基于 WordNet 和 Kernel 方法的 Web 服务发现机制研究

王东睿,杨 庚,陈 蕾,张迎周
(南京邮电大学 计算机学院,江苏 南京 210003)

摘 要:目前,传统的基于语法的Web服务发现机制智能性较差,已无法满足用户需求。而基于语义的Web服务发现针对不同的领域和情景都需建立本体库,为发现机制的建立带来了一定的复杂性。将以上两者进行结合,提出一种基于WordNet和Kernel方法的Web服务发现机制。首先利用Kernel-WordNet-VSM对服务进行分类,其中,WordNet用作对抽取的特征向量降维,并采用Kernel函数计算向量之间的相似度。然后利用WordNet概念链中词之间的最短路径,从服务的功能属性方面,对用户的原始请求和服务进行语义层次上的匹配。从而在服务分类的精度上得到一定提高,在服务发现的智能性方面得到了一定的改善。

关键词:Web服务; 服务发现; WorldNet; Kernel函数
中图分类号:TP393 **文献标识码**:A **文章编号**:1673-629X(2010)12-0069-04

Web Services Discovery Mechanism Based on WordNet and Kernel

WANG Dong-rui, YANG Geng, CHEN Lei, ZHANG Ying-zhou
(College of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

Abstract: At present, because of the poor intelligence, the traditional syntactic-based service discovery has been unable to satisfy customer's expectation. For building ontology library in the different areas and situations, establishment of mechanism for the semantic-based Web service discovery is very complicated. Combining the above two approachs, a mechanism for Web service discovery based on WordNet and the Kernel function is proposed in this paper. First, choose Kernel-WordNet-VSM can be utilized to solve the services classification, and WordNet is used to make vector dimension reduction on the extracted feature, Kernel function also calculate the similarity between vectors. Then minimum path between the words in the concept chain of wordNet is used to match user's requests and service on the functional attributes of service based on semantic level. Thus in terms of intelligence service discovery has been improved to some extend while the accuracy is improved in classification of services on a certain.

Key words: Web services; services discovery; WorldNet; Kernel methods

0 引 言

Web服务是分布式计算技术发展中的一场新的革命,随着Web服务的广泛应用,如何准确的查找到用户所需要的服务变得越来越重要[1]。服务发现已成为了服务计算领域核心问题。现有的服务发现方法主要分为以下两类:

一是语法级的服务发现[2]。主要是从语法层上对服务的描述信息进行相似度的计算。例如,早期的UDDI框架及引入信息检索领域中的方法如VSM[3]对服务之间及请求进行匹配[4,5],但这些发现机制由于缺乏语义的支持,许多关键词在不同的领域都有不同的语义,因而无法准确完整地满足用户的查找需求。

二是基于语义的服务发现[6]。主要是借鉴了语义Web领域的一些方法和技术,通过利用领域本体中的概念和属性来对服务属性进行描述[6],利用逻辑推理机来进行语义层次上的匹配。例如卡耐基梅隆大学的DAML-S Matchmaking项目[7]。虽然这种方法能够增强服务发现的智能化水平,提高匹配的精确度,但是对于所有领域都要构建一个完整的本体库,并能很好

收稿日期:2010-05-04;**修回日期**:2010-08-07
基金项目:国家自然科学基金项目(60873231, 60973046);江苏省自然科学基金(BK2009426);江苏省高校自然科学基金(08KJB520006);网络与交换技术国家重点实验室开发基金(SKLNST-2008-1-04)
作者简介:王东睿(1984-),男,江苏南京人,硕士研究生,研究方向为计算机通信与网间互连技术;杨 庚,教授,博士生导师,研究方向为计算机网络与安全、计算机通信与网间互连技术等。

