

异构环境下语义 Web 服务发现算法研究

甄景波^{1,2}, 王志晓², 王计林³

(1 中国矿业大学 计算机科学与技术学院, 江苏 徐州 221116;

2 神华宁夏煤业集团, 宁夏 银川 750001)

摘 要: 随着网络上越来越多种类各异的 Web 服务, 如何对其进行有效发现是实现动态 Web 服务组合的关键. 本文研究了异构环境下服务发现语义信息的匹配问题, 利用领域本体解决了异构数据的综合利用与共享, 并通过实验表明了本文提出方法的有效性.

关键词: 异构环境; 语义 Web; 服务发现; 算法

中图分类号: TP393

文献标识码: A

文章编号: 1000-7180(2012)03-0043-04

Semantic Web Services Discovery Algorithm Research in Heterogeneous Environment

ZHEN Jing-bo^{1,2}, WANG Zhi-xiao², WANG Ji-lin³

(1 School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China;

2 Shenhua Ningxia Coal Industry Group, Yinchuan 750001, China)

Abstract: As more and more different types of networks of Web services development, how to find them effectively is the key to implement the dynamic Web service composition. This paper studies the matching problem of service discovery semantic information in heterogeneous environment. By making use of domain ontology, this paper solves the comprehensive utilization and sharing of heterogeneous data, and experiments show that the proposed method this chapter discussed is effective.

Key words: heterogeneous environment; semantic web; service discovery; algorithm

1 引言

Web 服务发现是 Web 服务系统架构中的一个重要组成部分, 是 Web 服务发展的关键技术之一. Web 服务发现是指用户以某种方式在不同类型的 Web 服务中找到其需要的服务, 以执行 Web 服务请求^[1]. 现有的 Web 服务发现方法分为: 句法级服务发现和语义级服务发现^[2]. 句法级服务发现通过关键字匹配进行服务搜索, 它着重对服务的接口和实现细节进行定义, 但不支持对服务功能和行为的语义描述, 因此实现起来较为简单. 语义级服务发现能够有效的利用语义信息和领域本体, 实现 Web 服务的自动语义匹配和搜索, 具有较高的查准率.

当前的语义 Web 服务研究大多是基于单本体

或单一数据环境. 而在实际应用情况下, 不同的领域专家, 甚至是同一领域对于同一事物也可能采用不同的术语和方法, 因此就会产生多种领域本体. 此外, 当服务发现涉及到不同领域时, 服务的请求和发布有可能采用不同的方式来描述的, 因此进行服务发现时就会面临大量的异构服务, 利用异构环境语义信息匹配可以很好的解决上述问题. 本文在研究异构环境语义信息匹配的基础上, 提出了基于二分图的输入输出参数匹配算法, 并对算法进行实验分析.

2 语义 Web 服务描述和请求模型

Web 服务的描述模型是语义 Web 服务发现和组合的基础. 服务提供者对服务功能和用户交互的

方式进行设定,将 Web 服务的输入、输出参数、前提条件、执行效果及 QoS 等用语义信息即本体概念封装之后,以统一的接口形式对外公布,为服务的语义发现和发布奠定基础。

在综合比较文献[1-4]的基础上,本文将语义 Web 服务描述模型定义为元组 SWD。

$SWD = \langle SN, SD, IS, OS, PreD, EffD, QoS, O_d \rangle$

其中:SN 为语义 Web 服务的名称;

$SD = \{sd_1, sd_2, \dots, sd_n\}$ 为语义 Web 服务的描述, sd_n 为概念的集合;

$IS = \{is_1, is_2, \dots, is_n\}$ 为语义 Web 服务的输入, is_n 代表服务的各输入参数;

$OS = \{os_1, os_2, \dots, os_n\}$ 为语义 Web 服务的输出, os_n 代表服务的各输出参数;

$PreD = \{pred_1, pred_2, \dots, pred_n\}$ 为语义 Web 服务的前提条件集合, $pred_n$ 代表具体前提条件的参数;

$EffD = \{effd_1, effd_2, \dots, effd_n\}$ 为语义 Web 服务的效果集合, $effd_n$ 代表具体效果的参数;

$QoS = \langle Time, Cost, Reliability \rangle$ 为服务质量的非功能属性描述,包括调用服务所需的时间、费用、可靠性等;

