式表明简单类型同样具有复杂的结构,所以把简单类型转换为 OWL 本体中的类。

步骤 4 XML schema 中的全局 element 被转换成 OWL 本体中的类(owl: class): 由于全局的 element 经常使用一个匿名类型(复杂类型或者简单类型)作为自己的类型,结构和复杂类型相同,所以同样要转换成为 OWL 本体中的类。转换后的 owl: Class 的 ID 和该全局 element 的名称是一样的。

步骤 5 如果 XML schema 中的局部 element 的类型是 XML schema 中提前定义的简单类型,那么这个局部 element 转换为 owl: DatatypeProperty, element 的类型将转换为属性的值域(range), element 所属的复杂类型将成为属性的定义域(domain)。

步骤 6 如果 XML schema 中的局部 element 的类型是 XML schema 中新定义的类型,那么这个局部 element 转换为 owl: ObjectProperty。同时,如果该局部 element 在 XML schema 中属于两个不同的复杂类型,则转换后的属性使用 owl: unionOf 将两个类并起来作为它的 domain。

步骤 7 如果局部 element 的类型是匿名类型,那么要先将匿名类型转换为 owl: Class,然后将 element 转换为 owl: ObjectProperty。注意到新转换的类的 ID 和该 element 的名称是一致的,为了避免和 element 对应的属性 ID 冲突,在类的 ID 后面加上数字来加以区分。

相比文[3,5,6]方法,本信息抽取过程保留了 WSDL 文件中 XML schema 的更多有效信息,例如:不仅考虑了复杂类型和概念的对应关系,还考虑了它们的子元素和子概念之间的对应关系;增加

了对匿名类型的处理。抽取的语义信息以 OWL 的方式进行保存。

### 3 抽取信息的语义标注

WSDL 文件经过信息抽取后,生成了具有类和类之间关系的信息结构,对这些信息进行标注,将其表示为领域本体中的概念和关系,成为机器可理解的领域本体信息。

语义标注的过程实际上就是相似度计算的过程,计算抽取信息与领域本体中的概念之间的相似度,相似度最大者对应的概念为抽取信息对应的概念。因为 WSDL 中抽出的信息包含两方面的内容:实体(类和属性)的命名信息,以及 XML Schema 中隐含的这些实体之间的结构关系。因此,在相似度计算中可以利用这两部分信息的内容来计算与领域本体的相似度。

#### 3.1 名称相似度的计算

实体的名称相似度通过 Levenstein<sup>[10]</sup>编辑距离来计算。编辑距离是指将一个对象转化为另一个对象所需操作的最小代价,例如将“abc”转化成“abcd”的编辑距离是 1,因为只需要进行一个操作,就是增加一个字母“d”;而将“abc”转化成“ad”的编辑距离是 2,因为需要进行两个操作,即删除字母“c”和将字母“b”换成“d”。对于两个字符串  $s$  和  $t$  之间的编辑距离,其操作集合是:1)从  $s$  中删除一个字符;2)在  $s$  中加入一个字符;3)用  $t$  的一个字符代替  $s$  的一个字符。

因此,名称相似度为

$$\text{NameSim}(s,t) = 1 - \frac{L(s,t)}{\text{MaxLength}(s,t)}$$

其中  $\text{MaxLength}(s,t)$  表示字符串  $s$  和  $t$  中的最大长度, $L(s,t)$  则是  $s$  和  $t$  之间的编辑距离。例如“abc”和“abcd”的名称相似度为 3/4。

#### 3.2 结构相似度的计算

结构相似度的计算实际上就是计算这个实体与另一个实体与其他实体之间关系的相似程度,包含直接产生的关系以及间接的关系。本文只利用了实体与其他实体之间的直接关系,包括这个实体的属性、约束以及这个实体的子类 and 父类的上下位关系。

**定义 1** 实体的邻结点集合:一个实体  $E$  的邻结点集合  $\text{Neighbor}(E)$  为与  $E$  有直接连接的实体集合。对一个类而言,它的父类、子类、属性组成了它的邻结点集。对一个属性而言,它的 domain、range 组

成了它的邻结点集。

在结构相似度计算中,用实体的邻结点集合表示实体的结构特性,因此,两个实体的结构相似度可以通过计算它们的邻结点集和的最大名称匹配度平均值得到。

用  $E_r$  表示抽取出的实体, $E_t$  表示已存在本体中的某个实体。它们的邻结点集合分别为  $\text{Neighbor}(E_r) = \{S_1, S_2, \dots, S_n\}$  和  $\text{Neighbor}(E_t) = \{T_1, T_2, \dots, T_m\}$ 。