地应用到服务发现中,却是一个复杂而庞大的工作。

针对上述问题,本文提出了一种结合性方法,主要分为两个步骤。首先利用 VSM 机制[8]对服务进行分类,减轻下一步服务需求与服务匹配的难度。在 VSM 机制中,利用了 WordNet 对特征向量进行降维,并引入 Kernel 函数计算向量间的相似度,提高分类的精确度。然后,在客户请求和服务匹配阶段,从服务功能性的角度,采用 WordNet 概念链中概念间的距离这一具有语义层次的度量值来查找用户最终所需要的服务。

1 WordNet 特征及概念链

WordNet 是一个联机英语词汇检索系统,由 Princeton 大学研制[9]。它作为语言学本体库,同时又是一部语义词典,在自然语言处理研究方面应用非常广泛。在 WordNet 中,网络节点由字形(word form)标识,分为动词、形容词、副词和功能词等 5 种。节点之间的关系分为同义关系(synonymy)、反义关系(antonymy)、继承关系(hyponymy)、部分/整体关系(meronymy)、形态关系(morphological relation)等[10]。其中,除了形态关系主要处理字形之间的联系外,其他 4 种关系都是字之间的语义关系。

由同义词集合代表的大部分名词概念间的上下位关系(hypernymy - hyponymy)提供了很好的概念层次结构,处于更高层次的概念代表更抽象的意义;反之,处于较低层次的概念代表的意义就较具体。同时,WorldNet 的 API 提供了相应的函数可以得到概念链中两个词即两个节点的最小路径距离。

2 基于 WorldNet 和 Kernel 方法的服务发现机制研究

基于 WorldNet 和 Kernel 方法的服务发现机制模型如图 1 所示,首先利用 VSM 机制对服务进行分类,减轻下一步服务需求与服务匹配的难度。在 VSM 机制中,利用了 WordNet 对向量进行降维,并引入 Kernel 方法计算服务间的相似度,从而提高分类的精确度。然后,在服务请求和服务匹配这一阶段,从服务功能性的角度,采用 WordNet 概念链中概念间的距离结合 Wu - Palmer[11]等提出的一种改进的最短路径距离计算服务请求信息提取的关键词和 M 矩阵中关键词的语义相似度,从而找到用户最终所需要的服务。

