第 34 卷 第 15 期 计 算 机 工 程 2008 年 8 月
Vol.34 No.15 Computer Engineering August 2008
·软件技术与数据库· 文章编号：1000—3428(2008)15—0051—03 文献标识码：A 中图分类号：TP311

基于本体概念相似度的语义 Web 服务匹配算法

彭 晖[1,2,3]，史忠植[1]，邱莉榕[1,3]，常 亮[1,3]
(1. 中国科学院计算技术研究所智能信息处理重点实验室，北京 100080；2. 湖南科技大学计算机科学与工程学院，湘潭 411201；
3. 中国科学院研究生院，北京 100039)

摘 要：通过定义本体中概念之间的语义距离来计算本体概念之间的相似度，提出一种基于该相似度的 Web 服务的精确匹配算法，新的算法与经典的 OWL-S/UDDI 匹配算法比较，不仅在等级上保持一致，而且使同一等级或不同等级之间的服务匹配都达到精确的程度。用 GEIS 系统中 Web 服务的数据进行两种算法的性能测试，得出相似度匹配算法的平均查准率是 OWL-S/UDDI 匹配算法的 1.8 倍，平均查准率是 OWL-S/UDDI 匹配算法的 1.4 倍。

关键词：语义 Web 服务；服务匹配；语义距离；本体概念相似度

Matching Algorithm of Semantic Web Service Based on Similarity of Ontology Concepts

PENG Hui[1,2,3], SHI Zhong-zhi[1], QIU Li-rong[1,3], CHANG Liang[1,3]
(1. Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080;
2. School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan 411201;
3. Graduate University of Chinese Academy of Sciences, Beijing 100039)

【Abstract】 A semantic distance between ontology concepts is defined to calculate the similarity between ontology concepts. Based on the similarity, a new semantic Web service matching algorithm is proposed. This algorithm abides by the grade of the classic OWL-S/UDDI algorithm and gives the precise matching between Web services. Experimental results show that the average precision of the new algorithm is 1.8 times than the OWL-S/UDDI algorithm and the average recall of the new algorithm is 1.4 times than the OWL-S/UDDI algorithm.

【Key words】 semantic Web service; service matching; semantic distance; similarity of ontology concepts

1 概述

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 服务系统中一个关键问题。现有的 Web 服务描述文件 WSDL[1]主要描述了 Web 服务的调用操作方式，而缺少对 Web 服务功能的描述；服务注册机制 UDDI[2]通过对服务注册信息(如服务名称，分类，公司名称等)进行关键词的精确匹配来发现服务，这种语法级的服务匹配在服务的查全率和查准率方面都无法达到令人满意的效果。如何在现有服务描述中加入服务的功能描述，即语义信息，通过服务语义的匹配来准确地查找服务成为关注的焦点。

在 W3C 组织提出语义 Web 服务描述语言 OWL-S[3]之后，卡内基梅隆大学的 Massimo Paolucci 等人提出了语义 Web 服务的 OWL-S/UDDI 匹配算法[4]，该算法通过对本体中概念的包含关系的推理将 Web 服务匹配分为 4 个不同的等级。它成为语义 Web 服务匹配的一个经典算法，经常被其他 Web 服务匹配算法引用或作为不同算法比较的基础[5-6]。