因此,结构相似度为

$$\text{StructureSim}(E_r, E_t) =$$

$$\max \left( \sum_{k=1}^q \text{NameSim}(S_{j_k}, T_{i_k}) \right) / q,$$

其中:

$$q = \min(n, m), \{j_1, j_2, \dots, j_q\} \in \{1, 2, \dots, n\},$$

$$\{i_1, i_2, \dots, i_q\} \in \{1, 2, \dots, m\}.$$

从上面定义可以看出,两个实体的结构相似度是用它们的邻结点集的最佳名称匹配度来表示的,即对于两个实体,从它们的邻结点集之间所有可能的名称匹配中选择匹配之和最大的一组,该组的平均值就是这两个实体的结构相似度。

#### 3.3 实体相似度的计算

结合实体的名称相似度和结构相似度,可以计算出实体之间的相似度。两个实体  $E_r$  和  $E_t$  的相似度为:

$$\text{Similarity} = w_1 \times \text{NameSim}(E_r, E_t) + w_2 \times \text{StructureSim}(E_r, E_t),$$

其中  $w_1 + w_2 = 1$ 。

$w_1$  和  $w_2$  的取值表明实体名称和结构对相似度计算的共享,该取值可以是经验值,也可以通过训练学习出来,本文取  $w_1$  的值为 0.6,  $w_2$  的值为 0.4。

在对抽取出的实体进行标注时,计算该实体与领域本体的实体相似度,则相似度最大的领域本体实体为该实体的标注信息。

## 4 OWL-S 的生成

通过抽取 WSDL 中的语义信息,并进行语义标注后,进一步就可以生成 Web 服务的 OWL-S 描述。如图 1 所示,基于 WSDL 生成 OWL-S 的转换关系如下:

1) WSDL 中的所有 input 消息部分(message part)转换到 OWL-S 的 Profile 中的 input。

2) WSDL 中的所有 output 消息部分(message

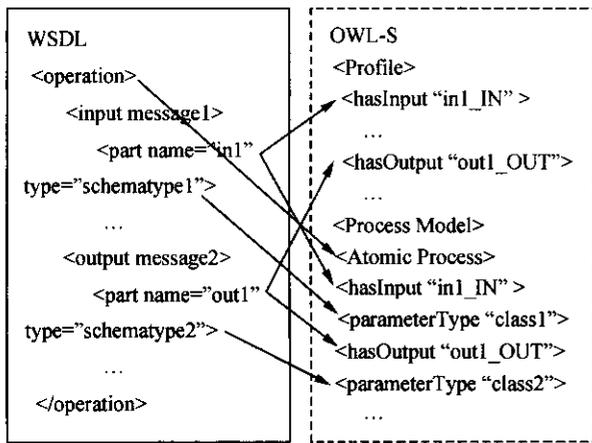


图1 WSDL 到 OWL-S 的转换关系

part) 转换到 OWL-S 的 Profile 中的 output。

3) WSDL 中的所有 operation 转换到 OWL-S 的 Process Model 中的 Atomic Process。

4) WSDL 中的所有 operation 的 input 消息部分(message part)转换到 OWL-S 的 Process Model 中的相应 Atomic Process 的 input。

5) WSDL 中的所有 operation 的 output 消息部分(message part)转换到 OWL-S 的 Process Model 中的相应 Atomic Process 的 output。

6) WSDL 中的所有 XML Schema 转换到 OWL-S 的 Process Model 中的本体概念。

以上转换规则中的前 5 条都可以直接进行转换。第 6 条规则是这个转换的核心规则,其转换所用的本体概念就是本文第 3 部分对抽取信息进行语义标注得到的本体概念。

## 5 实验及分析

从 XMethods.com 和 Amazon.com 中得到的 856 个 WSDL 文件中,选择了旅游领域中的 49 个文件和天气领域中的 16 个文件;并利用现有的旅游本体和天气本体<sup>[11]</sup>来进行标注。由于领域本体的通用性将影响本文方法的效果,因此为了修正本体的定义,从上述 65 个 WSDL 文件中随机挑选了 20 个,称为“修正文件”,通过对比它们与现有本体的不同来逐步修改本体;其余的 45 个文件称为“独立文件”。

### 5.1 评价指标

在实验之前,首先用上述两个领域本体对所有的 65 个 WSDL 文件进行人工标注,形成标准答案集。对于每个 WSDL 文件标注结果的评价指标定义如下:

$$\text{准确率} = \frac{\text{正确的标注数量}}{\text{总标注数量}}$$

其中“总标注数量”是 WSDL 文件被本文方法标注的数量,“正确的标注数量”是本文方法标注与人工标注一致的数量。

### 5.2 实验分析

表 1 中列出了所有 WSDL 文件进行机器标注后的平均正确率,从中可以看出:在每个领域中,“修正文件”的正确率比“独立文件”的正确率要高,因为“修正文件”的 XML schema 在结构上与领域本体的类有很大的相似性,其被正确标注的可能性更高。“修正文件”与“独立文件”之间的结果差异表明领域本体制定的通用程度,如果领域本体能够较全面地归纳了领域内的各种概念,则机器标注的正确率会提高。但就整体而言,本文方法标注的平均正确率至少在 83.6% 以上,在领域本体逐步完善的情况下,正确率将至少在 92.5% 以上,明显比文[5]中方法的正确率要高。