O_d 为语义 Web 服务提供方的领域本体,语义 Web 服务描述模型中的概念均来源于该本体。

与描述模型类似,本文定义语义 Web 服务请求模型为元组 SWR。

$SWR = \langle RN, RD, IR, OR, PreR, EffR, RQoS, O_r \rangle$

其中,各变量含义与语义 Web 服务描述模型相似。

3 概念相关度、语义相似度的概念和计算方法

3.1 概念相关度及其计算方法

概念相关度指概念之间相关的程度。与相关性不同的是概念所表达的一些特征不直接重合。在文献[5]的基础上,本文在[0,1]区间内取一个实数值来表示度量的结果,该实数取数值 0(1)表示两个概念完全不相关(相关)。具体方法采用 Hirst-St-Onge 语义相关度算法^[6-7],其基本思想是:

当两个词在 WordNet 同义词集(Synset)中有一条较短的路径相连时,在语义上就具有较大的相关度。当此路径不存在时, $RelHS(w_1, w_2) = 0$ 。计算

公式为:

$$RelHS(w_1, w_2) = c - len(w_1, w_2) - k * turns(w_1, w_2) \quad (1)$$

其中, c 和 k 是两个常参数, $turns(w_1, w_2)$ 代表在 Synset 中的路径转向次数, $len(w_1, w_2)$ 是路径长度。

3.2 语义相似度及其计算方法

语义相似度指概念之间自身语义的相似程度,即从概念的含义来计算相似度。可见语义相关度和语义相似度成正比关系,本文通过 WordNet 语义词典来计算概念的语义相似度,其基本思想是:

两个单词通过上位关系连接的距离越近,它们的相似度就越大,反之它们的相似度越小。如果它们在有限上位层次中没有共同的父结点,则 $Simd(w_1, w_2) = 0$ 。参考文献[7]给出语义相似度计算公式为:

$$Simd(s_1, s_2) = 2 * \log P(s) / (\log P(s_1) + \log P(s_2)) \quad (2)$$

其中, $P(s) = \text{count}(s) / \text{total}$ 表示在 WordNet 中词义结点 s 及其子结点所包含的单词个数在整个词典中所占的比例, total 是 WordNet 的单词总数。

另外, $w_1 \in s_1, w_2 \in s_2$, 表示单词 w_1 和 w_2 分别位于结点 s_1 和 s_2 中, 结点 s 是 s_1 和 s_2 的公共祖先结点。令 $s(w_1) = \{s_{1i} | i = 1, 2, \dots, m\}$ 和 $s(w_2) = \{s_{1j} | j = 1, 2, \dots, n\}$ 分别表示单词 w_1 和 w_2 的所有词义, 则两个单词的相似度定义为它们之间语义相似度的最大值:

$$Simd(w_1, w_2) = \max(Simd(s_{1i}, s_{2j})) \quad (3)$$

$$s_{1i} \in s(w_1), s_{2j} \in s(w_2)$$

4 异构环境下语义 Web 服务发现算法

异构环境下语义 Web 服务发现算法的基本思想是:

(1) 根据概念相关度的定义,利用公式(1)对服务请求 SWR 中的 RN 与候选语义 Web 服务 SWD 中的 SN 进行概念相关度计算,筛选掉与服务请求无关的 Web 服务;

(2) 对候选语义 Web 服务进行前提条件和执行效果的匹配。如果概念相关度为 0 或前提条件和执行效果二者有其一不能匹配,则不再对候选服务进行语义相似度的计算,否则利用公式(3)对 SWR 与 SWD 进行 SD 与 RD 的语义相似度计算,并采用基于二分图的输入输出参数匹配算法 IOMatch 来衡量 SWR 与 SWD 的相似度;

(3) 利用 QoS 和 RQoS 对发现的结果进行排序,将最优结果排在最前。

4.1 基于二分图的输入输出参数匹配算法

SWR 与 SWD 的输入输出参数匹配的实质是:

- (1) SWR 与 SWD 的 or_n 与 os_m 之间进行匹配计算;
- (2) SWD 与 SWR 的 is_n 与 ir_m ($n \leq m$) 之间进行匹配计算。