本文针对该算法匹配不精确的问题，提出一种基于本体概念相似度的 Web 服务匹配算法来匹配服务的语义信息。

2 OWS-S/UDDI 匹配算法

2.1 OWL-S，本体及分类树

在 OWL-S 中，服务的功能用服务的输入、输出、前提和结果表示[3]，服务的功能匹配表现为服务需求方和服务发布方的输入、输出、前提和结果的匹配。

图 1 是一个表示书籍的分类关系的本体。

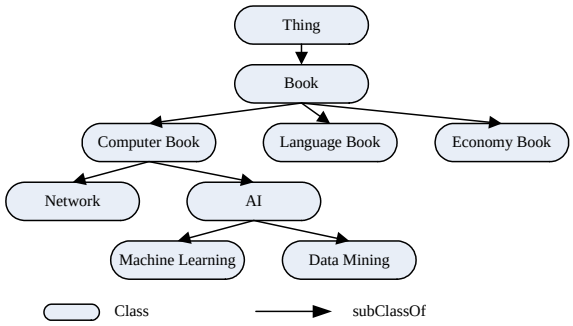

图 1 表示书籍分类关系的本体

在语义 Web 服务中，服务需求和发布双方一般采用共同的领域本体来准确表示服务的输入、输出、前提和结果中的信息。通过对本体中概念关系的推理和分析，可知服务需求和发布方的匹配程度。本体中最主要的关系是概念之间的子类关系(subClassOf)，也称继承关系[7]。由子类关系可以定义

基金项目：国家自然科学基金资助项目(90604017)；国家“973”计划基金资助重点项目(2003CB317004)
作者简介：彭 晖(1969－)，女，副教授、在职博士研究生，主研方向：语义 Web 服务，Web 数据挖掘，CSCW；史忠植，研究员、博士生导师；邱莉榕、常 亮，博士研究生
收稿日期：2007-12-20 **E-mail：**pengh@ics.ict.ac.cn

概念之间的包含关系。如果概念 A 是概念 B 的子类(subclass)，则概念 B 包含(subsums)概念 A。包含关系是可传递的。考虑概念之间的单继承关系，本体可以表示成一棵分类树。

2.2 OWL-S/UDDI 服务匹配算法

OWL-S/UDDI 匹配算法是利用分类树中概念之间的包含关系来判断服务的需求发布方的匹配程度。

OWL-S/UDDI 匹配算法的核心思想是，如果请求的服务的输入包括了发布的服务的输入，并且发布的服务输出包括了请求的服务的输出，则请求的服务与发布的服务在语义上是匹配的。设请求的某个输出为 outR，发布的某个输出为 outA，则每一对输出的匹配算法可描述为[4]：

```
degreeOfMatch(outR,outA):
if outA=outR then return exact
if outR subclassOf outA then return exact
if outA subsumes outR then return plugIn
if outR subsumes outA then return subsumes
else return fail
```

每一对输入的匹配结果与输出的匹配的包含关系是相反的，即 degreeOfMatch(outR,outA)应用到每一对输入的匹配为 degreeOfMatch(inA, inR)。

根据算法描述，如果请求的服务的输出与发布的服务的输出相同或请求的输出是发布的输出的子类，返回结果为精确匹配(exact)；如果发布的输出包含请求的输出，但不是其直接父类，返回结果为可替代匹配(plugIn)；如果请求的输出包含发布的输出，返回结果为包含匹配(subsumes)；否则为不匹配(fail)。

3 基于本体概念相似度的 Web 服务匹配算法

3.1 OWL-S/UDDI 算法匹配结果分析

OWL-S/UDDI 匹配算法将本体中概念的匹配分为精确匹配(exact)、可替代匹配(plugIn)、包含匹配(subsumes)和不匹配 4 类。

在匹配算法中，不存在包含关系的概念之间，认为其没有语义联系，返回结果为“不匹配(fail)”。这一点符合 Web 服务的匹配要求。比如在图 1 中如果请求的服务是查询“computer book”，而发布的服务是查询“和 history book”，虽然 2 个概念都是“book”的子类，但因为发布的服务不能满足请求，则认为两者之间是不匹配的。

对于可替代匹配由于发布的服务的输出包含了请求的服务的输出(或请求的服务的输入包含了发布的服务的输入)，而包含匹配是请求的服务输出包含的发布的服务的输出(或发布的服务的输入包含了请求的服务的输入)，因此可替代匹配的匹配度高于包含匹配。在图 1 中，如果请求查询“data mining”书籍，发布“book”查询的服务通常情况下是可以满足请求的，这时返回结果为“可替代”匹配。如请求查询任意的“book”，发布“computer book”查询的服务只有少数情况能满足请求，这时返回结果为“包含”。因此，4 类匹配的匹配度由高到低的排序是：精确匹配，可替代匹配，包含匹配和不匹配。