表 1 标注结果的平均正确率

参量	正确率/%		
	修正文件	独立文件	平均
旅游领域	92.5	78.6	83.6
天气领域	96.4	79.5	86.5

## 6 结束语

Web 服务语义标注是当前语义 Web 服务应用的一个关键性问题。本文提出一种对 WSDL 进行 Web 服务语义标注的方法。首先分析了 WSDL 与 OWL-S 的关系;其次利用 WSDL 文件中的 XML schema 数据类型与本体概念之间的相似性,从 WSDL 文件的 XML schema 中抽取语义信息;然后利用实体的名称相似度和结构相似度对抽取信息进行语义标注;最后依据 WSDL 到 OWL-S 的转换规则,生成 Web 服务的 OWL-S 描述文件。通过实验将方法应用于旅游领域和天气领域的 Web 服务语义标注,证明该方法标注的正确率都在 92.5% 以上,较好地解决了 Web 服务的自动语义标注问题。

### 参考文献 (References)

- [1] Singh P M, Huhns N M. Service-Oriented Computing Semantics, Processes, Agents [M]. England: John Wiley&Sons, Ltd, 2005.
- [2] 梁邦勇, 李涓子, 王克宏. 基于语义 Web 的网页推荐模型 [J]. 清华大学学报(自然科学版), 2004, 44(9): 1272 - 1276.

配置管理子系统 STACMS (Software Testing Artifact Configuraiton Management System)。实验表明,LEBM 是一种实用的软件配置管理模型。

#### 4 结束语

本文提出的 LEBM 模型将各类软件工件统一抽象为配置项,通过配置项类型支持不同类型工件的定义,通过包含关系定义支持不同粒度软件工件的管理,通过维护配置项版本间的关系支持变更影响分析,可从配置项、配置、基线等多个层次、多个视点展现软件系统的演化过程。LEBM 综合了 Check In/Check Out 模型、组织模型、长事务模型、变更集模型特点,并吸取基于构件的 SCM 中的部分概念,是一种易于嵌入到各种 CASE 工具中的 SCM 模型。

#### 参考文献 (References)

- [1] Dart Susan. Spectrum of functionality in configuration management systems [R]. CMU/SEI-90-TR-11 ESD-90-TR-212, Carnegie Mellon University, 1990.
- [2] 徐晓春,李高健. 软件配置管理 [M]. 北京:清华大学出版社,2002. 19-44.  
XU Xiaochun, LI Gaojian. Software Configuration Management [M]. Beijing: Tsinghua University Press, 2002. 19-44. (in Chinese)
- [3] Nguyen T N, Munson E V, Boyland J T, et al. Molhado: Object-oriented architectural software configuration management [C] // Proceedings of the 20th IEEE International Conference on Software Maintenance (ICSM'04). Chicago: IEEE, 2004: 510-510.

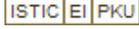
- [4] White B A. 软件配置管理策略与 Rational ClearCase [M]. 尤克滨,李纪华,王宁. 北京:人民邮电出版社,2003: 11-47.  
White B A. Software configuration management strategies and rational ClearCase [M]. You K B, Li J H, Wang N. Beijing: POSTS&TELECOM Press, 2003: 11-47. (in Chinese)
- [5] Nguyen T N, Munson E V, Boyland J T. Object-oriented, structural software configuration management [J]. *ACM SIGPLAN Notices*, 2004, **39**(10): 35-36.
- [6] Ruan L, Zhong Y. A new configuration management modal for software based on distributed components and layered architecture [C] // Proc of the 4th International Conference on Parallel and Distributed Computing, Applications and Technologies, 2003(PDCAT'2003). Chengdu: IEEE, 2003: 665-669.
- [7] Feiler P H. Configuration Management Models in Commercial Environments [R]. CMU/SEI-91-TR-7, Carnegie Mellon University, 1991.
- [8] Mei H, Zhang L, Yang F Q. A software configuration management model for supporting component-based software development [J]. *ACM SIGSOFT Software Engineering Notes*, 2001, **26**(2): 53-58.
- [9] 王映辉,张世琨,刘瑜,等. 基于可达矩阵的软件体系结构演化波及效应分析 [J]. 软件学报, 2004, **15**(8): 1107-1115.  
WANG Yinghui, ZHANG Shikun, LIU Yu, et al. Ripple-effect analysis of software architecture evolution based on reachability matrix [J]. *Journal of Software*, 2004, **15**(8): 1107-1115. (in Chinese)

