

基于 SQWRL 的语义 Web 服务发现

王 海¹, 范 琳², 李增智²

(1 西安交通大学 计算机系, 陕西 西安 710049; 2 西安邮电学院 计算机系, 陕西 西安 710061)

摘 要: 提出以扩展的语义 Web 规则语言 SQWRL 作为服务请求描述语言, 以 OWL-S 作为服务发布描述语言, 进行有效的服务发现. 给出了服务发现的实例, 通过实验, 证实了该方法具有实用性强、简单可靠的特点.

关键词: 语义 Web; 服务发现; SQWRL; OWL-S

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2010)09-0076-04

Semantic Web Service Discovery Based on SQWRL

WANG Hai¹, FAN Lin², LI Zeng-zhi¹

(1 Department of Computer, Xi'an Jiaotong University, Xi'an 710049, China;

2 Department of Computer, Xi'an Institute of Post and Telecommunications, Xi'an 710061, China)

Abstract: In this paper, the author argue that using the Semantic Query-Enhanced Web Rule Language as a service request description language and the OWL-S as service advertisement language to do effect service discovery. Through an real discovery example and the corresponding analysis, the method can be proved effective, reliable and easy-to-use.

Key words: semantic Web; service discovery; SQWRL; OWL-S

1 引言

随着国际互联网络的发展, Web 应用从局部化发展到全球化, 从集中式发展到分布式, 作为一种新兴的 Web 应用模式. Web 服务发现即根据服务使用者(service requester)提供的服务请求, 找到满足请求的服务. 作为实现服务共享、复用的关键支撑技术, 受到学术界和工业界的广泛关注.

针对以 UDDI 为代表的, 关键字匹配为依据的服务发现技术查全率与查准率不高的问题, 学术界已有不少相关研究. 文献[1]提出以描述逻辑 TBox 包含性检测为依据, 对服务请求描述与服务发布描述的输入输出进行相容性检测, 得到精确、包含、插拔和否定四种匹配类型. 文献[2]以描述逻辑的 ABox 一致性检测为工具, 对服务请求描述与服务发布描述做一致性检测, 得到精确、包含、插拔、相交和否定五种匹配类型. 描述逻辑开放世界语义推理要

求所有知识必须显示声明, 当服务功能较复杂时, 会造成知识库爆炸. 文献[3]通过对服务描述中输入输出和用户需求功能的测度来判断服务与用户需求的匹配程度, 将 UDDI 与功能匹配结合起来, 使得用户能够更好地进行服务查询.

服务发现的核心研究问题集中在服务描述及相应的发现算法. 服务描述又分为服务请求描述和服务发布描述; 服务发现算法也可以分为两个部分, 即服务匹配算法和结果评级算法(或称结果排序算法). 已有的服务发现研究工作主要集中在服务匹配算法, 相应提出了各种服务描述, 但并未将服务请求描述与服务发布描述的描述机制区分开来. 本质上这两者具有天然的不对称性, 服务发布描述通常是一个完整的服务描述, 全面而细致; 与此相反, 服务请求描述往往只描述服务请求者关心的部分, 并不是一个完整的服务描述. 考虑到这样的信息不对称性, 以往的服务发现算法假定服务请求与发布采用

相同描述^[1-3],以假想的完整服务描述作为服务请求描述,与发布服务描述中对应部分做比对以得到匹配结果.没有考虑到请求服务描述与发布服务描述之间的不对称性,特别是请求服务描述的特点,虽然查全率与查准率高,但是由于需要给出完整的假想请求服务描述的所有细节,方法难以实际应用.

本文根据目前语义 Web 的研究进展,提出以扩展的语义 Web 规则语言 SQWRL^[4]进行服务请求的描述,利用 OWL-S^[5]作为服务发布描述,进行有效的语义 Web 服务发现.

