

一种网络化软件的按需服务发现方法

曹步清^{1,2} 李 兵¹

(武汉大学计算机学院 & 软件工程国家重点实验室 武汉 430072)¹

(湖南科技大学计算机科学与工程学院 & 知识处理与网络化制造省教育厅重点实验室 湘潭 411201)²

摘 要 针对网络化软件的多种多样、个性化、模糊的用户需求,如何从海量、异构、动态的网络服务资源中找到最符合用户需求的、高质量的服务来构造网络化软件,以提高用户的体验质量,变得尤为重要。提出一种网络化软件的按需服务发现方法,首先根据具体领域,利用事先所建的领域本体对具体需求实例和服务信息进行语义标识,然后建立起“基本+功能+质量”的三阶段语义匹配模型。3 个阶段逐步递进,通过判断是否大于预定阈值,找到最符合用户需求的服务资源来构造网络化软件;最后,应用物流领域实例对方法的性能进行了比较测试,结果表明,所提方法提高了网络化软件的需求和服务之间的语义匹配精度,能够找到最佳的服务资源服务来构造网络化软件。

关键词 网络化软件,按需服务,服务质量,领域本体

中图法分类号 TP301 文献标识码 A

On-demand Service Discovery Method of Networked Software

CAO Bu-qing^{1,2} LI Bing¹

(School of Computer & State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, China)¹

(School of Computer Science and Engineering & Key Laboratory of Knowledge Management and Network-based Manufacture, Hunan University of Science and Technology, Xiangtan 411201, China)²

Abstract It is very important how to find the best service resources which meet to users' requirement and has high quality of service to construct networked software and improve users' quality of experience, according to diverse, individual and vague users' requirements of networked software. This paper proposed an on-demand service discovery method of networked software. First of all, according to specific domain, specific requirement instance and service resources were be marked by domain ontology. Then, a semantic matching model which consists of basic, function and QoS matching phase was built. This three phases gradually increase with predetermined-threshold judgment. Finally the best service resources which meet to users' requirement were found to construct networked software. Based on logistics field, the experiment was done for performance comparison testing, and the results indicated that this method can improve the matching precision of the requirement and service of networked software, and select the best service resources to construct networked software.

Keywords Networked software, On-demand service, Quality of service, Domain ontology

1 引言

当前,在 Web 服务、语义 Web 技术快速发展和日益成熟的基础上,以网络为基础的软件创新研究和以用户为中心的按需服务,推动着软件开发服务化的发展。软件从服务于系统到服务于网络,软件应用从“为我所有”到“为我所用”,其发展表现出明显的网络化趋势。面向服务的软件开发强调发现和重用已有的服务资源,通过聚集服务资源来满足用户需求。由此,一些新的软件开发理念和范型应运而生^[1]。网络化软件(Networked Software)是一类以网络为基础的复杂信息系统,其基本组成是纷繁复杂的网络资源(尤其是 Web 服务资

源),它通过发现、重用和聚集服务资源来满足用户需求,是这些新的软件开发理念和范型的典型代表。为用户提供多元化、个性化、高质量的服务,提高用户体验质量,是网络化软件追求的最高目标^[2]。

按需服务发现是网络化软件的实现基础,是提高其服务资源重用、加速服务技术应用的重要前提,它以查全率、查准率和查询效率等指标来评价服务发现的效果^[3]。因此,如何从纷繁复杂的网络资源(尤其是 Web 服务资源)中找到最能满足用户需求的、高质量的服务,即如何设计快速而有效的服务发现方法,以提高用户的体验质量,变得尤为重要。网络化软件的按需服务发现本质上是一个自动定位满足用户需求的

到稿日期:2011-02-21 返修日期:2011-05-30 本文受国家重点基础研究发展计划(2007CB310801),国家自然科学基金(60873083),湖北省青年杰出人才基金(2008CDB351),中央高校基本科研业务费专项资金(211274675),湖南省教育厅科学研究项目(09C401)资助。

曹步清(1979—),男,博士生,讲师,CCF 会员,主要研究方向为服务计算、软件工程等,E-mail:cao6990050@163.com;李 兵(1969—),男,博士,教授,博士生导师,主要研究方向为服务计算、复杂网络等,E-mail:bingli@whu.edu.cn(通信作者)。