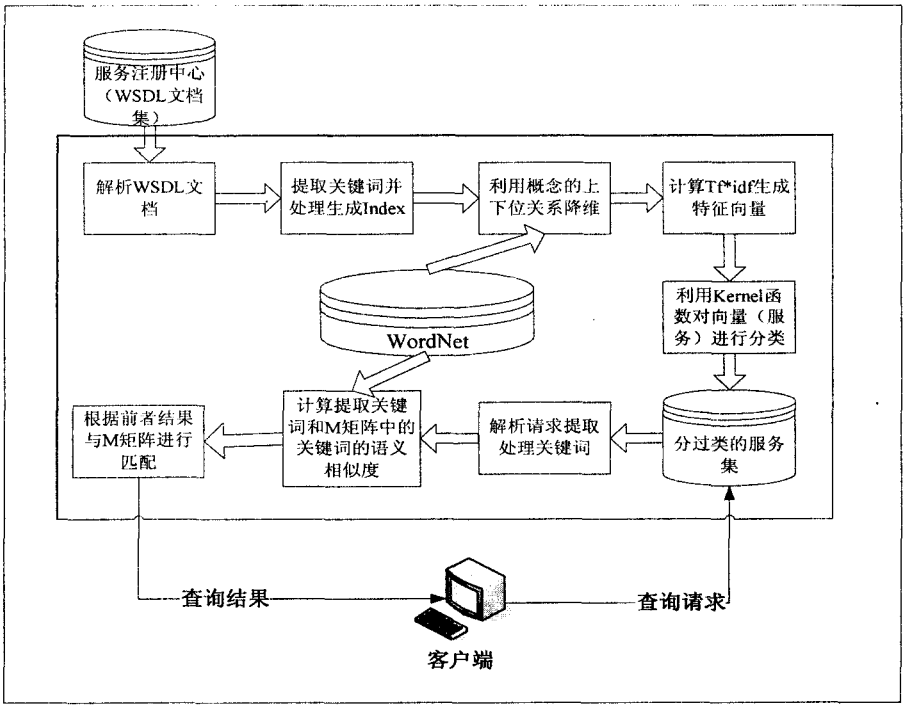

图 1 服务发现机制流程图

2.1 基于 WordNet 的特征向量生成

传统的向量空间模型将每个文档的查询都用等长的向量[5],即同一组 keyword 集合来表示。此时每个文档的 keyword 都被赋予一定的权重 w_{ij},用以表示 keyword i 对于文档 d_i 的重要程度,文档集 D 中文档 d_j 的权值向量表示为 $d_j = (w_{1j}, w_{2j}, \cdots, w_{nj}) = \sum W_{ij}t_1$,权重的 w_{ij} 大小由 tf * idf 方法来确定:

$$w_{ij} = tf_{ij} \times idf_{ij} = \log(1 + f_{ij}) \times \log(n/n_i) \quad (1)$$

其中,f_{ij} 表示 keyword i 在文档 d_i 中出现的次数,文档频率 df 为包含该 keyword 的文档数与所有文档数的比值,逆文档频率通常使用 $\log(n/n_i)$ 计算得到,其中 n_i 为包含 keyword i 的文件个数。

同理,对于 Web 服务,文献[9]中将 WSDL 文档作为抽取文件。WSDL 文档主要从 operation ,message , type ,description 四个方面描述了服务的功能,绑定方式和一些基本属性等。在这里借鉴其方法并选取了 operation 和 message 作为抽取部分。

传统空间向量模型并没有考虑到特征向量中不同的 keyword 是否具有同义或包含和被包含的语义关系,这不仅导致了向量的维数过大,向量过于稀疏,还导致了本是代表同一概念的两个词却分别用两个不同的词来处理。为弥补这一缺陷,引入 WordNet 中的上

位关系(hypernymy)/下位关系(hyponymy)。在生成特征向量之前,也就是对解析提取过的 keyword 处理过后再加处理,如果其中有两个词之间有直接上下位关系,那么对它们进行合并成上位词。

2.2 基于 Kernel 方法的相似度计算

传统空间向量模型在计算相似度时通常采用向量余弦夹角,欧式距离和 Minkowski 距离。然而,这里提取出的特征向量组成的是一个较大的稀疏矩阵,对于这些特征向量进行分类,是一个非线性,高维模式分类问题,如果采取传统的分类方法,由于线性不可分而严重影响到精确度。因此,在这里引入了 Kernel 函数。

Kernel 函数在机器学习中特别是支持向量机(SVM)[13]中已经得到成功应用。SVM 特别适合解决非线性,高维模式分类识别问题。采用 Kernel 函数,需要满足 Mercer 条件[12],我们已通过验证,该问题满足 Mercer 条件。更深入的理论研究参考文献[12]。

下面是目前主要研究的三种 Kernel 函数:

(1) Polynomial Kernel

$$K(x,y)=(a*<x,y>+c)^p, a>0, c\geqslant 0, p\in N \tag{2}$$

(2)Sigmoid Kernel

$$K(x,y)=\tanh(a*<x,y>+\beta), a>0, \beta>0 \tag{3}$$

(3)Radial basis Kernel

$$K(x,y)=\exp(-p\sum_{i=1}^{n}|x_i^a-y_j^\beta|^b), p>0, a>0, 0<b\leqslant 2 \tag{4}$$

现在,可以将 Kernel 函数用在服务相似度匹配中,过程如下:

步骤 1:将特征向量 v_{ws} 映射到高维空间 F:

$$\phi: v\in VS\subseteq R^n \mapsto \phi(v_{ws})\in F\subseteq R^H (n<<H)$$

步骤 2:计算基于 Kernel 函数的高维 F 空间的相似度:

$$K(x,y)=<\phi(x),\phi(y)> \tag{5}$$

$$\text{Similarity}(ws_i,ws_j)=\text{Cos}(\phi(v_{ws_i}),\phi(v_{ws_j}))=\frac{<\phi(v_{wsi}),\phi(v_{wsj})>}{\sqrt{<\phi(v_{wsi}),\phi(v_{wsj})>}*\sqrt{\phi(v_{wsi}),\phi(v_{wsj})>}}=\frac{K(v_{wsi},v_{wsj})}{\sqrt{K(v_{wsi},v_{wsj})*K(v_{wsi},v_{wsj})}} \tag{6}$$

2.3 基于 WordNet 概念距离的服务功能性查询

上面两小节解决了服务的分类问题。下一步的关键问题就是如何帮助客户在某一类服务中找到满足其请求信息的具体服务:

首先,利用 WSDL4J 对某一类服务集的 WSDL[13] 文档的 operation 部分进行解析,并抽取 keywords 加以处理。之所以只对 operation 部分解析,因为 operation 对应于服务具体的功能函数,其中包含函数名,传递参数,返回值以及功能描述[14]。

然后,计算 keywords 的权值。与前面 TF * IDF 有所不同的是,这里只计算 f_{ij} 值,且去除 id$f_{ij}=0$ 的 keywords(id$f_{ij}=0$ 说明该词在每一个服务中都出现过,无区分价值)。这样就可以建立一个矩阵 $\boldsymbol{M}$,用于下面匹配,m_{ij} 代表第 i 个 keyword 在第 j 个服务中的出现频率 f_{ij}。

最后,分别计算每一个 keyword 和矩阵 $\boldsymbol{M}$ 中每一项即每一个关键词的语义相似度。这里我们根据 WordNet 概念链上下位关系树中两个概念的最短路径长度,并引入 Wu - Palmer[11] 等提出的一种改进的最短路径距离作为语义相似度:

$$\text{Sim}_{wp}(k1,k2)=2\times\text{lenth}(r,k3)/\text{lenth}(k2,k3)+\text{lenth}(k2,k3)+2\times\text{lenth}(r,k3) \tag{7}$$

其中 r 为树的根节点,$k3$ 为概念 $k1$ 和 $k2$ 相同祖先节点中层次最低的一个概念,lenth($k2$,$k3$) 为 $k2$,$k3$ 在 WordNet 概念链上下位关系树中的最短路径长度,该值通过调用 WorldNet 提供的 API 可以得到。

一切工作做好之后,下面就可以让请求信息和服务进行匹配了。算法如下:

1. 获取 ReqKeywords (ReqKeywords 代表请求信息中提取并处理过的关键词集) 中每个词的最常用词性。

2. 依次判断 ReqKeywords 每一个 keywords 和 $\boldsymbol{M}$ 矩阵中每一个 keywords 是否属同一词性,如相同则计算它们的语义相似度,存储在映射 Similarity$<k,v>$ 中。(k 代表 $\boldsymbol{M}$ 矩阵中的每一 keywords,v 代表语义相似度 Sim_{wp}。

3. 找出每一个 Distances$<k,v>$ 中相似度值 v 最大的 keywords 将其映射存储在 MaxSimilarity$<k,v>$ 中。在这里我们可以设定一个门限 $d1$,当 MaxSimilarity$<k,v>$ 中 keywords 的数量 $<d1$ 时就认为无匹配服务。

4. 根据 $\boldsymbol{M}$ 矩阵找出 MaxSimilarity 中每一个 keyword k 对应的第 i 行,m_{ij} 非 0 的几个服务 j 将其存储在 result_k 中。设置一个门限 $d2$,在 n 个 result_k 中服务 j 如果出现的次数超过 $d2$,则将其存储在 LastResult 中,最后将 LastResult 存储的服务返回给用户。