2 相关工作

2.1 语义 Web 服务

Web 服务作为 Web 上的资源,在语义 Web 技术的推动下,自然地发展成为语义 Web 服务,其主要特点是以语义 Web 提供的各种元语言对 Web 服务进行描述,以实现服务的自动发现、组合等目标.

目前比较有代表性的语义 Web 服务研究主要是基于描述逻辑的 OWL-S 和基于框架逻辑的 WSMO/WSML. OWL-S(即过去的 DAML-S)是美国国防高级计划署资助项目“DAML 计划”所提出的一种方法,可以将其看作 OWL 的一个应用.而 WSMO/WSML 是欧洲数字企业研究所资助的 SDK 项目所提出的方法,作为另一种选择.由于 WSMO/WSML 采用的是其特有的描述语言 WSML,针对它的相关推理算法及工具研究还比较匮乏,因此本文讨论方法以 OWL-S 为基础.

OWL-S^[5]是一种以 OWL 定义的 Web 服务本体.它提供了一套核心标记语言,以一种明确的、计算机可理解的方式描述 Web 服务的属性和功能. ServiceProfile 描述服务能做什么,类似于黄页信息,包括服务的描述性信息,以及服务的输入、输出、前置条件和效果. ServiceModel 用以刻画服务如何工作,而 ServiceGrounding 则给出服务如何使用的细节.

2.2 扩展的语义 Web 规则语言 SQWRL

SQWRL (Semantic Query-Enhanced Web Rule Language)^[1]是基于 SWRL 的扩展本体规则描述语言. SQWRL 并不改变 SWRL 的语义,并且它使用标准的 SWRL 语法.

在 SWRL 基础上, SQWRL 扩展了一个 built-ins^①库,可以高效地把 SWRL 转换为一个查询,进

行信息抽取.这些 built-ins 定义在 SQWRL 本体中.它有一个缺省的名字空间前缀 sqwrl.该文件的一个拷贝可以在标准的 Protégé-OWL 库里找到.

基本 SQWRL 功能:

假定我们有一个简单本体^②,以 OWL + SWRL 描述其中有一个类 Person,它有两个子类 Male 和 Female 以及函数属性 hasAge 和 hasName.另一个牵扯到的类是 Car,它通过属性 hasCar 与 Person 关联.

在此,假定需求是抽取所有年龄小于 25 岁的 Person,以及他们的年龄, SQWRL 语句如下:

• $\text{Person} (? p) \wedge \text{hasAge} (? p, ? a) \wedge \text{swrlb:lessThan} (? a, 25)$

上述查询将返回个体和其年龄的值对.

如果要描述所有拥有轿车的人, SQWRL 语句如下:

• $\text{Person} (? p) \wedge \text{hasCar} (? p, ? c)$

以 built-ins 库里提供的 sqwrl:select, 利用利用 SQWRL 推理引擎,可以得到满足上述描述的个体.例如将上面的 SQWRL 语句改写为

• $\text{Person} (? p) \wedge \text{hasCar} (? p, ? c) \rightarrow \text{sqwrl:select} (? p, ? c)$

可以得到所有有轿车的 Person 以及他们的 Car.

3 服务发现原型系统结构

服务发现从本质上即定位那些满足特定需求的机器可处理的服务描述.由此,可以将服务发现问题形式化的描述如下:

定义 1 WS Discovery: $(Q, O) \rightarrow R$, 其中 Q 表示服务请求约束, O 表示待发现的服务描述集合, R 即从 O 中返回的满足 Q 的服务描述集合.

照这定义,为了进行有效的服务发现,需要约定:

(1) 采用何种服务描述,此问题前述讨论已有结果,即采用 OWL-S;

(2) 如何表达请求约束,前面已有讨论,以 SQWRL 本体规则语言进行请求约束描述.

由于 OWL-S 本质上是 OWL 语言的一个应用,而 SQWRL 与 OWL 的描述是相容的,在 SQWRL 可以直接引用 OWL 中的描述.因此,服务发现系统类似一个数据库查询系统结构,如图 1 所示.