服务过程,而服务匹配方法则是其难点所在。所谓服务匹配就是根据用户对服务接口的描述去匹配能够满足需求的服务。目前,服务匹配问题越来越受到关注。Massimo Paolucci等人于2002年提出了一种基于Web服务的语义匹配方法,其核心思想是,首先采用DAML-S服务描述语言来描述用户提出的服务请求以及服务提供者所发布的Web服务,然后通过服务与请求之间的语义匹配程度来确定不同的匹配级别(分成精确匹配、插入匹配、包含匹配和匹配失败4类)^[4]。此后很多文献中的思想都是在该经典算法的基础上扩展的。E. M. Maximilien提出了一种Web服务代理框架(WSAF)和QoS本体,以解决服务的动态发现问题^[5]。B. A. Soydan提出一种基于DAML注册库的QoS感知的语义Web服务发现方法,首先通过扩展UDDI来建立一个包含属性本体和评估数据的服务注册库,然后利用DAML语言的查询和操作等功能来进行基于属性本体的推理、服务的发布和发现以及优选等^[6]。Joyce El Haddad提出一种基于事务处理和QoS感知的Web服务选择算法,来优化语义服务发现和组合。该算法综合考虑用户的功能需求、事务型属性和服务质量属性,将用户的质量偏好作为权重,并从语义上将事务型属性定义为风险水平^[7]。吴健、吴朝晖等提出一种基于本体论和词汇语义相似度的Web服务发现方法,通过构建Web服务本体,给出一个明晰的Web服务发现研究对象^[8]。杨放春、苏森等提出一种混合QoS模型感知的语义Web服务组合策略,为解决混合QoS模型感知的语义Web服务组合问题,提出一种不确定多属性理论的全局优化决策算法。该算法分成两个部分:用于综合评估以实数型、区间型和语言型数据描述的服务质量信息;分布式与启发式相结合的算法执行框架,用来解决服务组合求解空间增大时的效率问题^[9]。孙萍、蒋昌俊提出一种利用服务聚类优化面向过程模型的语义Web服务发现方法,即利用本体和聚类技术,从服务的功能相似和过程相似两个层面对服务进行聚类预处理,从而在很大程度上降低服务的查找空间,提高服务的查找效率。进一步,针对服务请求中可能潜在的行为约束,建立基于Petri网的过程需求模型,将其与候选服务的过程模型进行比较,进一步筛选满足功能需求的可行服务,从而提高查找的精确度^[10]。另外,国际Web竞赛(WS-Challenge)^[11]就围绕自动服务发现与组合的性能评测,成功组织了6届国际比赛,竞赛的题目在语法级、语义级自动发现组合的基础上,逐渐发展到了最近的QoS敏感的自动服务发现与组合^[12,13]。

从以上研究可看出,利用本体对服务进行语义描述和标识,并设计高效的服务匹配算法来优化服务选择,已经成为一种广泛应用的方法。服务匹配算法设计的好坏,直接影响到服务发现的查全率和查准率,而基于本体的语义相似度度量与计算则是其关键所在。目前,基于本体的语义相似度计算方法研究非常热烈,形成的研究成果也非常丰富^[14],主要有基于距离的语义相似度计算方法^[15]、基于信息内容的语义相似度计算方法^[16]、基于属性的语义相似度计算^[17]以及混合式语义相似度计算方法^[18]。但是,这些方法绝大部分都从单一视角出发,会造成计算的片面性和不合理性,也将影响服务匹配的精度和质量,从而导致服务发现的效率不高。特别是,针对网络化软件的多种多样的、个性化的、模糊的用户需求,如何从海量的、异构的、动态的网络服务资源中找到最符合需

求的、高质量的服务来构造网络化软件,以提高用户的体验质量,变得尤为重要。为此,本文提出一种网络化软件的按需服务发现方法。首先针对具体领域,建立领域本体,并利用它们对功能和非功能的需求,以及服务注册库中服务的功能和质量信息进行语义标识。然后,综合考虑到领域本体的概念间结构关系、概念层次、同层中概念数量、概念间语义关系强度等因素,提出一种基于本体图的概念语义相似度度量与分析方法。接下来,利用已有的关键字匹配方法进行已语义标注的需求和服务的基本信息匹配,过滤掉与用户请求领域无关的服务;利用基于本体图的概念语义相似度匹配方法进行已语义标注的需求和服务的功能信息匹配,进一步过滤掉不符合用户请求的服务;采用质量度量匹配方法进行已语义标注的需求和服务的质量信息匹配,以在符合用户请求的、大量功能相似的服务中找到满足或优于用户请求的质量参数的服务,从而建立起“基本+功能+质量”的三阶段语义匹配模型,以提高需求和语义匹配的精度。最后,设计按需服务发现方法,以找到最符合需求的、高质量的服务。

本文第2节介绍基于领域本体的概念语义相似度度量与分析方法;第3节构建基于领域本体的按需服务“三阶段”语义匹配模型;第4节提出基于领域本体的按需服务发现方法,并以物流领域的具体实例进行了性能测试;最后总结并介绍下一步工作。

2 基于领域本体的概念语义相似度度量与分析

2.1 领域本体

领域本体是指通过对领域内基本的概念以及概念间关系的精确描述,使其成为人机之间、机器和机器之间互相理解的语义基础。它是这个领域概念化的、显式的规格说明,一般会用本体语言进行表示,能够满足特定领域范围内的语义Web应用。在语义Web范畴内,领域本体是关于该领域知识的概念化、形式化的规范,其作用是概念的描述、语义的揭示、一致性说明以及支持推理,从而完成对领域知识的有效组织、共享和复用。在网络化软件的按需服务中,领域本体将明确定义领域中实体概念、操作以及非功能需求属性及其之间的语义关联、约束规则和需要满足的公理,其主要作用是对具体领域的用户需求,以及与领域相关的、大量的服务资源进行语义描述和标识,从而使得领域内的需求和服务的相关术语及概念统一起来,实现知识共享,为网络化软件的需求和服务的匹配提供语义支持。其形式化定义如下。

