

基于本体和 QoS 的面向敏捷 供应链知识服务匹配研究 *

郝 玫, 王道平, 冯小东

(北京科技大学 经济管理学院, 北京 100083)

摘要: 为了加强面向敏捷供应链的知识服务的服务描述信息的机器可理解性,提高用户需求和服务能力之间功能匹配的准确性,在总结和分析相关服务发现和匹配算法的基础上,提出了一个基于扩展 OWL-S 和服务质量 QoS 的知识服务描述模型。提出一种基于本体计算语义相似度的方法,利用此方法实现服务的功能性匹配。最后用一个服务匹配实例说明了该匹配算法在充分考虑用户需求方面的有效性。

关键词: 敏捷供应链; 服务描述; 服务匹配; 服务质量; 语义相似度

中图分类号: TP311 **文献标志码:** A **文章编号:** 1001-3695(2010)07-2556-04

doi: 10.3969/j.issn.1001-3695.2010.07.043

Research on knowledge services matching agile supply chain-oriented based on ontology and QoS

HAO Mei, WANG Dao-ping, FENG Xiao-dong

(School of Economics & Management, University of Science & Technology Beijing, Beijing 100083, China)

Abstract: In order to strengthen the machine-understanding of description of services in agile supply chain-oriented knowledge services and increase the accuracy of matching between user needs and services capabilities, on the basis of summarizing and analyzing the relevant service discovery and matching algorithms, this paper proposed a knowledge services description model based on the expansion of the OWL-S and QoS. Besides, it also proposed an algorithm of calculating semantic similarity based on ontology, made using of which the functional matching was achieved. At last, it used a service matching example to illustrate the effectiveness of this matching algorithm in taking full account of user needs.

Key words: agile supply chain; services description; services matching; services quality; semantic similarity

全球竞争环境的不断变化给企业内部和外部活动带来了极大的影响。新环境需要企业有较好的响应外部环境的敏捷性。随着供应链各企业对敏捷性要求的不断增长,知识服务的理念开始进入供应链领域,并将在供应链管理的发展中起到越来越重要的作用。企业对知识的获取、创造、共享、传播以及应用的能力成为决定企业竞争能力的关键要素,知识继物流、信息流与资金流之后成为供应链管理的又一个重要内容。因此知识服务是实施敏捷供应链的企业必须考虑的。企业通过利用联系供应链上下游企业的知识服务系统,快速响应市场变化,增强敏捷供应链的敏捷性,进而提高企业的竞争力。知识服务系统主要是基于传统信息资源的 Web 服务,随着 Web 服务的大量涌现,从众多的服务中发现与用户需求相匹配的 Web 服务成为 Web 服务系统中的一个关键问题。因此需要有一种面向敏捷供应链的 Web 服务匹配策略,加强服务描述信息特别是服务质量信息的机器可理解性,充分考虑用户的需求,支持基于服务质量优劣的服务筛选^[1]。

1 基于 QoS (服务质量) 的知识服务描述模型

知识服务主要是以 Web 服务为实现形式,Web 服务发现

的过程是一个根据服务请求者提出的要求,在已经注册的服务集中寻找匹配用户需求服务的过程。一个完整的 Web 服务发现流程的主要步骤如下:

a) 提供者服务描述:服务提供者用描述语言来描述其提供的服务。

b) 服务发布:服务代理者在注册中心分类并发布服务提供者提供的服务描述。

c) 请求者需求服务描述:服务请求者询问代理者是否存在具有合适功能的服务,对请求服务的功能进行描述。

d) 服务匹配:服务代理者将请求者的需求与已有的服务描述进行匹配并返回结果。所谓服务匹配就是能够在现存的 Web 服务中找到与请求者的需求匹配度高的 Web 服务。

e) 调用服务:服务请求者根据发现的服务描述中绑定的服务地址激活调用目标 Web 服务。

服务描述是服务请求者和提供者之间关于服务的一致表示。OWL-S 提供了一种传统的 Web 服务描述模型,其中的 ServiceProfile 文件提供了服务的基本描述信息和有关的性能属性^[2]。

