一种快速高效的本体匹配方法

罗俊丽,黄广君,孙建国
LUO Jun-li,HUANG Guang-jun,SUN Jian-guo

河南科技大学 电子信息工程学院,河南 洛阳 471003
Electronic Information Engineering College,Henan University of Science & Technology,Luoyang,Henan 471003,China
E-mail:luojl0394@126.com

LUO Jun-li,HUANG Guang-jun,SUN Jian-guo.Efficient algorithm for ontology matching.Computer Engineering and Applications,2009,45(21):94-96.

Abstract: Ontology matching is one of the main methods to slove the problem of ontology heterogeneous.An efficient and precise similar messure is a pre-requisite of an ontology matching process.The paper affords an improved method to solve the complexity and impreciseness of the traditional similarity in ontology mapping.Firstly,the candidate matching set is established through classifying the ontology so as to reduce the amount of computation.Secondly,an integrated approach of based-instance computation, based-attribute computation,based-structure computation is designed to compute the concept similarity.Lastly,experimental results are given to demonstrate the effectiveness of the matching approach.

Key words: ontology matching;concept simlarity;classify;machine learning

摘 要:本体匹配是解决本体异构问题的主要方法之一,一个高效、精确的相似度计算方法是本体匹配的前提条件,针对目前本体匹配时计算复杂以及计算不精确的问题,提出了一种改进的本体匹配方法,该方法首先通过对本体库分类来确定候选匹配集,减少了相似度计算的工作量;进而根据本体的定义模型,从概念实例、概念属性和概念结构等方面来综合计算概念相似度,提高了相似度计算的精确度。实验表明该方法能在较少的时间复杂度上达到较好的匹配效果。

关键词:本体匹配;概念相似度;分类;机器学习
DOI:10.3778/j.issn.1002-8331.2009.21.027 **文章编号:**1002-8331(2009)21-0094-03 **文献标识码:**A **中图分类号:**TP301

1 引言

本体在人工智能、信息检索、Web 服务发现等领域中扮演着越来越重要的角色。随着本体应用的增多,如何解决本体间的互操作成为一个比较棘手的问题。本体匹配[1]能很好地解决本体的异构问题,本体匹配的核心内容是计算两个概念的相似度。因此,如何提高概念相似度计算的高效精确性就成了研究本体应用的关键技术之一。

目前计算两个本体 O_1 和 O_2 中概念的相似度时,本体中的每一对概念都被考虑在内,计算量很大。而有的两个概念根本就不相似,计算它们的相似度是没有必要的。

针对相似度的计算,业内已有不少研究。按本体的定义模型主要可分为以下 3 种[2]:

(1)基于概念定义的方法:它是指在进行相似度计算时主要参考本体中概念的名称、描述、约束等。如 COMA 系统所采用的匹配器主要利用了概念定义的模式信息。该方法仅仅利用概念自身的语义进行匹配,没有考虑属性和关系对概念的描述作用。

(2)基于概念实例的方法:该方法利用概念的实例作为计算概念间相似度的依据。典型的如华盛顿大学的 Glue 系统[3],它采用机器学习的方法来完成不同本体之间的匹配任务,但 Glue 系统没有考虑概念间属性的映射。基于实例的方法对于两个本体的实例集没有交集时就无能为力,而这种情况对于不同部门建立的本体是一种普遍现象。

(3)基于概念结构的方法:它是指在相似度计算时参考了概念间的层次结构,如结点关系(父结点、子结点)、语义邻居关系等。它一般不单独使用,而是和其他方法结合,用来提升整体的映射性能。

针对以上问题,提出了一种在分类基础上进行综合相似度计算的方法。对于本体 O_1 中的一个概念 A, 不是比较本体 O_2 中所有的概念,而是在本体 O_2 的一棵子树中进行比较;在计算概念相似度时分别基于概念实例、概念属性、概念结构来计算,然后进行相似度集成合并。

2 基本概念

为了更好地描述文章所提出的本体匹配方法,下面将给出本体及相似度计算的相关概念。

基金项目:教育部科学技术重点资助项目(Ministry of Education Science and Technology Funded Projects of China under Grant No.03081)。
作者简介:罗俊丽(1986-),女,研究生,主要研究领域为语义 Web,信息检索;黄广君(1963-),男,博士,副教授,主要研究领域为语义 Web,Web Service;孙建国(1976-),男,研究生,主要研究领域为语义 Web,本体应用。
收稿日期:2008-04-28 **修回日期:**2008-07-31