(上接第 1787 页)

#### 参考文献 (References)

- LIANG Bangyong, LI Juanzi, WANG Kehong. Web page recommendation model for the semantic Web [J]. *J Tsinghua Univ (Sci & Tech)*, 2004, **44**(9): 1272-1276.
- [3] Verma K, Sivashanmugam K, Sheth A, et al. METEOR-S WSDI: A scalable infrastructure of registries for semantic publication and discovery of Web services [J]. *Journal of Information Technology and Management, Special Issue on Universal Global Integration*, 2005, **6**(1): 17-39.
- [4] Hess A, Kushmerick N. Learning to attach semantic metadata to Web services [C] // Proc of the 2nd International Semantic Web Conference. Florida, USA: Springer, 2003: 258-273.
- [5] Patil A, Oundhakar S, Sheth A, et al. METEOR-S Web service annotation framework [C] // Proc of WWW2004. New York, USA: ACM Press, 2004: 553-562.
- [6] Herzog R, Lausen H, Roman D, et al. D10 v0.1 WSMO Registry [EB/OL]. 2004. <http://www.wsmo.org/2004/d10/v0.1/>.
- [7] Martin D, Burstein M, Hobbs J, et al. OWL-S: Semantic markup for Web Services [EB/OL]. 2004. <http://www.daml.org/services/owl-s/1.1/overview/>.
- [8] Ferdinand M, Zirpins C, Trastour D. Lifting XML schema to OWL [C] // Proc of the 4th International Conference on Web Engineering. Munich, Germany: Springer, 2004: 354-358.
- [9] ZHANG Duo, LI Juanzi, XU Bin. Web service annotation using ontology mapping [C] // Proc of 1st International Workshop on Service Oriented System Engineering. Beijing, China: IEEE, 2005: 235-242.
- [10] Kleiweg P. Levenshtein [EB/OL]. 2003. <http://www.let.rug.nl/~kleiweg/lev/levenshtein.html>.
- [11] ZHANG Po. Travel Ontology:: Schema Details [EB/OL]. 2005. <http://www.schemaweb.info/schema/SchemaDetails.aspx?id=236>.

# Web服务语义标注方法

作者: 许斌, 李涓子, 王克宏, XU Bin, LI Juanzi, WANG Kehong  
作者单位: 清华大学, 计算机科学与技术系, 北京, 100084  
刊名: 清华大学学报(自然科学版)   
英文刊名: JOURNAL OF TSINGHUA UNIVERSITY (SCIENCE AND TECHNOLOGY)  
年, 卷(期): 2006, 46(10)  
被引用次数: 6次

## 参考文献(11条)

1. Singh P M, Huhns N M [Service-Oriented Computing Semantics, Processes, Agents](#) 2005
2. 梁邦勇, 李涓子, 王克宏 [基于语义Web的网页推荐模型](#)[期刊论文]-[清华大学学报\(自然科学版\)](#) 2004(09)
3. Verma K, Sivashanmugam K, Sheth A [METEOR-S WSDL: A scalable infrastructure of registries for semantic publication and discovery of Web services](#) 2005(01)
4. Hess A, Kushmerick N [Learning to attach semantic metadata to Web services](#) 2003
5. Patil A, Oundhakar S, Sheth A [METEOR-S Web service annotation framework](#) 2004
6. Herzog R, Lausen H, Roman D [D10 v0.1 WSMO Registry](#) 2004
7. Martin D, Burstein M, Hobbs J [OWL-S: Semantic markup for Web Services](#) 2004
8. Ferdinand M, Zirpins C, Trastour D [Lifting XML schema to OWL](#) 2004
9. ZHANG Duo, LI Juanzi, XU Bin [Web service annotation using ontology mapping](#) 2005
10. Kleiweg P [Levenshtein](#) 2003
11. ZHANG Po [Travel Ontology::Schema Details](#) 2005

## 相似文献(10条)

### 1. 学位论文 钟凯 [基于OWL-S的Web Services语义标注技术研究](#) 2006

传统的Web服务技术(SOAP、WSDL和UDDI等)缺乏机器可理解的语义,限制了Web服务的自动化。如何将Web服务内容(包括服务的用途、功能以及如何访问)表示为机器可处理的形式,即带有语义是所谓下一代的Web服务——语义Web服务的中心问题。