定义 1 和 2 给出了传统的 Web 服务描述模型。

收稿日期: 2009-12-23; **修回日期:** 2010-01-28 **基金项目:** 国家自然科学基金资助项目(70872010)

作者简介: 郝玫(1979-),女,博士研究生,主要研究方向为知识管理、供应链管理(haomei@manage.ustb.edu.cn);王道平(1964-),男,教授,博士生导师,主要研究方向为管理决策分析、物流与价值链管理、智能管理系统;冯小东(1988-),男,硕士研究生,主要研究方向为供应链。

定义1 本体

本体主要描述某特定领域的相关概念之间的关系,为 Web 服务提供语义基础:

本体 $DO = \langle C, R \rangle$ 。其中: C 为概念集合, $R \subseteq C \times C$ 表示概念之间关系的集合。基本的关系如等价、继承、包含等。

在本文涉及的的本体库中,主要包括领域本体和服务类别本体两种。领域本体为某一特定领域内的概念及其之间的关系。服务类别本体采用树状结构,对服务进行分类,表示服务的类别。

定义2 基于 OWL-S 的传统 Web 服务描述模型

Web 服务描述为: $WS = \langle Id, DO, ServiceProvider, InPuts, OutPuts, ServiceCategory \rangle$ 。其中:Id 为服务的标志符;DO 为定义1中的领域本体;ServiceProvider 描述该 Web 服务提供者的有关信息;Inputs 为 Web 服务输入参数集合;Outputs 为 Web 服务输出参数集合;ServiceCategory 表示服务所属的类别^[3]。

虽然 OWL-S 引进本体描述 Web 服务功能和行为语义,能够对 Web 服务的语义描述提供很好的支持,但是其在 QoS 方面描述得不尽人意,并没有提供度量 QoS 的类和属性的详细定义。

对此,本文针对其不足提出了一种考虑 QoS 的 Web 服务描述模型,从而使得 Web 服务的发现匹配能够根据服务的 QoS 信息进行服务筛选。

定义3 Web 服务质量模型

Web 服务质量模型 $QoS = \langle time, cost, reliability, stability, reputation, \dots \rangle$, 是一个可扩充向量。其中:时间(time)是一个度量性能的通用指标,包括延迟时间、处理时间;费用(cost)是描述服务请求者需要为使用服务而付出的费用;可靠性(reliability)为成功执行次数与调用执行总次数的比率;稳定性(stability)用于描述同一 Web 服务在不同的调用时刻所需的响应时间的差异,可以用以前所有服务所花费时间的统计标准差表示;信誉度(reputation)用于描述服务使用者对他们所调用的 Web 服务的评价,可以用所有服务调用者评价的平均值表示^[4]。

定义4 基于 QoS 的 Web 服务描述模型

基于 QoS 的知识服务描述模型 $QWS = \langle WS, QoS \rangle$ 。其中:WS 为定义2中的 Web 服务描述模型, QoS 为定义3中的 Web 服务质量模型。该模型不仅引进本体描述 Web 服务功能和行为语义,能够对 Web 服务的语义描述提供很好的支持,并且对服务质量方面的描述也给予考虑。

2 基于 QoS 的知识服务匹配模型

2.1 知识服务匹配策略

Web 服务匹配就是根据本体库中各概念之间的关系,对于服务注册中心的每一个注册的广告服务利用其描述文件与服务请求者的请求服务的描述文件通过匹配算法进行匹配。若一个服务和用户需求匹配成功就将其添加到查询结果的列表中,并将它与其他匹配成功的服务根据匹配度进行排序(图1)。