定义1(领域本体) $DO = (Concept, Attribute, Instance, Relation, Axiom)$ 。

其中, DO 表示领域本体; $Concept$ 表示领域内的概念集合; $Attribute$ 表示领域内的概念属性的集合; $Instance$ 表示领域内的概念和属性的实例集合; $Relation$ 表示领域内的概念、属性以及实例间的关系集合; $Axiom$ 表示本体公理。网络化软件中,所构建的领域本体分成3种:实体本体(Entity Ontology)、操作本体(Operation Ontology)和情境本体(Context Ontology)。其中,实体本体描述的是一些名词性的概念,用来定义领域中所涉及的对象、对象的属性及其相互之间的关系;操作本体描述的是一些动词性的概念,用来定义领域中所涉及的操作及其相互之间的关系;情境本体描述的是领域中公共的情境信息,用来定义领域中所涉及的情境及其相互之

间的关系。以物流领域为例,建立的领域本体如图1和图2所示。其中,实体本体描述了领域内基本的实体名词概念,主要包括货物、货物状态、订单、参与者、交通路线等。其中,货物可分为一般货物和冷冻货物两类;订单进一步可分为装载订单、运送订单、分发订单、搬运订单和签名订单;参与者主要包括顾客与管理者,顾客有个人顾客和公司顾客两种类别,管理者可分为订购管理者、分发管理者和仓库管理者3类。操作本体描述了领域内基本的操作动词概念,包括订单的生成、增加、修改、删除、排序以及分派等操作。其中,订单的排序方式包括按缓急程度、交付日期以及目的地3种方式进行排序;交通的分派包括一车一订单、一车多订单、多车一订单以及网上分派4种方式。情境本体描述了领域内基本的情境概念,包括情境信息的类型(包括响应时间、费用、可用性、可靠性、吞吐量等)、情境信息的度量(包括情境信息的值、值的类型(包括数值型、布尔型、日期型、枚举型等)、情境关系符(包括大于、小于、等于等)、描述单位(包括秒钟、分钟、小时、千米、美元、人民币、分、百分数等)、情境信息的程度(包括高、中、低、好、一般、差等)。

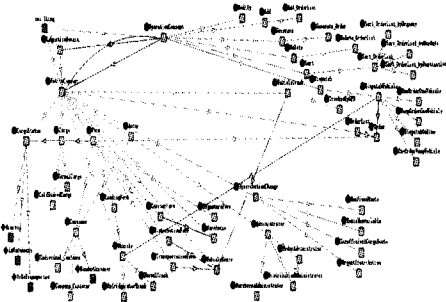


图1 物流领域的实体本体和操作本体

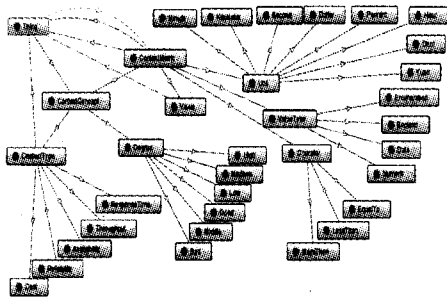


图2 物流领域的情境本体

2.2 概念语义相似度度量与分析

在基于领域本体的概念语义相似度计算中,概念之间的结构关系、概念所在的层次、同层中概念的数量、概念间语义关系的强度等都是语义相似度计算中很重要的影响因素。因此,综合考虑这些因素是很有必要的。以下把它们分别定义成关系因子、距离因子、层次因子、密度因子。

(1) 关系因子

概念之间的结构关系有很多,比如同义关系(Same-of)、反义关系(Opposite-of)、整体-部分关系(Part-of)、一般-特殊关系(Specific-of)等。同义关系是指两个概念的含义是一样的,比如“番茄”和“西红柿”;反义关系是指两个概念的含义是相反的,比如“左”和“右”;整体-部分关系是指概念之间的组成关系,比如“人”由“男人”和“女人”组成;一般-特殊关系是指子概念继承父概念的所有特征,同时可以适当扩展自身的特征,比如“书”和“计算机书”。结构关系不一样,则概念之

的语义相似度是不一样的。在本体图中,可以为有向边设定一个权值,突出结构关系的差异,使得语义距离的计算更加精确。为此,有向边的权值定义如下:

$$Weight(E) = \begin{cases} 0 \Leftrightarrow \text{Same-of} \\ 0.6 \Leftrightarrow \text{Specific-of} \\ 0.8 \Leftrightarrow \text{Part-of} \\ \infty \Leftrightarrow \text{Opposite-of} \end{cases} \quad (1)$$

例如,在如图1所示的物流领域实体本体中,“参与者(Actor)”由“顾客(Customer)”与“管理者(Administrator)”两者组成,其为整体-部分关系(Part-of),则 $Weight(Actor \rightarrow Customer) = Weight(Actor \rightarrow Administrator) = 0.8$ 。同样地, $Weight(OperationConcept \rightarrow Generate) = Weight(OperationConcept \rightarrow Add) = 0.8$ 。