本文在借鉴现有相关研究成果的基础上,提出了一个基于Web服务标记语言OWL-S的Web服务语义标注框架WSAnnotator。该标注框架能支持原子Web服务以及由原子服务经“顺序”执行方式组合而成的简单复合Web服务的语义标注。原子服务的标注以服务的WSDL文档为基础,利用OWL领域本体中的词汇对WSDL文档中的XML词汇进行标注,半自动添加必要的服务描述信息,并基于一组预定义的“从WSDL到OWL-S”映射规则自动产生服务的OWL-S描述(包括Service本体及其Process、Profile和Grounding子本体实例)。简单复合服务的标注基于已标注的原子服务,其OWL-S描述按一定的规则与算法步骤,并利用OWL-S的“Sequence”构造子通过组合原子服务的OWL-S描述而自动产生。框架在一个原型系统中得到了有效的实现,案例研究表明了本文提议的方法是可行的。

与现有的Web(原子)服务语义标注方法/工具相比,WSAnnotator作为一个半自动的Web服务标注框架具有以下特色:(1)不仅能支持原子服务的语义标注,而且也能支持基于顺序执行方式的简单复合服务的语义标注;(2)能产生基于标准Web本体语言的Web服务OWL-S描述;(3)Web服务的语义描述更为完整。总之,本文研究成果对推动Web服务自动化具有一定的实际意义。

### 2. 学位论文 董熙 [Web Service语义标注方法的研究](#) 2006

本课题的研究受到国家自然科学基金项目“电子商务服务的分布式模型的研究”(项目编号90204010)的资助(2003.1~2005.12)。

WSDL已经使得Web服务中信息的交换成为可能,但是由于工业界过于注重具体实现,没有预见Web服务中的语义发现与发布的需要,所以当前的Web服务不够成熟。但是也不可能扔掉以前所有的成果,重新构建一个系统,所以将WSDL语义化是当前Web服务的一个重要任务。

本文从数学角度给出了创建语义Web的关键点,以及语义Web的形式化定义,并展示了如何从语义Web中得到完备的语义信息,指出了一种语义有效性检查的机制,其中语义Web的纵向拓展是其所强化的重要根源。并且详细介绍了Web服务注解框架MWSAF。MWSAF的主要目的是半自动的匹配WSDL元素与本体,并用相关的本体在Web服务的WSDL描述中加入注解。在进行概念匹配时,有基于概念名称的和基于概念结构的两种匹配方式。

本文分析了原基于概念名称的匹配算法,指出了它在匹配概念时仅仅参考单一匹配算法,而没有考虑不同匹配算法综合效用的缺点。为此,在改进后匹配算法中,通过考虑不同匹配算法对被匹配概念名称的总体评价来确定最终的匹配度。其中特别考虑到了单个匹配算法对于被匹配概念名称所得的匹配度不仅仅反映了被匹配概念名称的语义联系的强弱,也反映了该匹配算法对于当前被匹配概念计算所得的匹配度的可信性和适用性。

本文分析了原基于概念结构的匹配算法,指出了它在匹配概念时没有考虑不同子概念的不同重要性的缺点。为体现不同的子概念的不同重要性,在改进后匹配算法中,为子概念赋不同的权值。同时也证明可以从子概念的使用频率来导出子概念的匹配权值,并推导出子概念权值和其使用频率之间的定量关系。

最后对论文工作进行了总结和展望,探讨了基于概念结构的匹配算法尚需深入研究的地方,也指明了其广阔的应用前途。

### 3. 会议论文 房江太, 黄映辉, 李冠宇 [基于WSDL-S的Web服务语义标注方法](#) 2008

基于关键词的Web信息检索技术的查全率与查准率欠佳,难以支持人们有效地获取所需的信息。对此进行改善的一种思路是直接对Web服务信息进行语义标注。本文提出了一种Web服务语义标注方法:以现有的Web服务描述语言WSDL为基础,利用外部的领域模型中所定义的语义概念对WSDL所描述的Web服务的能力和要求进行标注,生成基于WSDL-S的Web服务语义描述。这样Web文档的内容就能够被计算机自动处理,从而实现Web信息的语义检索。

#### 4. 学位论文 田培军 语义Web和本体关键技术的研究及应用 2009

Web是信息化时代到来的标志之一。Web是互联网上最重要的,也是与普通的用户最贴近的应用。自1989年诞生以来,Web不断更新发展,从最初的静态HTML页面,到现在的动态Web页面;从最初对内容的简单描述,到今天将内容的结构和表示分离,无论从形式上还是功能上都发生了很大变化。但到了20世纪90年代中期,随着Web信息量的膨胀,人类意识到缺乏自动处理Web海量信息的有效技术。因而人类面临着一个尴尬的境地:我们创建了Web这个庞大知识库,却无法充分而有效地利用它。针对这个问题,Berners-Lee于2000年提出了下一代Web的概念—语义网。语义网汲取了人工智能、哲学和逻辑等学科的研究成果,希望对Web上信息的表示和获取方式进行重大改进,以解决目前使用Web时存在的问题。本文涉及的工作正是顺应这一趋势,展开了对语义网相关技术的研究。