定义 1(本体) 关于本体的形式化定义很多,采用文献[4]提出的定义形式:$O=(C,I,P,Hc,R,A^0)$。其中,C 表示概念的集合;I 表示实例的集合;P 表示描述概念特征的属性集合;Hc 表示概念的层次关系;R 表示概念间的关系;A^0 表示本体公理的集合。

定义 2(本体匹配) 本体匹配领域的研究较多,各种定义、表达因研究方法不同而有较大差别,这里将本体匹配定义为两个本体概念之间的相似程度。

在进行本体匹配时常用的方法有计算概念名的编辑距离和两个节点间的基距离以及计算实例的概率分布等。编辑距离为字符串转换所需的最小数目的单元编辑操作,包括字符的插入、删除、替换及相邻字符的调换。其定义为:

$$Sim(s_1,s_2)=\max\{0,\frac{m_1-d}{m_1}\} \tag{1}$$

其中 d 是节点 n_1,n_2 的名称 s_1,s_2 间的编辑距离,m_l 为两字符串长度较短者的长度。

两个结点间的基距离定义为:

$$Dist(A,B)=1-\frac{2m}{n_1+n_2} \tag{2}$$

其中,n_1、n_2 分别表示结点 A 在本体 O_1、结点 B 在本体 O_2 中的词的个数,m 为其中重叠的词的个数。

根据 Jaccard 系数,当基于实例计算两个概念的相似度时,可将其表示为:

$$Sim(A,B)=\frac{P(A\cap B)}{P(A\cup B)}=\frac{P(A,B)}{P(A,B)+P(\overline{A},B)+P(A,\overline{B})} \tag{3}$$

其中,$P(A,B)$表示一个实例即属于概念 A 又属于概念 B 的概率,$P(A,\overline{B})$表示一个实例属于概念 A 而不属于概念 B 的概率,$P(\overline{A},B)$表示一个实例不属于概念 A 而属于概念 B 的概率。

3　基于分类的本体匹配方法

为了解决概念相似度计算复杂度和精确性的问题,提出了基于分类的本体匹配方法。这种方法的基本思想是:根据本体定义中良好的概念层次结构,通过分类将本体库划分为若干小型的本体树,对于分类树的根节点(也称为本体库的分类节点)建立相似映射,这样所有的相似度计算均限制在建立了相似映射的分类树根节点所在的分类树之间,从而大大降低了概念相似度计算的复杂度。然后用提出的综合的相似度方法来计算分类子树中概念节点的相似度。

3.1　候选匹配集的选择

对于本体 O_1 中的一个概念 A,本体 O_2 中一般只有部分概念与它基本相似。因此,根据所划分的分类本体树(如图 1,有分类节点和叶子节点),先过滤出本体 O_2 种最相关的概念,产生一组候选概念集。通过对概念对的数量进行限制,可以降低相似度计算的复杂度。

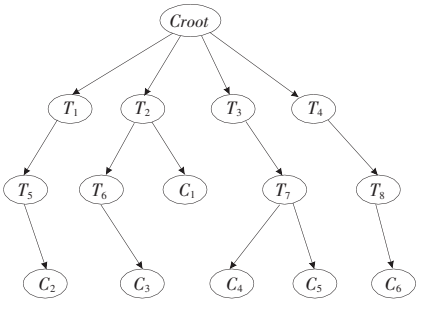

图 1　本体分类树

假定 C_i 和 C_j 为本体概念,且 $C_i\in O_1$ 和 $C_j\in O_2$。T_i 和 T_j 也为本体概念,且 $T_i\in O_1$ 和 $T_j\in O_2$,但 T_i 和 T_j 不为叶子节点概念。将 $tree_i=\{\forall C_i|C_i\subset T_i\}$称作以 T_i 为根节点的分类树 $tree_i$($tree_j$ 同理),T_i 和 T_j 称为本体库的分类节点。

(1)通常,如果两个本体库 O_1 和 O_2 中,两个分类节点是相似的,即 $\mathrm{sim}(T_i,T_j)>t$(t 为相似度阈值),可以认为分类树 $tree_i$ 和 $tree_j$ 所包含的所有节点之间是可能匹配的,从而减小了相似度计算的范围。