(2) 距离因子

语义距离是指本体图中连接两个概念之间的最短路径的长度。概念之间的语义距离越大,其相似度越低。反之亦然。下式定义了概念之间的语义距离计算公式。

$$Dis(v1, v2) = \sum_{i=1}^{minlength(v1, v2)} weight(i) \quad (2)$$

式中, $Dis(v1, v2)$ 表示节点所表示的概念 $v1$ 与 $v2$ 之间的语义距离, $weight(i)$ 代表连接 $v1, v2$ 的最短路径 $minlength(v1, v2)$ 上的有向边的权值。例如, $Dis(Generate, Add) = weight(OperationConcept \rightarrow Generate) + weight(OperationConcept \rightarrow Add) = 1.6$ 。

(3) 层次因子

在考虑到概念节点间的语义距离的同时,还应该考虑到概念节点所在的层次。本体图中,下层概念节点是对其相应的概念上层节点的细化,它包含的含义更加具体,参与语义匹配的精度更高。

$$Level(v1, v2) = \frac{depth(v1) + depth(v2)}{2depth(G)} \quad (3)$$

式中, $depth(vi)$ 表示概念节点 $vi (i=1 \sim n)$ 的深度; $depth(G)$ 表示本体图 G 的深度。例如, $Level(Generate, Add) = \frac{depth(Generate) + depth(Add)}{2depth(G)} = \frac{1+1}{2 \times 3} = \frac{1}{3}$ 。

(4) 密度因子

本体图中不同概念节点的密度是不一样的。有些概念节点的兄弟节点很多,密度很大,说明细化程度较高,与图中其他概念节点之间的相似度就小;有些概念节点的兄弟节点很少,密度很小,说明细化程度较低,与图中其他概念节点之间的相似度就高。

$$Density(v1, v2) = \frac{T}{sibling(v1) + sibling(v2) + T} \quad (4)$$

式中, $sibling(vi)$ 表示概念节点 $vi (i=1 \sim n)$ 的兄弟节点的数量。 $T (T \geq 1)$ 为一个可以适当调节的参数。例如, $Density(Generate, Add) = \frac{T}{sibling(Generate) + sibling(Add) + T} = \frac{5}{5+5+5} = \frac{1}{3}$ (T 取值为5)。

综合考虑以上因素,提出基于本体图的概念语义相似度计算方法如下:

$$Sim(v1, v2) = \alpha \times \frac{\lambda}{Dis(v1, v2) + \lambda} + \beta \times Level(v1, v2) + \gamma \times Density(v1, v2) \quad (5)$$

3 基于领域本体的按需服务“三阶段”语义匹配模型

在 SOA 中, 服务提供者发布服务的相关描述信息至服务注册中心; 服务注册中心则管理和维护服务的相关信息, 并与服务提供者、消费者进行交互; 服务请求者则向服务注册中心提出具体的服务查找请求^[19]。由此, 网络化软件按需服务发现的问题可转化为已语义标注过的用户需求和描述之间的匹配问题。基于领域本体的按需服务匹配分成 3 个层次: 利用已有的关键字匹配方法进行已标注需求和已标注服务的基本信息匹配; 利用前述的基于本体图的概念语义相似度匹配方法进行已标注需求和已标注服务的功能信息匹配; 采用质量度量匹配方法进行已标注需求的情境信息和已标注服务的质量信息的匹配, 以在符合用户需求情境的、大量功能相似的服务中找到满足或优于用户需求的质量参数的服务。3 个阶段逐步递进, 通过判断是否大于预定阈值来定位和获取最符合用户需求的服务资源。

考虑到要进行需求和语义匹配, 在用户的需求和服务注册库中的服务经过语义标注之后, 本文将它们以一个语义三元组模型的形式来加以描述。

定义 2 语义三元组模型 $WS = \langle WS_{basic}, WS_{func}, WS_{QoS} \rangle$ 其中, WS_{basic} 表示语义标注的需求和服务的基本信息, 包括所属领域、类别等信息, $WS_{basic} = \langle \langle domain \rangle, \langle category \rangle \rangle$; WS_{func} 表示语义标注的需求和服务的功能信息, 包括名称、操作及其输入、输出等信息, $WS_{func} = \langle name, operation^* \langle \langle input^* \rangle, \langle output^* \rangle \rangle \rangle$, 其中操作可能包含多个, 而每个操作包含多个输入、输出参数; WS_{QoS} 表示语义标注的需求情境信息和服务质量信息, 包括情境和质量信息的类型、关系操作、取值、程度和单位描述等信息, $WS_{QoS} = \langle ContextType^*, Operation^*, Value^*/Degree^*, Unit^* \rangle$ 。

服务发现的关键问题是已语义标注过的用户需求和描述之间的匹配问题, 包括基本匹配、功能匹配和质量匹配 3 个层次, 本文给出的以下匹配算法就是针对这 3 个层次来展开的。令 WS_R 代表已标注过的需求的三元组信息, WS_A 代表服务注册库中已扩展和标注过的服务的三元组信息。

