文章编号:1001-5132（2010）03-0038-06

基于混合模型语义 Web 服务匹配机制的研究与实现

张迎新，潘善亮*

（宁波大学 信息科学与工程学院，浙江 宁波 315211）

摘要：服务发现和匹配是面向服务计算架构需要解决的核心问题之一，而现有语义 web 服务发现机制适应范围较窄，匹配效率较低，具有较大提升空间. 提出了一种基于服务属性和功能描述的混合服务匹配方法，该方法利用服务本体来扩展服务特征信息，采用 LSA 方法进行服务相似匹配以提高服务的查全率，再利用本体推理机制来提高服务的查准率. 实验证明：此种混合方法能提高服务发现效率.

关键词：Web 服务；服务发现；潜在语义分析；本体

中图分类号：TP391 文献标识码：A

Web 服务是解决 Web 上软件发布和共享的新技术，它的目标是使得 Web 上软件的发布和共享与 Web 上数据的发布和共享变得同样简单和方便，Web 服务同时也是一个自描述、自包含、松耦合、模块化的应用模式. Web 服务匹配就是查找能够满足用户需求的服务的过程. 语义 Web 服务的主要方法是利用 Ontology 来描述 Web 服务，然后通过这些带有语义信息的描述实现服务的自动发现、调用和组合. 目前有 2 种流行的语义 Web 服务描述框架：ESSI 组织提出的 Web 服务模型本体框架(Web Service Modeling Ontology, WSMO)和 Darpa 组织提出的 Web 本体语言描述服务框架(Ontology Web Language for Services, OWL-S). 由于 OWL-S 对 Web 服务领域标准和语义 Web 领域标准的兼容性较好，并且具有开放灵活的定义方式，正逐渐成为语义 Web 服务描述框架的推荐标准.

服务匹配发现技术是面向服务计算框架的首要和关键问题. 目前，Web 服务发现方法主要有两类：(1)在语法级上实现依赖于关键字匹配的服务发现机制，服务描述一般采用 WSDL，如 IBM 的 UDDI 系统就是基于预定义分类的语法级服务发现方法，该方法实现简单，但结果不理想；(2)基于语义推理的服务发现方法，称之为语义 Web 服务发现，它的服务描述一般采用 WSMO 或 OWL-S 语言. 随着语义 Web 服务研究的发展，此类方法是当前研究的热点. 研究者在语义 Web 服务发现的方法上已经做了许多工作，如文献[1]提出了根据 Web 服务的输入、输出对应概念的上下位(subClassOf)关系将服务语义相似性分为 Exact、Plug-in、Subsumes、Fail 等 4 种类型，文献[2-5]均在文献[1]的基础上做了改进. 但这些方法对服务相似性区分过于简单，仅利用概念间的上下位关系来计算概

收稿日期：2009-11-20. 宁波大学学报（理工版）网址: http://3xb.nbu.edu.cn
基金项目：浙江省自然科学基金（Y107751）；宁波市博士基金（2004A610004）.
第一作者：张迎新（1983－），女，吉林蛟河人，在读硕士研究生，主要研究方向: Web 服务发现与组合. E-mail: zyx28366@163.com
*通讯作者：潘善亮（1970－），男，浙江宁波人，博士/副教授，主要研究方向: Web 服务及信息检索. E-mail: panshanliang@nbu.edu.cn

念间的相似度. 笔者在此提出了一种基于 LSA 和 GCSM (Generalized Cosine-Similarity Measure)的混合的服务相似计算方法，用于服务发现中，实验结果表明该方法具有更好的查全率和查准率.

1　相关研究

目前，随着 Web 服务技术的广泛应用，网络上存在大量并不断增长的 Web 服务，服务的动态发现、组合、执行及监控成为面向服务计算(Service-Oriented Computing, SOC)得到进一步广泛应用的主要障碍. 而 UDDI 系统支持的基于关键词的服务匹配机制仅实现了语法层次上的匹配，导致许多语义上匹配的服务不能够被发现，服务发现的性能低下.