OWL-S/UDDI 算法通过对本体中类的包含关系的推理，给出服务发布方和需求方之间的匹配等级，通过返回不同匹配等级的服务提高的服务的查准率和查全率，但它最大的缺点在于不能给出服务之间的精确匹配，影响了服务匹配质量。如图 1 所示，如果需求方的输出是 Machine Learning，提供方的输出不管是 AI, Computer Book 还是 Book，返回的结果都是可替代匹配，但其实三者的匹配程度相差是很大的，这不利于在大量的服务中准确地查找所需的服务，为了解决以上问题，本文提出用本体概念相似度来进行服务的精确匹配。新的算法要求既保持原算法的匹配等级的合理性，又能提供精确的匹配。

3.2 本体概念相似度

对于表示本体的分类树，可以用 2 个不同节点之间的距离来衡量节点概念之间的相似度。为了满足服务匹配的要求，定义分类树中节点的距离如下：

定义 1 分类树定义为一棵有向树，对于树中的每一条有向边<*vp*,*vq*>，如果 *vp* 是 *vq* 的父节点，*vq* 是 *vp* 的子节点，则 *vp* 包含 *vq*。

定义 2 对于分类树中的任意 2 个节点 *vp*,*vq* 的距离 distance(*vp*,*vq*)，定义为：

(1)如果 *vp* 与 *vq* 为树中相同节点，则 distance(*vp*,*vq*)=0。

(2)如果从节点 *vp* 没有路径到达 *vq*，且从节点 *vp* 也没有路径到达 *vq*，则 distance(*vp*,*vq*)=∞。

(3)如果从节点 *vp* 到达节点 *vq* 有路径，则 distance(*vp*,*vq*)为从 *vp* 到达 *vq* 的路径的长度。

(4)如果从 *vq* 到达 *vp* 有路径，则 distance(*vp*,*vq*)为从 *vq* 到达 *vp* 的路径长度的负数。

例如，对于图 1 有：

distance(computer book, computer book)=0, distance(computer book, machie learning)=2, distance(machie learning, computer book)=-2, distance(computer book, language book)=∞

定义 3 分类树中 2 个节点所表示的概念的相似度定义为：假设节点 *vp* 表示的概念为 C_{vp}，节点 *v*,*q* 表示的概念为 C_{vq}。

$$sim(c_{vp},c_{vq})=\begin{cases}1 & \text{if } distance(vp,vq)=0\\ 1/distance(vp,vq)+1 & \text{if } distance(vp,vq)>0\\ \frac{1}{2}+1/|distance(vp,vq)|+1 & \text{if } distance(vp,vq)<0\\ 0 & \text{if } distance(vp,vq)=\infty\end{cases}$$

通过以上定义，分类树中 2 个概念的匹配程度是 0~1 之间的一个具体实数，数值越大，节点之间的距离越短，节点所对应的概念的相似度越高。按以上定义，原 OWL-S/UDDI 算法中的“精确匹配”的相似度为 1，“不匹配”的相似度为 0，“可替代匹配”的相似度为一个大于 0.5 小于 1 的实数，“包含匹配”的相似度为一个大于 0 小于等于 0.5 的实数。例如，对于图 1 有：

sim(computer book, computer book)=1, sim(computer book, machie learning)=1/3, sim(machie learning, computer book)=5/6, sim(computer book, language book)= 0

这样定义的相似度匹配既保持了 OWL-S/UDDI 算法的匹配等级的合理性，又提供了精确的匹配。如果发布的服务与请求的服务中有多个输入参数和输出参数，则可以取匹配度最高的参数来匹配某一个请求的输入或输出，最终的匹配结果可以是请求服务所有输入输出参数的加权平均值。