3.1 基于关键字的基本匹配

基本匹配包括需求和所属领域、类别等信息的匹配, 用 $BMD(WS_R, WS_{basic}, WS_A, WS_{basic})$ 表示。需求和所属领域是指它们所在的领域是否一致, 即与领域本体所描述的概念是否属于同一领域, 如物流、交通等领域, 用 $WS_{basic, domain}$ 表示; 类别是指具体领域下的需求和功能类别是否一致, 如物流领域下的仓储管理、货物配送等, 用 $WS_{basic, category}$ 表示。通过此次匹配, 可以过滤掉大量与用户请求无关的领域以及类别, 从而极大地缩小了搜索的范围, 提高了匹配的效率和。若用户需求 WS_R 与已发布的服务 WS_A 属同一领域中的同一类别, 那么匹配成功; 否则, 匹配失败。因此, 基本匹配的计算公式如下:

$$BMD(WS_R, WS_{basic}, WS_A, WS_{basic}) = \begin{cases} 1, & (WS_R, WS_{basic, domain} = WS_A, WS_{basic, domain}) \wedge \\ & (WS_R, WS_{basic, category} = WS_A, WS_{basic, category}) \\ 0, & (WS_R, WS_{basic, domain} \neq WS_A, WS_{basic, domain}) \vee \\ & (WS_R, WS_{basic, category} \neq WS_A, WS_{basic, category}) \end{cases} \quad (6)$$

3.2 基于本体概念语义相似度度量的功能匹配

服务的功能是指服务的功能描述信息, 主要包括服务的名称、操作及其输入、输出等信息。因此, 功能匹配主要包括

两个方面的匹配: 已语义标注的需求和服务的名称匹配和操作信息(输入和输出信息)匹配, 用 $FMD(WS_R, WS_{func}, WS_A, WS_{func})$ 表示(其中, 已语义标注的需求和服务的名称匹配用 $FNMD(WS_R, WS_{func, name}, WS_A, WS_{func, name})$ 表示, 操作信息匹配用 $FOMD(WS_R, WS_{func, operation}, WS_A, WS_{func, operation})$ 表示), 其计算公式如下:

$$FMD(WS_R, WS_{func}, WS_A, WS_{func}) = \begin{cases} \text{if } FNMD(WS_R, WS_{func, name}, WS_A, WS_{func, name}) \leq 1/3 \\ \text{Then } FOMD(WS_R, WS_{func, operation}, WS_A, WS_{func, operation}) \\ \text{else } FNMD(WS_R, WS_{func, name}, WS_A, WS_{func, name}) > 1/3 \\ \text{Then } \frac{1}{2} [FNMD(WS_R, WS_{func, name}, WS_A, WS_{func, name}) + \\ FOMD(WS_R, WS_{func, operation}, WS_A, WS_{func, operation})] \end{cases} \quad (7)$$

式中, $FNMD(WS_R, WS_{func, name}, WS_A, WS_{func, name})$ 用式(5)计算, 即采用基于本体图的概念语义相似度计算方法得到。假设用户需求的功能操作个数为 X , 用 $WS_R, WS_{func, operation}(i)$ 表示, $i=1 \sim X$; 已发布服务的功能操作个数为 Y , 用 $WS_A, WS_{func, operation}(j)$ 表示, $j=1 \sim Y$, 那么 $FOMD(WS_R, WS_{func, operation}, WS_A, WS_{func, operation})$ 的计算过程如下^[20]:

- A) 如果 $X=Y=0$, 则 $FOMD(WS_R, WS_{func, operation}, WS_A, WS_{func, operation})=0$;
- B) 如果 X, Y 中有一个为 0, 则 $FOMD(WS_R, WS_{func, operation}, WS_A, WS_{func, operation})=0$;
- C) 如果 X, Y 都不为 0, 则构造功能操作相似度矩阵

$$M = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1Y} \\ m_{21} & m_{22} & \cdots & m_{2Y} \\ \cdots & \cdots & \cdots & \cdots \\ m_{X1} & m_{X2} & \cdots & m_{XY} \end{bmatrix}$$

式中, m_{ij} 表示用户需求的功能操作 $WS_R, WS_{func, operation}(i)$ 和已发布服务的功能操作 $WS_A, WS_{func, operation}(j)$ 的相似度, 通过式(5)进行计算, 即采用基于本体图的概念语义相似度计算方法得到。

- D) 令 $K = \min(X, Y)$, $Q_i = 0, i=1 \sim K$ 。首先, 找出矩阵 M 中值最大的元素 m_{ij} , 令 $Q_i = m_{ij}$; 然后, 删除 m_{ij} 所在的行与列, 得到矩阵 M' 。重复执行这个步骤, 直到 M' 为空, 从而得到相似度序列 $Q_1, Q_2, \dots, Q_i, \dots, Q_K$ 。

- E) 针对这 K 对相匹配的功能操作, 用 $FOMD_i(WS_R, WS_{func, operation, input}, WS_A, WS_{func, operation, input})$ 表示第 i 对相匹配的功能操作的输入参数的匹配度, $i=1 \sim K$ 。令用户需求的每个功能操作的输入参数个数为 X' , 用 $WS_R, WS_{func, operation, input}(i')$ 表示, $i'=1 \sim X'$; 已发布服务的每个功能操作个数为 Y' 个, 用 $WS_A, WS_{func, operation, output}(j')$ 表示, $j'=1 \sim Y'$, 那么 $FOMD_i(WS_R, WS_{func, operation, input}, WS_A, WS_{func, operation, input})$ 类似于 $FOMD(WS_R, WS_{func, operation}, WS_A, WS_{func, operation})$ 的计算方法: 首先, 采用从 A) 至 D) 的步骤计算出输入参数相似度序列 $Q'_1, Q'_2, \dots, Q'_i, \dots, Q'_{K'}$, 其中 $K' = \min(X', Y')$; 然后, 计算 $FOMD_i(WS_R, WS_{func, operation, input}, WS_A, WS_{func, operation, input}) = \sum_{i=1}^{K'} a'_i \times Q'_i, \sum_{i=1}^{K'} a'_i = 1$ 。