潜在语义分析(latent Semantic Analysis, LSA)是一种基于潜在概念索引的检索技术[6-7]，被广泛应用于信息检索、文本分类、自动问答系统等领域中. LSA利用奇异值分解(Singular Value Decomposition, SVD)生成的潜概念(Latent Concept, LC)索引来进行信息检索. 近来，潜在语义分析方法被用于Web服务发现[8-12]，如文献[8]提出了一种基于服务语料库的启发式半自动Web服务分类方法；文献[9]利用潜在语义分析(LSA)获得简短的Web服务文本描述间的语义关系; Corella和Paliwal[10]等提出了一种本体和潜在语义分析相结合的服务发现方法；Sajjanhar等人设计了一种基于奇异值分解的Web服务匹配算法[11].

语义Web服务作为Web服务的语义扩展，在服务描述中添加了丰富的语义信息. W3C组织在DAML-S基础上提出来的一个基于OWL的Web服务本体——OWL-S[13]. 它通过提供一个核心的构造集使得服务提供者能够以清晰的、计算机可理解的方式描述服务的属性和功能等. 近来很多研究者已经提出了基于OWL-S的语义Web服务匹配，如文献[14]提出了一个基于Service Profile的Web服务发现方法，该方法采用Service Profile描述用户请求，并将该描述与注册Web服务的Service Profile进行匹配. 匹配过程中利用OWL所描述的本体知识，对服务输入/输出概念与用户请求的输入/输出概念进行语义匹配.

笔者提出的基于 LSA 和 GCSM 混合的服务相似计算方法，首先利用奇异值分解对源服务集进行初步的基于语法级的筛选，然后利用 GCSM 做进一步基于语义级的服务匹配，并通过 2 次筛选扩展了服务功能语义匹配方法，提高了服务查全率与查准率.

2　系统模型

笔者引入潜在语义分析和服务本体推理相结合的方法来进行服务的发现，且查询服务通过混合的方法计算目标服务集内服务的相似度，并根据相似度的大小发现合适的服务.

我们采用信息检索领域中潜在语义分析和服务本体语义推理相结合的方法来综合考虑服务的相似度. 目前，语义 Web 服务之间的语义距离经典的方法就是利用服务接口参数在本体中的上下位关系分为 Exact、Plug-in、Subsumes、Fail 这 4 种类型，它们相似关系的粒度太大，笔者引入 GCSM 方法来细化这种语义相似关系，并称该方法为语义推理方法，具体做法见以下分析. 语义推理方法的弱点是它仅考虑概念在本体树中概念间的继承关系，而忽略了概念之间的二元关系. 为弥补这个不足，我们引入 LSA 方法挖掘概念之间的潜在相似关系. LSA 方法能够利用概率统计的方法挖掘词之间、词与文档之间以及文档之间的潜在语义关系，该方法运用到本文中的目的是为挖掘服务接口参数和服务描述信息中这些词之间的语义关系.

系统包括以下几个步骤：(1)服务预处理：提取服务集中各服务的输入、输出参数和描述信息，形成服务-索引项二维特征矩阵；(2)查询服务的查询

接口扩展; (3)利用 LSA 计算查询服务与服务类中各服务的语义相似度; (4)利用 GCSM 方法计算查询服务与服务集中各服务的语义相似度. 图 1 为笔者提出的语义 Web 服务检索算法流程.