3 实验结果与分析

实验分为两个步骤,第一步对服务进行分类并对结果进行分析,第二步通过一个具体例子在其中的某

一类服务中展示服务查询匹配过程。

3.1 服务的分类与结果分析

从 S. OH 所提供的 WSDL 文档集中选取了 40 个真实服务的 WSDL 文档,它们分为 4 类服务: Communication, CountryInformation, Converter, Mathematics。分别采用两种方法对其进行分类,传统的 VSM 方法和文章所提出的 Kernel - WordNet - VSM 方法。核函数选取的是 Polynomial 函数。利用三种度量方式对分类结果进行评估,分别是: R - precision ,top - n precision ,average precision。分类测试评估结果如图 2 所示。

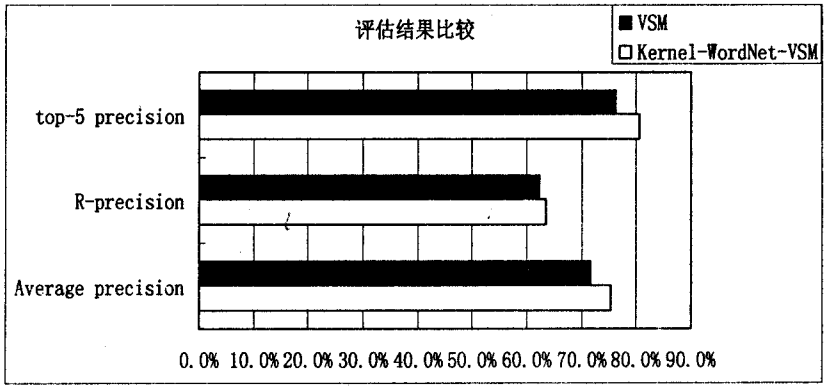

图 2 分类结果评估图

3.2 客户查询服务案例测试

分类完成后,现在,假设某客户希望选择澳洲几个城市旅游,需要了解澳洲各地的气温。通过前面的分类,客户甲选择在已分好的 country-Information 类型里查询。客户甲的请求信息为 "The temperature of capital in Australia." 州府城市属于澳洲比较有代表性城市且覆盖澳洲大部分地方。

在我们设计编写的程序中,通过分词及 WordNet 中 stopwords 处理,请求信息被提取成以下三个关键词: temperature, capital, Australia。但是在这个服务的 WSDL 文档中并没有出现 capital,只是有这些州府城市的名字如: Adelaide。接着, temperature, capital, Australia 这三个词和所构建的 M 矩阵中每一关键词按照公式(7)进行语义相似度计算,分别得到与这三个词相似度最大的所有关键词组成三组映射如图 3: MaxSimilarity$<k,v>$映射图(图中的数字为 WordNet 概念链中两个词的语义相似度)得到 MaxSimilarity $<k,v>$后,接着在 M 矩阵中查找到这 8 个关键词对应的服务。