4 算法性能分析与测试

北京市应急联动与社会综合保障系统(GEIS[8])，集成了北京市警务、交通和医疗方面的信息，为紧急事件提供实时处理。全市警务、交通和医疗部门各自开发了不同的数据库信息系统，也开发了大量的 Web 服务来处理相应数据库系统中的信息，如：预警查询，危险源查询，警情分析，车辆查询，车辆调度，医院查询等。建立起 Web 服务所使用的领域本体，并用 OWL-S 描述各服务，形成语义 Web 服务描述文件，将这些描述文件发布到 GEIS UDDI 中，当接到应急事件，

将应急事件的需求与 GEIS UDDI 中的服务进行匹配，将处理结果返回给用户。从 GEIS 系统中分别选取 50 个警务信息处理 Web 服务，50 个交通信息处理 Web 服务，50 个医疗信息处理 Web 服务共 150 个服务，分别采用 OWL-S/UDDI 匹配和本文的精确匹配，要求返回的结果分别为精确度为：0.2 以上，0.4 以上，0.5 以上，0.6 以上，0.8 以上或为 1.0 的服务。对于要求返回精确度为 0.2 以上或 0.4 以上的服务，OWL-S/UDDI 匹配取包含匹配的结果；对于要求返回精确度为 0.5 以上、0.6 以上或 0.8 以上的服务，OWL-S/UDDI 匹配取可替代匹配的结果，实验结果如图 2、图 3 所示。

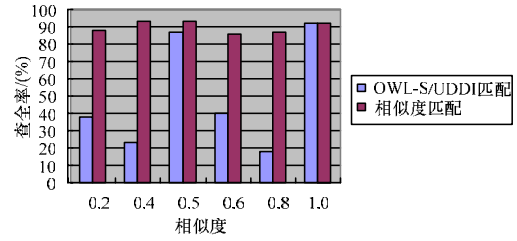

图 2　2 种算法的查准率比较

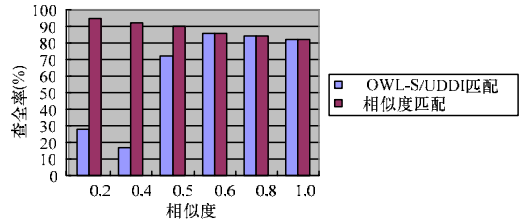

图 3　2 种算法的查全率比较

试验结果得出相似度匹配算法的平均查准率是 OWL-S/UDDI 匹配算法的 1.8 倍，平均查全率是 OWL-S/UDDI 匹配算法的 1.4 倍，在相似度等于等级的划分界线时，如 0.5, 1 时，两者的匹配程度是相近的。

5　结束语

本文提出了一种本体概念语义相似度的 Web 服务匹配方法，该方法能精确匹配用本体概念描述服务请求方和发布方的功能，且匹配结果与 OWL-S/UDDI 匹配的等级保持一致。实验结果证明，该算法比 OWL-S/UDDI 匹配方法具有更高的查准率与查全率。

参考文献

[1] Web Service Description Language[EB/OL]. (2007-06-26). http://www. w3.org/TR/2007/REC-wsdl20-20070626.

[2] Universal Description, Discovery and Integration(UDDI)[EB/OL]. (2005- 05-20). http://www.uddi.org/specification.html.

[3] The OWL Services Coalition. OWL-S: Semantic Markup for Web Services[EB/OL]. (2004-05-20). http://www.w3.org/Submission/OWL-S/.

[4] Paolucci M, Kawamura T, Payne T R, et al. Semantic Matching of Web Services Capabilities[C]//Proceedings of the 1st International Semantic Web Service. Las Vegas, Nevada, USA: [s. n.], 2003.

[5] 吴　健, 吴朝晖, 李　莹, 等. 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): 595-601.