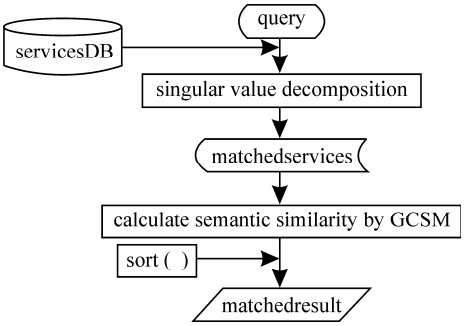

图 1 Web 服务检索

3 算法与实现

3.1 构建服务-索引项矩阵

利用OWL-S查询接口，读取OWL-S服务的输入参数、输出参数和服务描述信息，对这些信息做预处理，如标点符号处理、常用词消去处理、索引项大小写转换等. 预处理的结果是包含索引项(即出现在输入参数、输出参数和服务描述信息中的那些词)的 1 个服务特征向量 $S_i=\{t_1,t_2,\cdots,t_m\}$，服务向量 S_i 中的 t_i 代表服务索引项term$_i$的权值. 采用矩阵保存已经注册的n个服务的每个term的权值. 即: $T=\{S_1,S_2,\cdots,S_n\}$,

$$T=\begin{pmatrix} t_{11} & t_{12} & \cdots & t_{1n} \\ t_{21} & t_{22} & \cdots & t_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ t_{m1} & t_{m2} & \cdots & t_{mn} \end{pmatrix}.$$

矩阵列代表服务，行代表服务中的每个索引项，t_{ij} 是服务 j 的索引项 i 的权值. 权值的统计方法有很多，笔者采用 TF*IDF (Term Frequency-Inverse Document Frequency)算法计算每个服务 I/O 中的每个索引项(term)的权重值，将相应服务的权重值列表保存到矩阵中. 权值计算公式为 $W_i=tf_i\cdot idf_i=tf_i\cdot(\log(N/n_i)+1)$，其中，$tf_i$ 表示每个索引项在某个 Web 服务特征中的重要程度(在这里就是词出现的频率); idf_i 表示 1 个索引项对于区分 Web 服务的重要程度; N 为整个服务集中的服务个数; n 为包含该索引项的 Web 服务个数.

3.2 查询服务的查询接口扩展

为简便系统的实现，用户的查询也是用 1 个 OWL-S 描述的服务，称为查询服务，我们的目标就是找到与该查询匹配的服务. 同样，查询服务也要做相应的标点符号处理、常用词消去处理、索引项大小写转换等预处理. 为了提高系统的查全率，对查询服务的接口做如下扩展(也称为概念展开): 在服务本体树中，搜索以查询服务索引词为叶子节点的所有上层概念，并将这些概念加入到查询服务的索引词集合中.

由于查询服务的本体信息可能在需要操作的不同的本体文件中，我们可通过本体间的连接来实现概念扩展. 查询服务索引词的扩展实际上是扩展了服务的特征向量，这样就能极大可能地找到相关服务，提高服务的查全率.

3.3 利用 LSA 做服务发现初步筛选

通过以上处理，得到注册服务的服务-索引项矩阵和查询服务的查询特征向量，系统利用 LSA 方法做服务匹配的初步筛选. LSA 利用截断的 SVD 生成的低维潜在语义空间来描述元素间的语义结构. 它不同于向量空间模型(Vector Space Model, VSM)，潜在语义空间中对应于向量各个维度的不再是各个元素，而是存在于元素之间的潜在概念，其本质就是用潜在概念的线性组合来描述服务特征参数.

服务-索引项矩阵 T 是 1 个典型的稀疏矩阵，因为很多索引项是不会出现在大多数服务中的. 所以，通过对 T 矩阵的奇异值变换可以降低矩阵的维度，将索引项和服务之间的关系在更少、更能表示其特征的语义空间中表示出来. 通过 SVD，矩阵 T 可以分解为 3 个矩阵的乘积:

$$T\approx T_k=U_k\Sigma_k V_k^{\mathrm{T}},$$

其中，$U_k^{\mathrm{T}}U_k=V_k^{\mathrm{T}}V_k=I_k$，$U_k$ 和 V_k 的列分别被称

为矩阵 T_k 的左、右奇异向量；Σ_k 是对角矩阵，对角元素被称为矩阵 T_k 的奇异值．U_k 矩阵中的行向量对应原矩阵T的索引项向量，V_k 矩阵中的行向量则对应原矩阵T的Web服务向量．这里的 U_k 矩阵和 V_k 矩阵中的单个项不一定是非负数，索引词之间以及Web服务之间的关系是通过向量行间的相关关系来获得．U矩阵最初的k个左奇异向量构成 $m\times k$ 矩阵U_k，为了用k维向量 $s^{(k)}$ 近似地表示服务向量，考虑把向量s投影到U_k空间，则服务向量s在 U_k下的坐标用 $s^{(k)}=U_k^{\mathrm{T}}s$ 求得．