本文的研究主要包括以下两个方面:

(1)首先对本体论及语义Web等关键概念进行了阐述,主要包括本体的基本概念、构建方式、评估标准、语义Web体系结构等,在此基础上开发了一种基于本体的检验医学信息系统。该系统采用OWL语言描述检验医学领域的本体知识库,通过Jena推理机从本体库中推理、读取病人的相关信息,接着通过Java技术和开源框架对其做了相关仿真实验来证明其优势、可行性和正确性。

(2)其次,在对Web服务相关技术和研究现状进行深入调研的基础上,着重讨论了语义标注技术和服务匹配。首先对比了各种语义标注工具,并采用最新的语义标注规范SAWADL对标准的服务描述文件WSDL进行了语义标注,为服务匹配、组合奠定了基础。接着提出了一种基于服务模板的匹配方式,讨论了功能匹配、I/O接口匹配、服务质量匹配算法,在此基础上开发了一个服务匹配框架的原型。

最后总结了存在的问题以及今后需要改进的方向。通过本文的研究表明:语义网技术在语义检索、语义Web服务等方面的相关技术已经比较成熟,必将推动今后的企业实际应用向着智能化和自动化方向发展。

#### 5. 学位论文 张玮 基于语义标注的Web服务发现框架及实现 2009

Web服务作为一种新兴的Web应用模式和分布式计算模型,从根本上解决了企业之间及企业内部异构系统之间的互操作和互通信问题。然而现有的主流Web服务发现方法是采用基于UDDI框架的服务匹配,由于UDDI注册中心只能提供基于关键字的查询,并且缺乏对语义信息的支持,使得服务匹配是基于关键字和语法级的匹配,不能实现基于服务功能的匹配,从而造成服务查准率查全率较低。另一方面,WSDL描述语言是语法级的服务描述语言,主要提供Web服务的物理信息,即调用具体服务所需的技术细节,缺乏对服务功能的语义描述,因此不能很好的提供服务的功能信息。

针对目前Web服务中遇到的问题,研究界提出了将语义Web的技术应用到Web服务领域,为Web服务的描述提供语义信息,即语义Web服务(Semantic Web Services)。旨在实现服务发现的自动化,在服务发现阶段借助于本体语义标注和逻辑推理,加强服务描述信息的机器可理解性,可以实现服务的灵活匹配,提高服务匹配的质量。本文所作的研究如下:

1、论文比较全面地介绍了Web服务及其现有的服务匹配技术,语义Web等的相关知识背景,以及它们的最新发展状况。

2、设计并实现了基于语义的Web服务发现框架,主要包括以下功能模块:1)通过SAWSDL标准把相应的本体概念标注到WSDL中;2)扩展UDDI注册中心,使其能够支持服务的语义信息;3)语义Web服务匹配模块利用服务本体和领域本体提供的语义信息并结合词汇网络和语义推理实现服务的匹配。

3、研究基于语义的Web服务发现匹配算法,该算法采用了分层匹配的思想,首先通过根据词汇网络得出服务需求模板的部分匹配度,然后通过语义推理推出各词汇的语义匹配度,根据相应公式计算出Web服务最后的匹配度,判断是否与用户要求匹配。最后按服务匹配度显示,提高了用户请求与服务匹配的准确性与简效性。

#### 6. 期刊论文 张大陆,吕韬,ZHANG Dalu, LV Tao 基于概念频率的Web服务语义标注 - 同济大学学报(自然科学版)

2008, 36(1)

由于web服务描述语言(WSDL)过于注重信息交流的底层技术而忽略其在语义方面的功能,所以需要在WSDL文档中添加语义,即语义标注。提出基于概念频率的web服务语义标注方法,通过得到概念的不同属性的使用频率,进而转化成各属性在标注中的权值来提高语义标注的准确性,并且通过实验验证了改进后的算法相对于原算法在标注准确性方面有所提高。

#### 7. 学位论文 王鹏 基于本体的Web服务发现关键技术与模型研究 2008

近年来,Web服务以其协议标准化、低耦合以及平台无关性的优点,在应用集成方面得到了迅猛的发展。随之而来的问题是,如何在海量的Web服务中找到合适的服务,即服务发现,已经成为Web服务研究领域的热点问题之一。目前,Web服务发现机制主要采用基于关键字的查询,这种方法存在明显的不足,因为服务很难通过一两个关键字描述清楚,而且不同组织或个人对某个领域的理解不同,往往会导致服务发现的查准率和查全率低。