① <http://www.daml.org/2004/04/swrl/builtins>

② <http://swrl.stanford.edu/ontologies/examples/family.sqwrl.owl>

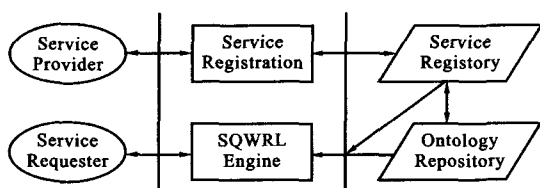


图1 服务发现系统结构

4 实例分析

4.1 网上书店

该例子来自 Amazon.com, 其服务以语义 Web 服务的形式对外提供, 在此不失一般性, 及论文篇幅限制, 我们选取四个典型的经过简化的服务作为测试用例, 如表 1 所示, 服务功能从其输入输出观察不言自明: (服务实际以 OWL-S 形式发布)

表1 网上书店服务

服务名	输入	输出
SearchByTitle	book: Title	book: Book
SearchByAuthor	book: Author	book: Book
GetBookPrice	book: Book	book: Price
GetBookAuthor	book: Book	book: Author

4.2 基于 SQWRL 的服务请求约束

例 1: 对于想对书目进行检索的用户, 可能想得到书目检索服务, 其服务请求如下:

```
service: Service(? s) ∧ service: isPresentedBy(? p, ? s) ∧ profile: hasInput(? p, ? input) ∧ process: parameterType(? input, ? it) ∧ tbox: isSubClassOf(? it, book: Book)
```

运行结果为 GetBookPrice 和 GetBookAuthor.

例 2: 对于获取给定书目信息相关的服务, 其服务请求如下:

```
service: Service(? s) ∧ service: isPresentedBy(? p, ? s) ∧ profile: hasOutput(? p, ? output) ∧ process: parameterType(? output, ? it) ∧ tbox: isSubClassOf(? it, book: Book)
```

运行结果为 SearchByTitle 和 SearchByAuthor.

例 3: 对于获取给定书目的价格的服务, 其请求约束如下:

```
service: Service(? s) ∧ service: isPresentedBy(? p, ? s) ∧ profile: hasInput(? p, ? input) ∧ profile: hasOutput(? p, ? output) ∧ process: parameterType(? input, ? it) ∧ tbox: isSubClassOf(? it, book: Book) ∧ process: parameterType(? output, ? ot)
```

```
∧ tbox: isSubClassOf(? ot, book: Price)
```

运行结果为, GetBookPrice.

4.3 结果分析

上述三个实例, 恰恰反映了三种典型的服务请求类型:

(1) 给定输出信息的服务请求, 目标是发现那些得到给定类型输出信息的有关服务, 如例 1, 其结果就是得到输出为给定书目的服务集合;

(2) 给定输入信息的服务请求, 目标是发现那些与给定类型输入信息有关的服务. 如例 2, 即发现那些与书目有关的服务;

(3) 同时给定输入输出信息的服务请求, 这类请求的目的很明确, 就是要得到特定的服务. 如例 3, 其结果就是得到了按照给定书目查询书价的服务.

以 SQWRL 语言描述服务请求, 发现 OWL-S 描述的语义服务, 较以往的方法具有明显的优势:

(4) 充分考虑了服务请求者与服务发布者对服务相关信息掌握上的不对称性, 将服务请求描述与服务发布描述的描述机制分离开来. 服务请求者不需要构建完整的 OWL-S 服务描述, 只需给出其相关部分的请求约束即可, 这样做既合理, 又方便, 提高了服务发现的易用性;

(5) SQWRL 语言具有丰富的表达能力, 其不仅能够直接引用 OWL 表达的本体, 还可以通过推理得到满足约束的实例以及本体包含的隐式信息. 且内置了丰富的数值、逻辑比较谓词以及简单的统计功能, 使用非常方便.