服务查询向量与服务向量相同，也根据到U_k张成的空间投影，可以用k维向量表示．

查询服务与所有服务间的相似度可采用求余弦值方法得到，选取合适阀值作为选取服务界限．服务向量 s 和查询向量 q 间的相似度 $sim\_l(d,q)$ 根据下式进行计算:

$$sim\_l(d,q)=\cos(U_k^{\mathrm{T}}d,U_k^{\mathrm{T}}q)=\frac{(U_k^{\mathrm{T}}d)\cdot(U_k^{\mathrm{T}}q)}{\|U_k^{\mathrm{T}}d\|\cdot\|U_k^{\mathrm{T}}q\|}.$$

3.4　基于本体做进一步服务参数匹配

通过 3.3 节描述的基于 LSA 方法计算出来的服务相似度仅考虑了服务特征信息的词相似，它可以用来初步筛选服务，但是不能进一步进行服务的精确匹配．我们此时可考虑基于本体的服务功能参数的语义匹配方法．

一般的服务匹配必须满足以下条件：(1)候选服务的输出满足请求服务的输出；(2)候选服务的输入满足请求服务的输入，即服务正常运转所需要的输入必须有用户请求提供．但由于用户在请求时不知道存在什么样的服务，而服务在制定时也不能预期会存在什么样的服务请求，因此服务完全匹配只是一种理想状态，在实际应用中进行匹配时绝大部分不是完全匹配．文献[5]引入弹性匹配算法来解决此问题，该方法将匹配结果划分为四种类别：完全匹配(Exact)、插拔匹配(Plug-in)、包含匹配(Subsume)、匹配失败(Fail)，这种分类方法还是有问题的．例如，某服务将输出参数声明为 owl:Thing，那么它就可能与所有的服务均是 Plug-in 匹配关系，这样就很难区别同样是插拔匹配中的服务 A 和服务 B 的匹配度．

针对上述问题引入基于GCSM算法[14]的Web服务语义相似度概念，利用Web服务输入、输出参数中索引项的语义相似度来衡量服务的相似度．在Web服务的领域本体树中，索引项是该树中的结点，可用如下公式来定义 2 个索引项的语义相似度:

$$sim\_d(uri1,uri2)=\frac{2\times depth(LCA(uri1,uri2))}{depth(uri1)+depth(uri2)},$$

其中，$uri1$ 和 $uri2$ 分别表示为 2 个索引项的引用；$depth(uri1)$ 和 $depth(uri2)$ 分别表示 2 个索引项在本体树中的深度；$depth(LCA(uri1,uri2))$ 表示 $uri1$ 和 $uri2$ 在本体树中最近共同祖先的深度．可以看出 $Sim\_d(uri1,uri2)$ 的值区间为(0,1)，当 2 个索引项相同时相似度为 1．通过以上计算方法，可以得到 2 个服务参数匹配相似度的具体数值，该数值在区间(0,1)，值越大越相似．

3.5　算法实现

为简化系统的实现，查询服务是 1 个 OWL-S 服务文件．系统利用服务本体信息对查询服务做查询参数扩展，然后计算经过 SVD 分解的候选服务集中的服务与查询服务的潜在词义相似度，得到相似度大于某个阀值的经过筛选的候选服务集，在这个新的候选服务集上再利用 GCSM 方法做查询参数匹配得到语义相似度，潜在词义相似度和语义相似度都是介于 0 和 1 之间的实数．文中将潜在词义分析与本体推理语义相似相结合的方法来实现服务检索．