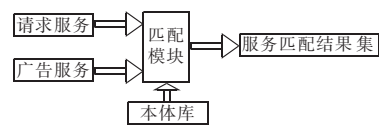


图1 Web 服务匹配策略

本文在总结相关文献基础上提出知识服务匹配(即图1中服务匹配模块)应包含以下过程:

a) 服务功能类别的匹配。为了提高匹配的效率和,首先根据请求服务(request)的服务类别(service category),利用本体库中的服务类别本体对所有广告服务(advertise)进行匹配计算,得到服务类别匹配度高的服务集合,这样可以减少大量不必要的服务匹配。本文在后面将具体介绍匹配算法。

b) 服务功能的匹配。其主要是对请求服务与广告服务的输入、输出参数集的匹配。

c) 服务质量的匹配。通过综合 QoS 各属性的值,进行 Web 服务质量评估,对满足功能匹配的服务进行筛选,去除服务质量差的服务,这部分将在以后研究。

d) 服务匹配度排序和选择。根据以上步骤的匹配结果,得到功能匹配度和服务质量匹配度不同的一个服务集合,然后在候选服务集合中根据服务功能匹配度和服务质量匹配度进行服务匹配等级的排序,将排好序的服务集合返回给用户。

2.2 基于语义相似度的匹配算法

在服务匹配模块中,服务功能类别和服务功能的匹配是基于语义相似度的匹配过程,基于语义相似度的服务匹配通过计算两个本体概念之间的语义相似度(一个介于0和1之间的数值)来得到更精确的匹配结果,克服了 Massimo Paolucci 等人提出的基于 DAML-S 的服务匹配方法在同一等级内部的不同情况不能加以区分,以及只考虑了本体概念之间的继承关系这两方面的不足。

本文在参考相关算法之上提出了一种计算本体中概念语义相似度的方法,下面介绍本体中概念语义相似度的计算步骤。

1) 语义距离计算

概念之间的语义距离为概念之间在本体加权有向图中的最小距离,此最小距离为概念之间最小路径的权值和。

定义5 公共祖先集(common ancestors, CA)是两个本体概念节点所有的祖先中共同的祖先集合。

定义6 最低共同祖先(lowest common ancestor, LCA)是指两个本体概念节点的公共祖先集中深度最深的祖先节点。

定义7 概念间语义距离(semantic distance, SD)是两个本体概念节点与最低共同祖先距离之和。概念间语义距离计算公式如式(1)所示:

$$SD(C_1, C_2) = (\text{dis}(C_1, LCA) + \text{dis}(LCA, C_2)) \quad (1)$$

其中:LCA 表示 C_1 、 C_2 的最低共同祖先节点, $\text{dis}(A, B)$ 表示概念 A 和 B 之间的距离。

$\text{dis}(A, B)$ 的计算方法如式(2)所示:

$$\text{dis}(A, B) = \sum_{C_i \in \text{path}(A, B)} \text{type}(C_i) \quad (2)$$

其中: $\text{path}(A, B)$ 是概念节点 A 和 B 之间最短路径中所有节点的集合; $\text{type}(C_i)$ 表示概念节点 C_i 和直接父节点之间在对应的本体中关系的类型对应的权值,其中等价关系为 0, 继承关

系为 1,包含关系为 2,自定义关系为 $n(n \geq 2)$ 。

2) 概念深度计算

定义 8 概念间语义深度 (semantic depth, Sdep) 是概念节点的最低共同祖先的深度。

式(3)表示语义深度计算公式:

$$Sdep(A, B) = (\text{depth}(LCA)) \quad (3)$$

其中: $\text{depth}(LCA)$ 表示 A, B 最低共同祖先的深度,并规定本体树根节点的深度为 1。

3) 概念间的相似度计算

语义相似度的计算方法是通过调用相似度函数将语义距离或语义深度转换成语义相似度,即此函数的输入为语义距离或语义深度,输出为语义相似度。

基于语义距离的语义相似度函数为

$$f_{dis}(SD) = \frac{a}{SD + a} \quad (4)$$

其中: SD 是式(1)所描述的两个概念间的语义距离; a 是控制语义函数灵敏度的参数, a 取值为 $a \geq 1$, a 取值越小,函数对 SD 的变化越敏感。