本文将本体引入到Web服务发现领域,用OWL-S本体描述Web服务,服务提供者可以用本体来标记服务,服务请求者也可以利用本体来表达查询请求,这样就可以通过匹配语义请求和服务描述来获得更精确的结果,使得匹配达到语义层次的深度。

本文在国家自然科学基金项目的支持下,对基于本体的Web服务发现进行了研究。首先,本文探讨了本体的构建技术,根据需要建立了船舶本体用以支持语义标注,并介绍了如何使用OWL-S本体语言来描述Web服务;其次,采用将OWL-S的Service Profile本体映射到UDDI中的方法扩展了UDDI注册中心,使之支持语义信息;然后,研究了基于本体的Web服务发现的匹配算法,采用分层匹配的原则,分别在服务分类、服务描述、10和QoS各层中采用不同的匹配算法,尤其是在服务的功能性匹配(10匹配)中,改进了基于平均关系距离(ERD)的服务匹配算法,体现出本体在Web服务发现中的优势。最后,本文提出了一个基于本体的Web服务发现系统的模型,并在此基础上实现了一个原型系统,通过一个案例对提出的算法进行了实验,分析了实验结果,证实了所提出方案的可行性。

本文对传统的Web服务发现进行了分析,在服务描述和服务匹配方面,利用本体对其进行扩展,使之支持语义,能从根本上解决传统Web服务发现查准率和查全率不高的缺点。本文提出的Web服务发现方法和技术对提高服务发现的效率有一定的理论和应用价值。

#### 8. 期刊论文 杨艳萍,谭庆平,YANG Yan-ping, TAN Qing-ping Web服务自动语义标注的本体定位方法研究 - 计算机

工程与科学2008, 30(4)

面向服务架构中,分布式网络计算的实现依赖于服务交互问题的有效解决。为此,服务接口必须采用机器可理解的方式描述,从而为服务的动态发现和组合提供底层支持。服务语义标注技术满足了上述需求,它是指通过共享域本体中机器可理解的元数据表示服务元素。本文将服务语义标注过程分解为域标注和概念标注两个阶段,重点针对域标注问题,并提出了一种基于机器学习的域标注算法,对实际服务的标注实验验证了该算法的有效性。

#### 9. 学位论文 杨艳萍 自动Web服务组合关键技术研究 2007

企业业务集成是信息领域的核心问题之一,Web服务是当前基于Internet构造跨企业分布应用的标准框架。Web服务组合技术用于实现服务模块之间的有效集成,成为衔接以Web服务为基础的信息设施和企业业务应用集成之间的桥梁。只有当应用程序和业务流程能够通过服务组合实现复杂的业务应用时,Web服务技术的全部潜力才能充分发挥。如何组合这些分布、自治的已有Web服务从而构筑新的企业业务应用成为软件工程领域一个新的热点问题。

本文以上述需求为背景,展开自动Web服务组合关键技术研究,主要工作包括:

(1)现状的总结与分析:首先综述了自动Web服务组合方法迄今为止的国内外研究进展,将现有Web服务组合方法分为三大类,即基于工作流技术的方法、基于智能规划的方法和基于程序综合的方法,介绍了每种方法的代表性研究,分析了各种方法的优缺点。

(2)自动Web服务语义标注方法:服务描述是服务组合的基础,服务语义标注是实现服务语义描述的一种有效手段。本文将Web服务标注分为域定位和概念映射两个阶段,针对现有的自动语义标注方法主要关注概念映射的问题,提出了一种基于机器学习的域定位算法MD-kNN400(Multiple Domains kNN for Ontology Discovery),有效地解决了Web服务语义标注的域定位问题,可作现有自动语义标注方法的有效补充。

(3)基于语义的高效服务匹配方法:服务发现在Web服务组合实现过程中占有重要地位,而服务匹配是实现服务发现的关键。本文提出了一种基于包含推理的语义服务匹配方法SMOSH(Services Matching On Services Hierarchy),SMOSH利用OWL本体的自动推理功能建立服务分层,并利用服务分层实现服务匹配。本文从理论上证明了SMOSH建立的服务分层能有效提高服务匹配的效率。实验结果表明,与已有的服务匹配方法相比,SMOSH既保持

了较好的匹配性能,同时能在更短时间内响应服务请求。

(4) Web服务语义相似度量机制:为了量化服务描述与服务请求之间的匹配程度、增强服务匹配的灵活性,需要将这种匹配程度通过一种量化机制更精确地表示出来。OWL概念相似度量是使用OWL本体概念进行标注的服务相似度量的关键。本文提出了一种结构化的OWL概念相似度量机制,该机制通过知识库查询获取概念描述的底层语义计算其知识容量,再根据两概念的共同知识容量计算它们之间的语义相似度。