潜在词义分析与本体推理相结合的服务检索算法称为 SG 算法．该算法的输入是注册服务集的服务-索引项矩阵 T、查询服务 q 以及相似度阈值 α，输出是匹配的服务集合 MR，其中包括匹配的服务名称及相应的匹配度，并在语义空间中搜索与查询相关的服务．

4 实验分析

为检验 SG 服务匹配算法的效果，我们初步实现了 1 个查询匹配原型系统，将 SG 算法与经典的语义匹配算法 LS 算法(Logic Semantic，该算法在文献[1]中已经详细描述)做比较分析. LS 方法是基于本体的经典服务发现方法，它只给出 4 种语义匹配度{Exact, Plug-in, Subsume, Fail}.

我们的处理对象是用OWL-S描述的语义Web服务，使用马里兰大学提供的OWLS-API接口读取服务信息，并借助Pellet进行本体概念推理. 实验数据集来自OWLS-TC Version 2[16].

4.1 性能评估

为评估实验的检索性能，采用Van Rijsbergen提出的Precision-Recall微评估策略. 用Q表示测试查询集合，A为在Q中与所有查询相关文档的和，A_R表示注册服务中与查询$R \in Q$相关的服务集合. 对于每个查询R，我们考虑$\lambda = 20$衡量查全率最大值，$B_{\lambda R}$为在每个λ点的相关文档数；同理，我们利用在每个λ点检索到文档数$B_{\lambda R}$衡量相应的查准率. 查全率查准率微平均覆盖了所有查询，评估函数定义如下:

$$R_\lambda = \sum_{R \in Q} \frac{|A_R \cap B_{\lambda R}|}{|A|}, P_\lambda = \sum_{R \in Q} \frac{|A_R \cap B_{\lambda R}|}{|B_\lambda|}.$$

4.2 实验结果

例如我们注册了 576 个服务到匹配机中，用户要查询服务 q:car_price_service，匹配机中有 21 个服务与查询请求 q 相关. 为检验 SG 算法性能，我们利用 LS 与 SG 算法进行检索，查询结果分析如图 2 所示.

从实验中，可以得出如下结论：基于本体的经典服务发现 LS 方法的查全率与查准率都比较低，原因是该算法仅是基于 OWL-S 的功能语义匹配，仅考虑在本体树中的概念之间的继承关系，而忽略了概念之间的二元关系，未能挖掘服务特征某些概念之间潜在的语义关系，并且服务功能接口

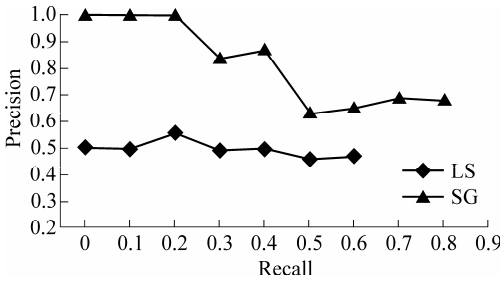

图 2 LS/SG 性能的 Recall-Precision 曲线

匹配关系仅分为 4 类，而笔者提出的 SG 匹配算法将潜在语义分析与本体推理相结合，通过 SVD 算法挖掘概念间的潜在语义关系，提高了服务查全率；并通过把服务接口匹配关系扩展到无限，利用本体做进一步服务参数匹配，提高服务查准率. SG 算法还给出了一个精确的服务匹配程度，此举更加有利用户选择精确的服务.

5 结语

Web 服务检索技术已成为制约 Web 服务进一步发展的一大瓶颈，笔者提出了潜在语义分析与本体推理相结合的算法，综合考虑了 Web 服务中潜在词义相似度与语义相似度，开发了原型系统，进行了性能测试. 实验结果表明，潜在语义分析与本体推理相结合的服务检索算法具有较高的查全率与查准率.