基于语义深度的语义相似度计算公式如式(5)所示:

$$f_{dep}(Sdep) = \frac{a}{a + \frac{b}{Sdep}} \quad (5)$$

其中: $Sdep$ 是两个概念间的概念深度; a, b 是控制函数灵敏度的参数。

利用上述两种公式可以得出概念间语义相似度计算公式。

定义 9 概念间语义相似度 (semantic similarity) 计算公式如下:

$$S_sim(C_1, C_2) = f_{dis}(SD) \times f_{dep}(Sdep) \quad (6)$$

4) 概念集之间的相似度计算

对于概念集合 A 和 $B: A = (A_1, A_2, A_3, \dots, A_n); B = (B_1, B_2, B_3, \dots, B_m)$ 。概念集合 A 到 B 的相似度为

$$\text{sim}(A, B) = \sum_{i=1}^n w_i \times \max_{j \in [1, m]} (S_sim(A_i, B_j)) \quad (7)$$

其中: w_i 为概念 A_i 的权重, w_i 为 $(0, 1)$ 之间的实数,之和为 1。

用同样的算法,可以得到 $\text{sim}(B, A)$ 。

概念集合 A, B 的语义相似度 (concept set similarity, CS_sim):

$$CS_sim(A, B) = \frac{1}{2} (\text{sim}(A, B) + \text{sim}(B, A)) \quad (8)$$

由此可见,用这种方式计算的概念集合之间的语义相似度的准确性很大程度上依赖于前面介绍的概念间的语义相似度。

2.3 基于本体概念之间语义相似度 Web 服务功能性匹配算法

利用上文提出的本体概念之间语义相似度计算公式,本文提出基于本体概念之间语义相似度的 Web 服务功能性匹配模型(图 2)。

下面具体介绍匹配算法:

a) 服务功能类别匹配

匹配时从请求服务的描述文件中提取出功能类别 (Req. ServiceCategory), 与服务注册中心的每一个广告服务描述文件中的功能类别 (Adv. ServiceCategory) 进行匹配度计算,即根据

式(6)计算它们的语义相似度 $S_sim(\text{Req. ServiceCategory}, \text{Adv. ServiceCategory})$ 。如果此值大于预先设定的阈值,表示此广告服务匹配成功,通过筛选,则可加入到返回服务集列表中进行下一步的匹配。

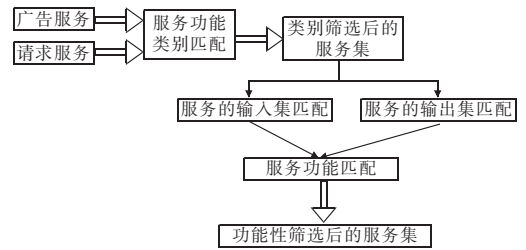


图 2 Web 服务功能性匹配模型

b) 服务功能匹配

服务功能匹配主要是对服务输入、输出参数集合进行匹配,对于满足服务功能类别匹配服务列表中的每一个广告服务和请求服务进行服务语义匹配度的计算,如果语义匹配度大于预先设定的阈值,则加入到返回服务集列表中。具体计算过程如下:

(a) 请求服务输出概念集和广告服务输出概念集匹配度计算

从请求服务和广告服务的描述文件中提取输出概念集:

请求服务输出概念集: $\text{ReqOutputs}[n] = \{\text{out}R_1, \text{out}R_2, \dots, \text{out}R_n\}$

广告服务输出概念集: $\text{AdvOutputs}[m] = \{\text{out}A_1, \text{out}A_2, \dots, \text{out}A_m\}$

式(9)给出了请求服务输出概念集和广告服务输出概念集匹配度计算方法:

$$\text{Match}(\text{ReqOutputs}, \text{AdvOutputs}) = \begin{cases} 0, & \text{AdvOutputs} = \emptyset, \text{ReqOutputs} \neq \emptyset \\ CS_sim(\text{ReqOutputs}, \text{AdvOutputs}), & \text{其他} \\ 1, & \text{AdvOutputs} \neq \emptyset, \text{ReqOutputs} = \emptyset \end{cases} \quad (9)$$