- F) 同样地, 针对这 K 对相匹配的功能操作, 用 $FOMD_i(WS_R, WS_{func, operation, output}, WS_A, WS_{func, operation, output})$ 表示第 i 对相匹配的功能操作的输出参数的匹配度, $i=1 \sim K$, 其计算方法类似于 $FOMD_i(WS_R, WS_{func, operation, input}, WS_A, WS_{func, operation, input})$, 具体过程见第 E) 步。

G)最后,计算得到服务操作信息匹配 $FOMD(WS_R,$

$$WS_{func, operation}, WS_A, WS_{func, operation}) = \left[\sum_{i=1}^K Q_i \times 0.5 \times [FOMD_i(WS_R, WS_{func, operation, input}, WS_A, WS_{func, operation, input}) + FOMD_i(WS_R, WS_{func, operation, output}, WS_A, WS_{func, operation, output})] \right] / \sum_{i=1}^K Q_i。$$

3.3 基于 QoS 度量的质量匹配

在已语义标注的需求和服务的基本匹配和功能匹配完成之后,有可能存在多个已发布的服务满足用户需求的情况。也就是说,还需要从多个符合功能匹配要求的服务中找到较好的服务返回给请求者。由此,有必要进行已语义标注的需求和服务的质量匹配,以使匹配到的服务既能满足功能方面的需求,也能满足非功能方面的需求。用 $NFMD(WS_R, WS_{QoS}, WS_A, WS_{QoS})$ 表示,其计算公式如下:

$$NFMD(WS_R, WS_{QoS}, WS_A, WS_{QoS}) = \prod_{i=1}^n F(WS_R, WS_{QoS_i}, WS_A, WS_{QoS_i}) \times \sum_{j=1}^n (W_j \times a_j) \quad (8)$$

$$F(WS_R, WS_{QoS_i}, WS_A, WS_{QoS_i}) = \begin{cases} 1, & \begin{cases} (WS_R, WS_{QoS_i}, ContextType = WS_A, WS_{QoS_i}, ContextType) \wedge \\ (WS_R, WS_{QoS_i}, Operator = WS_A, WS_{QoS_i}, Operator) \wedge \\ (WS_R, WS_{QoS_i}, Value(Degree) = WS_A, WS_{QoS_i}, Value(Degree)) \wedge \\ (WS_R, WS_{QoS_i}, Unit = WS_A, WS_{QoS_i}, Unit) \end{cases} \\ 0, & \begin{cases} (WS_R, WS_{QoS_i}, ContextType \neq WS_A, WS_{QoS_i}, ContextType) \vee \\ (WS_R, WS_{QoS_i}, Operator \neq WS_A, WS_{QoS_i}, Operator) \vee \\ (WS_R, WS_{QoS_i}, Value(Degree) \neq WS_A, WS_{QoS_i}, Value(Degree)) \vee \\ (WS_R, WS_{QoS_i}, Unit \neq WS_A, WS_{QoS_i}, Unit) \end{cases} \end{cases} \quad (9)$$

式中, $\prod_{i=1}^n F(WS_R, WS_{QoS_i}, WS_A, WS_{QoS_i})$ 是用来找到满足用户情境需求约束条件下的服务; $\sum_{j=1}^n (W_j \times a_j)$ 是用来找到用户情境需求所偏好的、质量匹配度最高的服务。 WS_R, WS_{QoS_i} 和 WS_A, WS_{QoS_i} 分别表示用户需求的情境信息和已发布服务的质量信息,比如响应时间、吞吐量等, $i = 1 \sim n$; $F(WS_R, WS_{QoS_i}, WS_A, WS_{QoS_i})$ 表明只有当需求情境和服务质量信息的类型、关系操作符、值(或程度描述)和单位描述完全相同时,才取值为1。事先需要将服务的质量描述信息进行转化,使其尽量与需求的情境信息保持一致,以便于匹配,如用户希望服务的响应时间小于 3ms ($\langle responsetime, lessthan, 3, ms \rangle$),而某服务的响应时间等于 2ms ($\langle responsetime, equalto, 2, ms \rangle$),则将 $\langle responsetime, equalto, 2, ms \rangle$ 转化成 $\langle responsetime, lessthan, 3, ms \rangle$,其计算方法与式(6)一样。例如,费用 $WS_R, WS_{QoS_1} \leq 3Yuan, WS_A, WS_{QoS_1} = 2Yuan$,则 $F(WS_R, WS_{QoS_1}, WS_A, WS_{QoS_1}) = 1$; 可用性 $WS_R, WS_{QoS_2} \geq 60\%, WS_A, WS_{QoS_2} = 50\%$,则 $F(WS_R, WS_{QoS_2}, WS_A, WS_{QoS_2}) = 0$ 。其中, a_j 表示各种增量型和减量型的、无量纲化处理后的服务质量归一化值, $j = 1 \sim n, 0 \leq a_j \leq 1$,而 $\sum_{j=1}^n (W_j \times a_j)$ 则表示根据用户需求所偏好的情境属性权重 W_j 计算得到的服务质量信息的综合值, $\sum_{j=1}^n W_j = 1$ 。