参考文献:

[1] Paolucci M, Kawamura T, Payne T R. Semantic matching of Web services capabilities[C]//Proc of the First Intl Semantic Web Conference, Italy: Sardinia, 2002:333-347.

[2] Tang S. Matching of Web service specifications using DAML-S descriptions[M]. Berlin: Technische University, 2004.

[3] Bramantoro A, Krishnaswamy S, Indrawan M. A semantic distance measure for matching Web services[C]// WWW2005 Web Service Semantics Workshop, SPRINGER-VERLAG, Germany, 2005:217-226

[4] 彭晖，史忠植，邱莉榕，等. 基于本体概念相似度的语义 Web 服务匹配算法[J]. 计算机工程, 2008, 34(15):51-

53.

[5] Ge Jike, Qiu Yuhui. Concept similarity matching based on semantic distance[C]//SKG APOS 2008 4th International Conference, Semantics, Knowledge and Grid, 2008: 380-383.

[6] Ling Hongfei, Yao Tianshun. Text browsing based on latent semantic indexing[J]. Journal of Chinese Information Processing, 2000, 14(5):241-245.

[7] Landauer T K, Foltz P W, Laham D. Introduction to latent semantic analysis[J]. Discourse Processes, 1998, 25:259-284.

[8] Ganesan P, Molina G H, Widom J. Exploiting hierarchical domain structure to compute similarity[J]. ACM Transactions on Information Systems, 2003, 21(1):64-93.

[9] Ma Jiangang. Web services discovery based on latent semantic approach[C]//2008 IEEE International Conference on Web Services, 2008:740-747.

[10] Corella M A, Castells P. Semi-automatic semantic based Web service classification[C]//Proc of the International Conference on Knowledge-based Intelligent Information and Engineering Systems, Springer, 2006: 459-470.

[11] Sajjanhar A, Hou J, Zhang Y. Algorithm for Web services matching[C]//Proceedings of the 6th Asia-Pacific Web Conference, China: Hangzhou, 2004:665-670.

[12] 刘柏嵩，贺赛龙. 一种基于 Web 的分类体系学习算法[J]. 宁波大学学报: 理工版, 2008, 21(1):62-67.

[13] Artin M D, Burstein M, Hobbs J, et al. W3C member submission OWL-S: Semantic markup for Web Services [EB/OL]. [2004-11-05]. http://www.w3.org/Submission/OWL-S.

[14] Paliwal A V, Adam N R, Bornhövd C. Web service discovery: Adding semantics through service request expansion and latent semantic indexing[C]//IEEE International Conference on Services Computing, 2007:106-113.

[15] Zhang Po. The research and implementation of Semantic Based Web Services[D]. Beijing: Department of Computer Science and Technology of Qinghua University, 2005.

[16] Khalid M A, Saarbrücken B F. OWLS-TC: OWL-S service retrieval test collection version2[EB/OL]. [2005-11-15]. http://projects.semwebcentral.org/projects/owls-tc.

Research and Implementation of Semantic Web Service Matching Method Based on a Hybrid Model

ZHANG Ying-xin, PAN Shan-liang*

(Faculty of Information Science and Technology, Ningbo University, Ningbo 315211, China)

Abstract: The web service discovery and matching remain to be one of the core problems in Service-Oriented Computing Architectures, while the current available semantic web service discovery mechanism has narrower match scope and lower efficiency, thus it leaves much room for improvement of its performance. This paper proposes an approach for hybrid service matching based on service properties and functional description. The characteristics of services are extended using service ontology; the latent semantic analysis method is adopted for service matching to improve recall ratio; the reasoning mechanism is utilized based on ontology to increase precision ratio. The experiments show that the efficiency of service discovery is improved using proposed hybrid method.

Key words: web service; service discovery; latent semantic analysis; ontology

CLC number: TP391　　**Document code:** A

（责任编辑　章践立）