其中: $\text{AdvOutputs} = \emptyset, \text{ReqOutputs} \neq \emptyset$ 表示请求服务期望得到输出,服务广告不提供输出; $\text{AdvOutputs} \neq \emptyset, \text{ReqOutputs} = \emptyset$ 表示请求服务不要求输出。

(b) 请求服务求输入概念集和广告服务输入概念集匹配度计算

从请求服务和广告服务的描述文件中抽取输入概念集:

请求服务输入概念集: $\text{ReqInputs}[n] = \{\text{in}R_1, \text{in}R_2, \dots, \text{in}R_n\}$

广告服务输入概念集: $\text{AdvInputs}[m] = \{\text{in}A_1, \text{in}A_2, \dots, \text{in}A_m\}$

式(10)给出了请求服务输入概念集和广告服务输入概念集匹配度计算方法:

$$\text{Match}(\text{ReqInputs}, \text{AdvInputs}) = \begin{cases} 0, & \text{AdvInputs} \neq \emptyset, \text{ReqInputs} = \emptyset \\ CS_sim(\text{ReqInputs}, \text{AdvInputs}), & \text{其他} \\ 1, & \text{AdvInputs} = \emptyset, \text{ReqInputs} \neq \emptyset \end{cases} \quad (10)$$

其中: $\text{AdvInputs} \neq \emptyset, \text{ReqInputs} = \emptyset$ 表示请求服务没有输入,广告服务提供输入; $\text{AdvInputs} = \emptyset, \text{ReqInputs} \neq \emptyset$ 表示广告服务输入为空。

(c) 服务功能的语义匹配度计算

式(11)给出了请求服务和广告服务之间的语义匹配度计算方法:

$$\text{SemMatch} = w_1 \times \text{Match}(\text{ReqOutputs}, \text{AdvOutputs}) + w_2 \times \text{Match}(\text{ReqInputs}, \text{AdvInputs}) \quad (11)$$

其中: $w_1 + w_2 = 1$, 分别表示输出集和输入集在 Web 服务功能匹配中的权重。

3 应用实例

以供应链中产品销售领域本体作为知识服务的领域本体, 存在以下五个知识服务:

Req: ServiceCategory = (需求预测)
 Inputs(0.4) = {价格(0.3), 市场占有率(0.3), 前期销售量(0.2), 经济增长率(0.1), 季节指数(0.1)};
 Outputs(0.6) = {需求量(0.4), 需求波动范围(0.2), 最优库存量(0.4)};
 (括号内数字为用户输入的相应权重值)
 Adv1: ServiceCategory = (需求预测)
 Inputs = {价格, 竞争对手平均价格, 前期销售量, CPI 指数};
 Outputs = {需求量, 需求波动范围, 最优价格};
 Adv2: ServiceCategory = (订单预测)
 Inputs = {价格水平, 前期销售量, 季节指数};
 Outputs = {订单期望数量, 最大订单数量, 最低订单数量};
 Adv3: ServiceCategory = (库存预测)
 Inputs = {上期产量, 需求量, 当期库存};
 Outputs = {库存水平, 安全库存};
 Adv4: ServiceCategory = (订单预测)
 Inputs = {价格, 市场份额, 前期销售量, 季节指数};
 Outputs = {订单量, 最优库存量};

其中: Adv1~4 为已注册的广告服务, 即可提供的服务; Req 为请求服务; ServiceCategory 为其服务类别描述; Inputs、Outputs 为其输入输出概念集, 这些均是从服务描述中提取的与功能匹配有关的信息, 其他与此无关的信息并没提取。功能性匹配过程如下:

a) 计算服务功能类别匹配度。结果显示, $S_{\text{sim}}(\text{Req}, \text{ServiceCategory}, \text{Adv3}, \text{ServiceCategory})$ 小于阈值, 类别筛选后的服务集为 (Adv1, Adv2, Adv4)。

b) 计算服务功能语义匹配度。结果显示, $\text{SemMatch}(\text{Req}, \text{Adv2})$ 小于阈值, 功能性筛选后的服务集为 (Adv1, Adv4)。此服务集用于以后的基于服务质量的筛选。