(5) 高效的Web服务自动组合算法:服务组合问题是NP完全问题,本文提出了一种基于图的自动Web服务组合法ASC-Graph(Automatic Services Composition Based On Graph)。ASC-Graph由组合规划图构造阶段和组合解搜索阶段构成。通过构造组合规划图,能够在多项式级时间内判定服务组合问题的可满足性。在组合解搜索阶段,提出了一种基于启发式信息的搜索算法。模拟实验结果表明,ASC-Graph能有效解决大规模服务的自动组合问题。

(6) Web服务行为的形式化建模理论:Web服务分为无状态Web服务和有状态Web服务,有状态Web服务的接口各操作之间具有严格的逻辑和时序关系。本文基于有色Petri网语义提出了Web服务行为形式化模型Service-Net。Service-Net不但能有效刻画Web服务内部操作之间的控制依赖以及数据依赖关系,还能表达Web服务作为一个独立通信系统与外部环境的消息交互。Service-Net模型在服务行为建模时全面考虑了控制流与数据流对服务内部行为及交互行为的影响,从而支持精确的服务行为描述和分析,为服务行为交互正确性判定和基于行为的服务匹配提供形式化基础。

(7) Web服务交互协议正确性验证方法:有状态Web服务的组合是一种交互式组合,服务之间通过消息交互完成各自服务会话而产生完整的交互协议。交互模型的正确性是保证服务交互成功的关键。基于Service-Net,本文定义了服务交互的形式化模型,提出了衡量服务交互模型正确与否的准则,以及相应的基于有色Petri网理论的判定和分析方法。

(8) 为了将本文提出的Web服务行为形式化建模理论和服务交互正确性验证方法应用到具体的Web服务执行平台,本文提出了标准服务流建模语言BPEI, (Business Process Execution Language)元模型到Service-Net构造元素的模型转换规则,给出了BPEI的有色Petri网语义。

综上所述,本文针对目前自动服务组合技术中亟待解决的几个关键问题提出了有效的解决方案,对于推进自动服务组合技术的理论研究和实用化具有较高的理论价值和应用价值。

## 10. 期刊论文 [孙红俊, 范玉顺, SUN Hong-jun, FAN Yu-shun 业务过程执行语言流程语义标注方法研究 -计算机集成制造系统](#)2009, 15(3)

为提高 workflow 模型在分布式执行过程中的自动化程度,解决语义异构问题,提出了一种基于本体的业务过程执行语言语义标注方法,构建了业务工程执行语言流程树和Web本体描述语言领域本体树。针对业务工程执行语言流程中的每个概念,分别从三个角度计算其与Web本体描述语言领域本体中所有概念之间的语义相似度,并选取与该概念复合相似度最大的概念建立映射关系。通过语义消歧过程,消除业务工程执行语言流程中存在的语义异构,并生成语义消歧树,进而基于一系列映射规则,将语义消歧树转化为Web服务本体描述语言描述的语义 workflow 模型。最后给出应用实例,并采用F1-measure对所提出的语义消歧方法进行评估。

## 引证文献(6条)

1. [操牡丹, 何前锋, 王柏 中医药方剂相似度模型](#)[期刊论文]-[计算机工程](#) 2009(16)
2. [龚小勇 基于QoS的Web服务发现与组合框架](#)[期刊论文]-[重庆电子工程职业学院学报](#) 2009(3)
3. [毕强, 韩毅, 牟冬梅 基于知识地图的多领域本体语义互联研究](#)[期刊论文]-[情报科学](#) 2009(3)
4. [龚小勇, 朱庆生, 武春岭, 李林 基于参数推导图自动构建Web服务组合流程](#)[期刊论文]-[计算机集成制造系统](#) 2008(9)
5. [解丹, 应时, 曹虹华, 贾向阳, 崔华, 肖波 基于语义的服务资源描述模型RDF4S](#)[期刊论文]-[武汉大学学报\(理学版\)](#) 2008(1)
6. [韩毅, 毕强 语义网格环境下数字图书馆知识组织的语义互联策略研究](#)[期刊论文]-[图书情报工作](#) 2007(8)

本文链接: [http://d.g.wanfangdata.com.cn/Periodical\\_qhdxxb200610037.aspx](http://d.g.wanfangdata.com.cn/Periodical_qhdxxb200610037.aspx)  
授权使用: 都晓东(wfqinghua), 授权号: 51a91657-5be0-4551-8092-9e1d00ee78c7

下载时间: 2010年10月28日