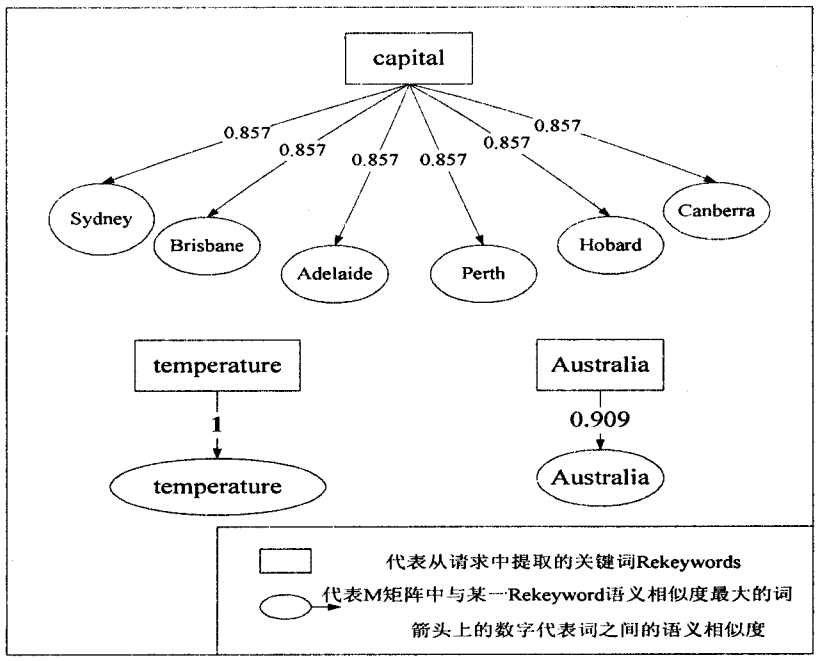

图 3 MinDistances$<k,v>$映射图

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13
Sydney	9	0	0	0	0	0	0	0	0	0	0	0	0
Brisbane	9	0	0	0	0	0	0	0	0	0	0	0	0
Adelaide	9	0	0	0	0	0	0	0	0	0	0	0	0
Perth	9	0	0	0	0	0	0	0	0	0	0	0	0
Hobart	9	0	0	0	0	0	0	0	0	0	0	0	0
Canberra	9	0	0	0	0	0	0	0	0	0	0	0	0
temperature	99	0	0	0	0	0	0	0	5	0	0	0	10
country	0	9	0	0	0	0	0	0	0	0	0	37	0

注: S1—S13 代表该类型的 13 个服务。m_{ij} 代表第 i 个 keyword 在第 j 个服务中的出现频率 f_{ij}。

图 4 M 矩阵中的 8 个关键词

在这里我们设置门限 $d2$ 为 5,在所得的 M 矩阵中,其项即关键词一共有 95 个,其中只有 8 个关键词与请求信息有关,则由图 4 所得行代表 8 个关键词,列代表该组的 13 个服务,则服务 $S1$ 在 8 个 result 中出现次数总和为 $7 > d2$,所以 $S1$ 为用户所需的服务。经 $S1$ 的客户端对其真实服务访问验证: S1 的确具有提供澳洲城市天气情况的服务功能,符合我们的最终结果。

4 结束语

鉴于目前 Web 服务发现机制两种主要研究方向的各自缺陷,提出了一种基于 WordNet 和 Kernel 方法的服务发现机制。该方法弥补了语法匹配缺少语义即

(下转第 76 页)

定任务的执行顺序，实现简单，在其可调度性界限内，可以保证任务的成功调度。具有运行开销小和能将频率高的任务顺利完成的特点。但是，RM 调度算法没有考虑到任务的等待时间，造成那些周期长但急需要执行的任务无法被调度。FCFS 算法仅用任务的等待时间来作为优先级的评价标准，这类算法容易实现，但效率不高，只顾及作业等候时间，没考虑作业所能承受等待时间的长短。文中针对这几种传统的只以任务的等待时间或者只以任务的周期来设置优先级，提出了一种根据任务的等待时间和周期来共同决定优先级的算法—剩余时间法，该算法根据任务的周期与等待时间的差值来动态修改任务的优先级。这样做既能考虑到任务的等待时间，也能考虑到任务所能容忍的等待时间的极限。并且在文章的后半部分，通过任务调度实验与 RM 算法进行比较，实验结果证明剩余时间算法能很好地提高任务的完成率及 CPU 的利用率。

参考文献：

[1] 董吉文，张 阳. 嵌入式实时操作系统任务调度算法的改进与应用[J]. 计算机应用，2009(9)：2516－2519.

[2] 陈文星，张辉宜，陶 陶，等. 嵌入式 Linux 的实时性改进技术[J]. 计算机技术与发展，2006，16(10)：114－117.

[3] 刘胜辉，马 嵩. 基于 Linux 内核的实时调度机制研究及应用[J]. 计算机工程与应用，2008，44(6)：121－123.