从这个例子可以看出, 输入、输出集权重完全是用户请求

服务的描述信息, 对于其他描述信息相同而输入、输出集权重不同的服务, 此匹配算法得到的是完全不同的结果。例如将 Req 中输入、输出集权重改为 0.6、0.4 后, 功能性筛选后的服务集改变为 (Adv1, Adv2, Adv4)。因此该算法充分利用了请求服务的描述信息, 即用户的需求信息。

4 结束语

本文在总结和分析相关服务发现和匹配算法的基础上, 提出了一个面向敏捷供应链的基于扩展 OWL-S 和 QoS 知识服务描述模型, 此模型对知识服务的服务质量信息进行了描述, 并在此基础上总结和分析了传统匹配算法及其不足, 提出了一种语义相似度计算方法, 并将此方法引入到服务功能类别匹配和输入输出匹配中。在此算法中, 计算概念集相似度时, 利用用户设定的各个概念的权重计算加权平均值与简单的算术平均值方法相比, 加强了服务描述信息的机器可理解性, 即更大程度上匹配了用户的需求。

此服务匹配不仅实现了用户需求和服务能力之间的功能性匹配, 而且支持基于服务质量匹配的服务筛选, 具体的基于服务质量的筛选将是以后的重点研究内容。

参考文献:

- [1] 杨敏才, 凌超, 王槐林. 供应链的知识管理系统[J]. 研究与发展管理, 2004, 16(1): 44-48.
- [2] 臧志, 金葆弘, 李玉明. 服务描述和服务匹配研究[J]. 计算机科学, 2008, 35(10): 266-287.
- [3] 胡建强, 邹峰, 王怀民, 等. Web 服务描述语言 QWSDL 和服务匹配模型研究[J]. 计算机学报, 2005, 28(4): 505-513.
- [4] 韩涛, 郭荷清, 高英. 基于 QoS 的 Web 服务评估算法[J]. 计算机科学, 2006, 33(5): 60-62.
- [5] 吴健, 吴朝晖, 李莹. 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.
- [6] 吴海鹏, 饶若楠. 一种基于服务本体及其词汇语义的 Web 服务匹配算法[J]. 计算机应用与软件, 2008, 25(5): 151-153.
- [7] MARTA S, PAN J. Towards semantically enhanced Web service repositories[J]. Journal of Web Semantics, 2007, 5(2): 142-150.
- [8] 谢琪, 张振兴. 基于 Apriori 算法和 OLAP 的关联规则挖掘模型设计[J]. 计算机应用, 2007, 27(6): 4-9.
- [9] 吴军. 铝电解过程控制智能算法的研究[J]. 工业控制计算机, 2004(5): 31-32.
- [10] BOGDANOVA G, GEORGIEVA T. Discovering the association rules in OLAP data cube with daily downloads of folklore materials[C]// Proc of International Conference on Computer Systems and Technologies. 2005: 1-6.
- [11] 文蓉. 数据挖掘中关联规则算法及其应用研究[D]. 长沙: 湖南大学, 2007.
- [12] 刘杰, 黄亚楼. 基于 BP 神经网络的非线性网络流量预测[J]. 计算机应用, 2007, 27(7): 1770-1772.
- [13] LI Jie-jia, WU Cheng-dong, LI Meng-xin. The application of distributed neural network fault diagnosis technique in process control[C]// Proc of the 6th World Congress on Intelligent Control and Automation. 2006: 5553-5556.
- [14] LEE S C, HEINBUCH D V. Training a neural-net-work base intrusion detector to recognize novel attacks[J]. IEEE Trans on SMC, 2001, 31(4): 294-299.
- [15] 韩力群. 人工神经网络理论、设计及应用[M]. 北京: 化学工业出版社, 2005.

(上接第 2539 页)