通过上述步骤能够保证 SWD 能尽可能满足 SWR 的所有要求,而 SWR 也能提供 SWD 所需的输入。

输入输出参数匹配问题可以转化为二分图的最佳匹配问题^[1],即求一个集合 $X = \{x_1, x_2, \dots, x_n\}$ 到另一个集合 $Y = \{y_1, y_2, \dots, y_m\}$ ($n \leq m$) 的二分匹配,从而使得和式 $\sum sim(x, y)$ 能够取得最大值。 $sim(x, y)$ 是任一属于 X 的个体概念 x 与任一属于 Y 的个体概念 y 之间的语义相似度,满足 $0 \leq sim(x, y) \leq 1$ 。但因二分图的最佳匹配问题是针对 $|X| = |Y|$ (即 $n = m$) 的匹配,无法直接应用于 $n \leq m$ 的情况,因此需将二分图最佳匹配的定义进行扩充,即给定一个带权二分图 $G = (X, Y, E)$ 和 G 的一个匹配 M ,个体概念集合 X, Y 表示图的点, E 为 $\forall x \in X, \forall y \in Y$ 两点 $\langle x, y \rangle$ 所连的边集,并给该边一个权重 $W_{xy} = sim(x, y)$,二分图满足条件 $|X| \leq |Y|$, M 是 G 的一个最佳扩展匹配,当且仅当:

- (1) 若 $|X| = |Y|$, M 是 G 的最佳匹配;
- (2) 若 $|X| < |Y|$, M 是覆盖集合 X 中的所有节点,且权和最大的匹配。

基于 KM 算法^[8],本文提出 SWR 与 SWD 的输入输出参数匹配算法 IOMatch。

输入: $G = (X, Y, E)$ and $|X| \leq |Y|$

输出: G 的最优扩展匹配

If ($|X| = |Y|$) {

调用 KM 算法计算 G 的最优匹配 M ;

}

else {

利用如下计算将 $G(X, Y, E)$ 变换到 $G'(X', Y, E')$:

$X' = X \cup Z, Z = \{z_1, z_2, \dots, z_k\} (k = |Y| - |X|)$;
 $E' = E \cup \{\langle z_i, y_j \rangle \mid 1 \leq i \leq |Z|, 1 \leq j \leq |Y|, W_{z_i y_j} = 0\}$;

利用 KM 算法计算 G' 的最佳匹配 M' ;

从 M' 中去除 $E'' = \{\langle z_i, y_j \rangle \mid z_i \in Z\}$, 将 M' 变换为 M ;

}

return M .

4.2 前提条件/效果匹配及 Web 服务发现算法

在进行服务发现时,对于服务请求模型和服务描述模型的匹配,还需要考虑服务的前提条件(Pre-conditions)和服务的效果(Effects)的匹配。前提条件指的是服务要产生预期结果所需要满足的条件。与服务的输入参数不同,它是服务调用的约束条件。如果服务的前提条件不能被满足,Web 服务则不能够被正常调用。效果指的是调用服务所产生的结果。用 $PreR$ 和 $EffR$ 来分别表示服务请求模型中的前提条件和效果, $PreD$ 和 $EffD$ 分别表示服务描述模型中的前提条件和效果。由 2.1 节的介绍,知道 $PreD, PreR$ 均为若干简单条件的集合,分别表示为: $PreD = \{pred_1, pred_2, \dots, pred_n\}$ 和 $PreR = \{prer_1, prer_2, \dots, prer_m\}$ 。

$PreD$ 与 $PreR$ 是否匹配,必须满足:

对任何 $pred_i \in PreD$, 存在一个 $prer_i \in PreR$ 且 $pred_i \supseteq prer_i$, 或 $PreD$ 为空, Web 服务描述模型无任何前提约束条件。可用描述逻辑表述如下:

$MatchPre(PreD, PreR) \leftarrow ((\forall pred_i, \exists prer_j : (pred_i \in PreD)) \wedge (prer_j \in PreR)) \wedge Subs(pred_i \in prer_j) \vee (PreD = \Phi)$ 。

其中, $Subs(pred_i, prer_j)$ 表示当 $pred_i$ 包含 $prer_j$ 时, $Subs(pred_i, prer_j)$ 为真, 否则为假。服务请求和服务描述之间的条件匹配为:

$SimPre(PreR, PreD)$

$$= \begin{cases} 1, & MatchPre(PreR, PreD) \text{ 为真} \\ 0, & \text{其他} \end{cases} \quad (4)$$

与前提条件匹配类似,效果匹配表述如下:

$MatchEff(EffR, EffD) \leftarrow ((\forall eff_n, \exists eff_d : ((eff_n \in EffR)) \wedge (eff_d \in EffD)) \wedge Subs(eff_n, eff_d))$

服务请求与服务描述之间的效果匹配表示为:

$SimEff(EffR, EffD)$

$$= \begin{cases} 1, & MatchEff(EffR, EffD) \text{ 为真} \\ 0, & \text{其他} \end{cases} \quad (5)$$

在上一节 IOMatch 算法基础上,结合前提条件/效果匹配的计算方法,本文给出异构环境下 Web 服务发现算法 GradeFinder 如下:

输入: SWR

输出: 排过序的 $SWD_1, SWD_2, \dots, SWD_n$

for each SWD_n do {

If ($RelHS(RN, SN_n) = c - len(RN, SN_n) - k$

* $turns(RN, SN_n) = 0$)

```

return 0;
else If (SimPre(PreR, PreD)=0 or SimEfff
(EffR, EffD)=0)
return 0;
else{
    调用 IOMatch 算法计算二分图(OR, OS)的最大
    权匹配,并求出最大相似度和  $\sum sim(OR, OS)_i, i=1, 2, \dots, n;$ 
    调用 IOMatch 算法计算二分图(IS, IR)的最大
    权匹配,并求出最大相似度和  $\sum sim(IS, IR)_j, j=1, 2, \dots, n;$ 
    根据  $(RelHS(RN, SN) * \sum sim(OR, OS)_i +$ 
 $(1 - RelHS(RN, SN)) * \sum sim(IS, IR)_j)$  的值对
    SWDn 进行大小排序;
    再对 SWDn 按 QoS 的值进行大小排序;
    return 按大小排过序的 SWDn;
}

```

5 异构环境下语义 Web 服务发现算法验证及分析

为测试算法的有效性,本文在参照文献[2]的测试方法的基础上,以煤矿企业煤炭生产管理系统及矿山资源检索系统等煤矿领域 30 个常用语义 Web 服务,每个语义 Web 服务对应构建一个领域本体.测试用服务请求为按照 2.1 节模型要求设计的 10 个常用服务请求:查询煤炭种类和报价、查询出省口信息、查询煤炭产量信息、查询铁路运价里程信息、查询公路运费信息、查询水路运费信息、查询港口信息、查询仓储信息、查询煤炭销量历史信息、查询转运费用信息.在 Java 环境下,对本算法进行有效性验证.将算法匹配结果与专家人工匹配的结果进行比较,对这 10 个服务请求采用不同的方法所获得的查全率和查准率统计分别如图 1、图 2 所示.

考虑 I/O 相似度和 QoS 值的曲线代表分级匹配算法执行结果统计数据,其他两个曲线分别为算法执行过程中仅考虑某一方面功能而得到的统计数据.专家人工匹配的查全率和查准率均为 100%.总体来看,分级匹配算法的查全率和查准率要优于只考虑某一方面功能的效果.从图 1、图 2 的曲线对比可以看出,对于概念解析比较简单的查询(如查询 4、5、6、7、10),查全率和查准率均较高,对于其他的查询,由于对应领域本体库相关概念和 QoS 约束不

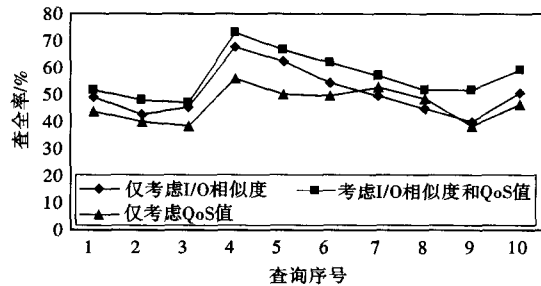


图 1 考虑不同因素时的查全率

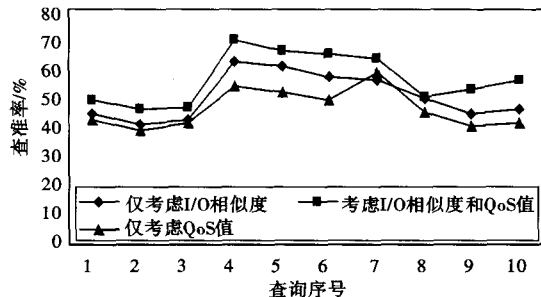


图 2 考虑不同因素时的查准率

同,查全率和查准率有时高,有时低一些.总之,实验说明分级匹配算法可行,并具备了较高的准确率.