[4] 谢 敏，李桥梁. 嵌入式实时操作系统任务调度算法优化[J]. 电子科技，2005(12)：24－26.

[5] Gafford J. Rate monotonic scheduling[J]. IEEE Micro，1991，11(3)：34－39.

[6] 邢科群，郝红卫，温天江. 两种经典实时调度算法的研究与实现[J]. 计算机工程与设计，2006，27(1)：117－119.

[7] 赵俊锋. 一种基于 FCFS 调度策略的多处理机系统的仿真模型[J]. 宁夏大学学报：自然科学版，2001(4)：415－418.

[8] 洪雪玉，张 凌，袁 华. Linux 下的实时调度算法[J]. 华南理工大学学报：自然科学版，2008，36(4)：14－19.

[9] 王永吉，陈秋萍. 单调速率及其扩展算法的可调度性判定[J]. 软件学报，2004(6)：799－814.

[10] Obenza R. Rate monotonic analysis for real time systems[J]. IEEE Computer，1993，26(3)：73－74.

[11] Brandt S A，Banachowski S，Lin Caixue，et al. Dynamic integrated scheduling of hard real－time，soft real－time and non－real－time processes[C]//Proceedings of the 24th IEEE Real－Time Systems Symposium(RTSS 2003). [s. l.]：[s. n.]，2003.

[12] Liu C L，Layland J. Scheduling algorithms for multiprogramming in a hard real－time environment[J]. J. ACM，1973，20(1)：46－61.

(上接第 72 页)

智能化低的弱点，同时通过利用 WordNet 减轻了在语义领域中对于每一领域都需建立本体库的复杂工作。下一阶段工作将尝试引入 WordNet 概念链中除了上下位关系以外的多种关系，并在服务匹配过程中综合考虑服务的多种属性。

参考文献：

[1] 白东伟. 基于语义的 web 服务发现与技术研究[D]. 北京：北京邮电大学，2007.

[2] 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.

[3] Baeza－Yates R，Ribiero－Neto B. Modern Information Retrieval[M]. [s. l.]：Addison Wesley，1999.

[4] Stroulia E，Wang Y. Structural and semantic matching for assessing Web service similarity[J]. International jornal of cooperative information systems，2005，14(4)：407－438.

[5] Kokash N A comparison of Web service Interface similarity measures[C]//3rd European starting AI researcher symposium. Riva delGarda，Italy：IOS Press，2006：220－231.

[6] 张丽坤，蒋 波. 基于本体的语义 Web 研究[J]. 计算机技术与发展，2007，17(6)：116－119.

[7] Paolucci M，Kawamura T，Payne T，et al. Semantic Matching of Web services Capabilities[C]//Proceedings of the International Semantic Web Conference. Sardinia，Italy：[s. n.]，2002：333－347.

[8] 陈江锋，于建军. 基于扩展 VSM 的 Web 服务发现[J]. 计算机工程，2008，34(12)：25－27.

[9] Voorhees E. Using WordNet for Text Retrieval[C]//Fellbaum C. in WordNet：An Electronic Lexical Database 1998. The Cambridge，MA，：MIT Press，1999：285－303.

[10] 张 剑，李春平. 基于 WordNet 概念向量空间模型的文本分类[J]. 计算机工程与应用，2006，42(4)：174－179.

[11] Wu Zhibiao，Palmer M. Verb semanties and lexical selection[C]//Inproceedings of the 32nd Annual Meeting of the Association for Computational Linguistics. New Mexico：[s. n.]，1994：133－138.

[12] Vapnik V N. The Nature of Statistical Learning Theory[M]. New York：Springer－Verlag Inc，1995.

[13] W3C. Web Services Description Language，Version 1.1[EB/OL]. 2001. http://www.w3.org/tr/WSDL.

[14] 叶 蕾，张 斌. 基于功能语义的 web 服务发现方法[J]. 计算机研究与发展，2007，44(8)：1357－1364.