5 实验及评价

为了对本文提出的基于 SQWRL 的服务发现方法进行比较评价, 使用 OWL-S 服务发现测试集 OWL-S TC V2^[6]. 该测试集包含符合 OWL-S 1.1 推荐标准的服务描述 570 多个, 覆盖 7 个应用领域. 分别是教育、医疗保险、食品、旅行、通信、经济和军事. OWL-S TC V2 同时提供了 28 个测试查询集(即服务请求), 每个查询与 10~20 个不等的服务相关. 作者对这 28 个测试查询以 SQWRL 形式重新构建, 并进行了实验. 为了对比分析, 同时将这 28 个测试查询按关键字进行了提取, 进行了以关键字相似度为依据的发现. 并且, 也以文献[1]中的描述逻辑包含性检测为基础的服务发现算法进行了测试.

从图 2 可以看出, 以 SQWRL 构建服务请求, 比

基于包含性检测的算法小幅改善,但是相对关键字的方法,有大幅度提升,查准率的情况与此类似.主要原因在于,SQWRL 形式的服务请求,可以涵盖包含性检测,并同时支持 QoS 约束,考虑到使用 SQWRL Build-ins.因而相对包含性检测类方法,有小幅改善.关键字方法由于仅仅考虑词汇相似度,以关键字的自然语言语义为基础,在处理二义性及多义性上存在困难,因此其发现结果不够理想.查准率对比图如图 3 所示.

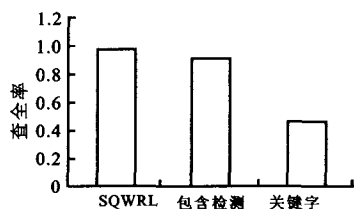


图 2 查全率对比图

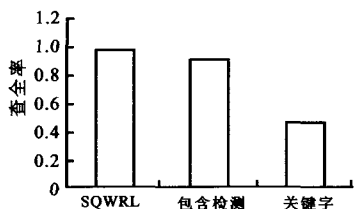


图 3 查准率对比图

6 结束语

语义 Web 服务发现,实际是对语义 Web 服务描述发现.服务描述分为发布描述和请求描述,这两者的关系并不对称.本文以语义 Web 最新研究成果

SQWRL 规则语言,作为服务请求描述语言,对 OWL-S 形式发布的服务进行发现.最后以 OWL-S TC V2 标准语义服务测试集合进行了实验,结果证明该方法查全率与查准率有所提升,并具有实用性强、简单可靠的特点.

参考文献:

- [1] Massimo Paolucci T R P, Takahiro Kawamura, Sycara K. Semantic matching of web services capabilities[C]// The semantic web ISWC2002, Proceedings of the first international semantic, web conference. Berlin: Springer-verlag, 2002:333 - 347.
- [2] Li L, Horrocks I. A software framework for matchmaking based on semantic web technology[C]// Proceedings of the 12th international conference on World Wide Web. New York, USA, ACM, 2003:331 - 339
- [3] 董媛媛,袁捷.基于 OWL-S 的 Web 服务匹配的研究[J].微电子学与计算机,2006,23(9):185 - 187.
- [4] O'Connor M J, Shankar R D, Tu S W, et al. Developing a web based application using OWL-and SWRL[C]// AAAI Spring Symposium. Stanford, USA, 2008.
- [5] Sem Web Central. OWL-S1.1 release[EB/OL]. [2009 - 10 - 11]. <http://www.daml.org/services/owl-s/1.1>.
- [6] Sem Web Central. Owls-TC:OWL-S service retrieval test collection[EB/OL]. [2009 - 10 - 11]. <http://projects.semwebcentral.org/projects/owls-tcV2>.

作者简介:

王海男,(1977-),博士研究生.研究方向为服务计算、面向服务的架构、语义 Web 服务.