4 基于领域本体的按需服务发现方法

4.1 方法概述

根据按需服务的“三阶段”语义匹配模型,基于领域本体的按需服务发现方法的整个过程如图3所示。

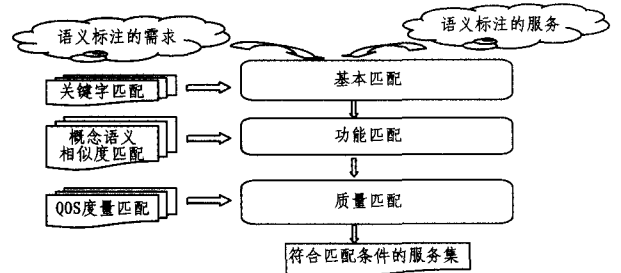


图3 基于领域本体的按需服务发现方法

(1)根据已标注过的语义用户需求三元组 WS_R 的基本信息,利用基本匹配方法 $BMD(WS_R, WS_{basic}, WS_A, WS_{basic})$,与已扩展和语义标注过的服务注册库中的服务三元组 WS_A 的基本信息进行匹配,以找到满足基本需求的服务集 Ω_1 。

(2)在服务集 Ω_1 的基础上,根据已语义标注过的用户需求三元组 WS_R 的功能信息,利用功能匹配方法 $FMD(WS_R, WS_{func}, WS_A, WS_{func})$,找到符合功能需求的、功能匹配度在一定阈值 θ_1 以上的服务集 Ω_2 。

(3)在服务集 Ω_2 的基础上,根据已语义标注过的用户需求三元组 WS_R 的质量信息,利用质量匹配方法 $NFMD(WS_R, WS_{QoS}, WS_A, WS_{QoS})$,找到满足质量约束条件的、用户需求所偏好的、质量匹配度最高的服务。

4.2 性能测试

面向物流领域,开发了 Web 服务语义互操作注册与管理平台,在该平台上基本实现了网络化软件的按需服务模型与发现模块(见图4、图5)。其基本实现流程为:(1)基于 Web 服务的 WSDL 地址,根据其所属领域(物流)进行服务基本信息和功能信息的注册和解析,并利用实体本体和操作本体对它们进行语义标识;(2)针对语义标注过的 Web 服务进行质量属性的扩充,并利用情境本体进行标注;(3)按需服务语义查询,通过输入服务请求信息,包括服务领域、输入、输出、质量等信息,采用按需服务“三阶段”语义匹配模型进行匹配,找到符合用户要求的、质量最优的服务。

将本文方法与基于关键字的服务发现方法在查准率和查全率两个性能指标上进行了比较分析,其结果(实验过程略)证实了本文方法的有效性,如图5所示。

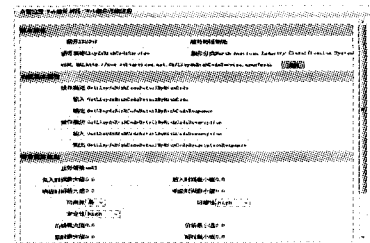


图4 注册和语义标识的 Web 服务详细信息

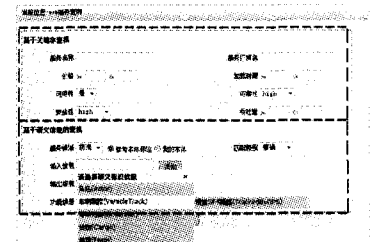


图5 关键字查询与按需服务语义查询比较

一种基于概率模型检测的 Web 服务组合验证方法,即采用扩展的有限自动机模型表示服务组合,并将其转换为 Markov 模型。然后采用概率模型检测器 PRISM 验证服务组合;最后通过实例进一步说明了方法的可行性。本文所采用的 Markov 模型不能表示时间信息。下一步工作将考虑扩展分析模型,使之能表示服务组合中的时间信息,并进行实验验证。

参考文献

- [1] 范贵生,虞慧群,陈丽琼,等. 基于 Petri 网的服务组合故障诊断与处理[J]. 软件学报,2010,21(2):231-243
- [2] Pleisch S, Schiper A. Approaches to fault-tolerant and transactional mobile agent execution—An algorithmic view[J]. ACM Computing Surveys,2004,36(3):219-262
- [3] 林惠民,张文辉. 模型检测:理论、方法与应用[J]. 电子学报,2002,30(12A):1907-1912
- [4] Kwiatkowska M, Norman G, Parker D. Advances and Challenges

(上接第 100 页)

测试结果表明,基于关键字的服务发现方法的查准率和查全率都较低;本文方法的查全率达到 80% 左右,查准率达到 90% 以上。因此,利用本文方法,能够提高需求和服务之间的语义匹配精度,缩小搜索范围,能够找到最符合用户需求的、高质量的服务资源来构造网络化软件。