6 结束语

本文主要针对异构环境下的语义 Web 服务发现问题,采用基于二分图匹配的发现策略.首先进行语义 Web 服务请求模型名称与语义 Web 服务描述模型的名称概念相关度的定义,并对前提条件和效果匹配情况进行计算,以筛选掉与服务请求无关的 Web 服务;其次进行语义 Web 服务描述与语义 Web 服务请求描述的语义相似度计算,并采用基于二分图的输入输出参数匹配算法来衡量二者的相似度,最后再对发现的结果进行排序,将最优结果排在最前面.利用领域本体解决了异构环境下数据的综合利用与共享,并通过实验表明了本文所提出方法的有效性.

参考文献:

- [1] 邓水光,尹建伟,李莹,等. 基于二分图匹配的语义 Web 服务发现方法[J]. 计算机学报, 2008, 31(8): 1364-1375.
- [2] 张磊,夏士雄,周勇,等. 基于多本体的语义 Web 服务发现研究[J]. 计算机工程与应用, 2009, 45(11): 165-171.
- [3] 刘明升. Web 服务动态组合的关键技术研究[D]. 大连:大连海事大学信息科学技术学院, 2008.

(下转第 50 页)

表3 包含2003年情况下春夏季所需引水量的
计算值和统计值

年份	统计值(单位: 10000 m ³)	计算值(单位: 10000 m ³)	绝对误差(单位: 10000 m ³)
2007-5-7	4991	6574	1583
2004-7-23	5690	6123	433
2005-6-17	3952	4578	626
2006-8-10	3610	2327	1283
2003-7-23	10927	9598	1329

根据以上对比分析可知,2003年的数据确实存在异常,在进行回归分析时,去除2003年数据是正确的处理;同时,根据引水季节的不同,把水质水量数据分成春夏季和秋冬季分别建立回归模型也是合理的.因此最终春夏季水质水量的关系为:

$$Y = 975X_1 + 390X_2 - 9493$$

秋冬季水质水量的关系为:

$$Y = 265X_1 + 233X_2$$

4 结束语

本文通过相关分析确定初始水质情况与达标时所需引水量之间的相关性,根据相关系数的变化情况,对异常数据进行了初步判断,并分析了季节变化对水质变化过程的影响,再根据相关分析的有关结论,分别建立不同情况下的水质水量回归模型,并通过回归模型的检验来确定最终的水质水量关系.本文方法取得了较好的结果.

在引水过程中,水质亦可能会受到城市工业污

染、大气降水等情况的影响,使水质的变化过程更为复杂.同时,由于数据的缺失,在某些年份水质随引水量的变化过程不能完全呈现.以上两点都可能会影响到水质水量模型的精度.如果能解决以上问题,则本文水质水量模型的精度将会得到进一步提高.

参考文献:

- [1] 吴浩云.引江济太调水实验关键技术研究 and 应用[J]. 中国水资源,2008(1):6-8.
- [2] 刘宁.对引江济太调水试验工程的初步认识和探讨[J].中国水利,2004(4):36-38.
- [3] 严剑波,阮晓红,孙瀚.多元回归分析在黄河水质预测中的应用[J].人民黄河,2010,32(3):35-36.
- [4] 李亦芳,程万里,刘建厅.基于人工神经网络和回归分析的水质预测[J].郑州大学学报,2008,29(1):106-109.
- [5] 徐业平.河流水质水量综合评价方法研究综述[J].水资源保护,2005,21(4):34-36.
- [6] 范金城,梅长林.数据分析[M].5版.北京:科学出版社,2006.
- [7] 汪嘉冈. SAS V8 基础教程[M].北京:中国统计出版社,2003.

作者简介:



朱跃龙 男,(1959—),博士,教授.研究方向为智能信息处理.

(上接第46页)

- [4] 李曼.基于语义的Web服务发现研究[D].长沙:中南大学信息科学与工程学院,2009.
- [5] 张忠平,田淑霞,刘洪强.一种综合的本体相似度计算方法[J].计算机科学,2008(12):142-145.
- [6] Budanitsky A, Hirst G. Evaluating word net-based measures of lexical semantic relatedness[C]// Association for Computational Linguistics. Sweden, 2005.
- [7] W3C. Simple object access protocol [EB/OL]. [2010-07-19]. <http://www.w3.org/TR/soap>.

- [8] Lovasz L, Plummer M. Matching theory [M]. Amsterdam: North-Holland, 1986.

作者简介:



甄景波 男,(1981—),硕士研究生.研究方向为语义网络服务发现.