[6] 胡建强, 邹　鹏, 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.

[7] 李善平, 尹　奇, 胡玉杰, 等. 本体论研究综述[J]. 计算机研究与发展, 2004, 41(7): 1041-1052.

[8] GEIS[Z]. (2006-10-11). http://www.intsci.ac.cn/geis/.

（上接第 43 页）

即 $|S_a(e_{11})|=3$, $|S_a(e_8)|=48$，则

$$\boldsymbol{R}(e_2,E_2)=\max\{(0.34+0.85)/2,(0\times3+0\times48)/(3+48)\}=0.595$$

由于这个数字高于与 e_2 相关的所有实体对的亲和度，因此应该把 e_2 放入实体组 E_2 中去。从现在开始，每当遇到与 e_{11},e_8 或 e_2 相关的实体，就计算它对实体组$\{e_{11},e_8,e_2\}$的亲和度，这样，高亲和度的实体组就会稳定增长。

(2)实体对中的 2 个实体都在已经形成的实体组中。该实例中接情况(1)继续取实体对，下一个实体对应该是 $E_6=\{(e_7,e_4)|r_{7,4}=0.76\}$，但 e_7 和 e_4 已经分别处在 2 个不同的实体组 E_3 和 E_1 中，是否要把这 2 个实体组合并起来呢？决定这一点，要计算 e_7 对$\{e_1,e_4\}$的亲和度和 e_4 对$\{e_6,e_7\}$的亲和度，假定求出的值分别为 0.90 和 0.37，且与 e_7 相关的所有实体对的亲和度都小于 0.90，则把 e_7 归入 E_1 中，得到新的实体组$\{e_1,e_4,e_7\}$，再计算 e_6 与$\{e_1,e_4,e_7\}$的亲和度，假定得到的值为 0.89，高于与 e_6 相关的所有实体对的亲和度，则把 e_6 归入$\{e_1,e_4,e_7\}$中，得到新的实体组$\{e_1,e_4,e_7,e_6\}$。

在亲和度矩阵中，最后遍历到的一些值都比较小，这些实体也许与任何其他实体都很少相关，可以用文件系统或独立的简单数据库来实施。设计人员应当仔细考察剩余的低亲和度的实体对，以确定它们是否属于已有的某个数据库。

最后分组结果还应当由规划小组和部门代表进行审核，并做出必要的调整，以便以后进行主题数据库的详细数据模型设计。

这种规划主题数据库的方法已经用于某汽车轮毂厂的实际规划设计中，结果取得了很好的实际应用效果，证明该方法具有可行性。

5　结束语

本文给出了业务模型的形式化描述，建立主题数据库时引进了数学公式，使实际问题转化为数学问题，给出了一个有依据可循的规范化方法，使整个主题数据库的规划过程更加严谨规范，对于今后相关问题的研究和企业数据环境的改造和重建工作起到一定的指导作用。值得一提的是，对于前面给出的概念、形式化描述和计算公式，如实体和活动的关系，设计出了对实体聚类的算法，即建立主题数据库过程的半自动化，由计算机得到初稿，再加以人工修改产生企业主题数据库的最终规划，这种采用计算建立企业主题数据库的方法需要的时间少，尤其是对于实体和活动数量大、实体和活动关系复杂以及信息资源规划人员经验少的情况，更有用武之地。

参考文献

[1] 高复先. 数据分析与数据模型[J]. 汽车情报, 2002, (15): 2-3.

[2] James M. Strategic Information Planning Methodologies[M]. 2nd ed. [S. 1.]: Prentice Hall, 1989.

[3] 高复先. 信息资源规划——信息化建设基础工程[M]. 北京: 清华大学出版社, 2002.

[4] 王玉书, 董丕明. 主题数据库规划合理性估计的数学公式[J]. 软件学报, 1997, 8(2): 93-98.

[5] 高复先. 营造数据环境[J]. 中国计算机用户, 2002, (44): 45-45.