结束语 本文提出一种网络化软件的按需服务发现方法。首先构建了需求和服务的 3 类领域本体,使得领域内的需求和服务的相关概念统一起来,实现了知识共享,为网络化软件的需求和服务的匹配提供了语义支持;接着重点讲述了基于本体图的概念语义相似度度量与分析,并以此为基础,建立了按需服务“基本+功能+质量”三阶段语义匹配模型;最后阐述了网络化软件的按需服务发现方法,并以物流领域的具体实例进行了性能测试。下一步工作将集中于探讨建立结构更加合理、知识更加丰富的领域本体,同时考虑如何将本文所提方法和模型应用于实际环境中。

参考文献

- [1] He Ke-qing, Wang Jian, Liang Peng. Semantic Interoperability Aggregation in Service Requirements Refinement [J]. Journal of Computer Science and Technology,2010,25(6):1103-1117
- [2] 何克清,彭蓉,李兵,等. 网络化软件[M]. 北京:科学出版社,2008
- [3] 吴朝晖,邓水光,吴健. 服务计算与技术[M]. 杭州:浙江大学出版社,2009
- [4] Paolucci M, Kawamura T, et al. Semantic Matching of Web Services Capabilities[C]//Proceedings of the 1st International Semantic Web Conference, 2002:333-347
- [5] Maximilien E M, Singh M P. A framework and ontology for dynamic Web services Selection [J]. Internet Computing, 2004, 8(5):84-93
- [6] Soydan B A, Singh Munindar P. A DAML-based repository for QoS-aware semantic Web service selection[C]//Proceedings of the IEEE International Conference on Web Services, California, USA,2004:368-375
- [7] El Haddad J, Manouvrier M, Rukoz M. TQoS: Transactional and

of Probabilistic Model Checking[C]//Communication Control and Computing. Allerton, IEEE press, 2010:1691-1698

- [5] 雷丽晖,段振华. 一种基于扩展有限自动机验证组合 Web 服务的方法[J]. 软件学报,2007,18(12):2980-2990
- [6] 何亚丽,戎玫,张广泉. 一种基于 UPPAAL 的 Web 服务组合模型检测方法[J]. 计算机科学,2010,37(11):122-125
- [7] 陈丁剑,吴健,马满福,等. 基于 Petri 网的 Web 服务组合建模[J]. 计算机科学,2006,33(5):128-130
- [8] 辜希武,卢正鼎. 基于 Pi-演算的 BP4LWS Web 服务组合形式化模型[J]. 计算机科学,2007,34(3):69-74
- [9] Kwiatkowska M, Norman G, Parker D. PRISM 2. 0: a tool for probabilistic model checking[C]//International Conference on the Quantitative Evaluation of Systems (QEST), Enschede, The Netherlands, IEEE Computer Society, 2004:27-30
- [10] Hansson H, Jonsson B. A logic for reasoning about time and reliability [J]. Formal Aspects of Computing, 1994, 6(5):512-535

QoS-aware Selection Algorithm for Automatic Web Service Composition [J]. IEEE Transactions on Services Computing, 2010,3(1):73-85

- [8] 吴健,吴朝晖. 基于本体论和词汇语义相似度的 Web 服务发现[J]. 计算机学报,2005,28(4):595-602
- [9] 杨放春,苏森,李祯. 混合 QoS 模型感知的语义 Web 服务组合策略[J]. 中国科学(E 辑:信息科学),2008,38(10):1697-1716
- [10] 孙萍,蒋昌俊. 利用服务聚类优化面向过程模型的语义 Web 服务发现[J]. 计算机学报,2008,31(8):1340-1353
- [11] Web Service Challenge [EB/OL]. <http://ws-challenge.georgetown.edu/wsc09/>
- [12] Jiang W, Zhang C, Huang Z, et al. Qsynth: A tool for QoS-aware automatic service composition[C]//Proceedings of the IEEE International Conference on Web Services, Miami, FL, USA, 2010
- [13] Bartalos P, Bielikovaa M. Semantic Web service composition framework based on parallel processing[C]//Proceedings of the 11th IEEE Conference on Commerce and Enterprise Computing, Vienna, Austria, 2009
- [14] 孙海霞,钱庆,成颖. 基于本体的语义相似度计算方法研究综述[J]. 现代图书情报技术,2010(1):51-56
- [15] Antoniou G, Van Harmelen F. A Semantic Web Primer [M]. The Mit Press, 2004:200-208
- [16] Lord P W, Stevens R D, Brass A, et al. Investigating Semantic Similarity Measures Across the Gene Ontology: The Relationship Between Sequence and Annotation [J]. Bioinformatics, 2003,19(10):1275-1283
- [17] 潘谦红,王炬,史忠植. 基于属性论的文本相似度计算[J]. 计算机学报,1999,22(6):651-655
- [18] Knappe R, Bulskov H, Andreassen T. On Similarity Measures for Content-based Querying[C]//Proceedings of the 10th International Fuzzy Systems Association World Congress, 2003:400-403
- [19] 喻坚,韩燕波. 面向服务的计算—原理和应用[M]. 北京:清华大学出版社,2006
- [20] 员红娟,叶飞跃,李霞,等. 基于语义的 Web 服务发现核心技术研究[J]. 计算机应用,2006,26(11):2661-2666