在计算分类根节点的匹配关系时,采用统计技术和概念词典相结合的方法来计算元素名称相似度。这种方法不仅在元素名称完全或部分相同的情况下有效,而且在名称完全不同但存在一定语义联系的情况下也非常有效。基于统计的方法有公式(1)给出,将其记为 Sim_{ed}。

文中参考 WordNet 语义词典来计算两个词语的语义相似度,并采用基于 Leacock Chodorow 的语义相似度可以表示为:

$$Sim_{dx}(s_1,s_2)=-\log\frac{len(s_1,s_2)}{2\cdot Depth}$$

其中,$Depth$ 为分类树的高度,$Len(s_1,s_2)$为两个词在语义词典中的最短路径长度。定义最终的相似度为:

$$Sim(s_1,s_2)=\frac{Sim_{ed}(s_1,s_2)+Sim_{dx}(s_1,s_2)}{2}$$

当 s_1、s_2 的相似度大于阈值时,它们之间是相似的。

(2)得到了相似的分类节点之后,就要对分类节点所引导的分类树包含的概念节点$\{sim(C_i,C_j)|C_i\in tree_i,C_j\in tree_j\}$进行相似度计算。计算方法采用下文所提出的综合相似度计算方法。

3.2　综合的概念相似度计算

概念相似度是指概念间自身语义的相似程度,两个概念所共同拥有的实例的数目以及相同的属性个数对概念的相似度有重要的影响,而本体的层次结构反映了概念之间关系,蕴含了大量的潜在语义。因此,从概念的实例、属性和结构 3 个方面来综合计算概念的相似度。

3.2.1　实例相似度计算

基于实例计算概念相似度的理论依据是:两个概念所拥有的相同实例的比重决定这两个概念的相似程度。

用公式(3)来计算基于实例的概念相似度,这样问题被简化成确定每个实例是否属于 $A\cap B$ 了。由于概念 A 的实例和概念 B 的实例是在两个本体框架下独立输入的,因此采用了朴素贝叶斯学习分类器来解决这个问题。以计算 $P(A,B)$为例,具体步骤为:

首先,分别以 O_1 中属于 A 的实例集和不属于 A 的实例集为正反训练样本,通过 Naïve Bayes 方法为概念 A 训练一个分类器。

然后,使用该分类器对 O_2 中的实例进行分类。假定 U_1 为 O_1 中的实例集合,U_2 为 O_2 中的实例集合,$N(U_1)$为 U_1 中实例的数目。$N_1(A,B)$为分类正确的 O_2 的实例的数目。同样可以以 O_2 中属于 B 的实例集为正例,O_2 中的其他实例为反例,通过 Naïve Bayes 方法为概念 B 训练一个分类器,则可以得到 $N_2(A,B)$为使用该分类器对 O_1 的实例进行分类所得的正例的数目。则:

$$P(A,B)=\frac{N_1(A,B)+N_2(A,B)}{N(U_1)+N(U_2)}$$

采用同样的步骤方法计算 $P(\bar{A},B)$,$P(A,\bar{B})$,可得:

$$P(\bar{A},B)=\frac{N_1(\bar{A},B)+N_2(\bar{A},B)}{N(U_1)+N(U_2)}$$

$$P(A,\bar{B})=\frac{N_1(A,\bar{B})+N_2(A,\bar{B})}{N(U_1)+N(U_2)}$$

将其代入公式(3)就得到了概念的实例相似度 $ISim(A,B)$。

3.2.2 属性相似度计算

在现实世界中,如果两个事物有很多属性相同,则说明这两个事物很相似,反之则相反。因此基于属性计算概念相似度的基本原理是:通过判断两个概念对应的属性集的相似程度来计算概念相似度。

本体中概念的属性按取值类型分为:Numeric、Date、Text和实例型。对于 Numeric 和 Date 型的属性,需要统计和计算属性值的最大值、最小值、平均值和值的分布状况来进行相似度匹配。实例型属性的值是一个实例,需要对这个实例文本化,然后将其作为 Text 型的属性来处理。

由于概念的实例是对该概念的每一个属性都分配了一个相应的值,因此对于 Text 类型的数据,可以采用基于实例的方法进行计算。

假定 A 是本体 O_1 中的概念,B 是本体 O_2 中的概念,如果 a_i 是 A 的属性,通过类型检查可以找到 B 中可能与之相等的属性 $b_j(j=1,\cdots,m)$。根据公式(3)a_i 和 b_j 的相似度可以用以下公式计算:

$$ASim(a_i,b_j)=\frac{P(a_i\cap b_j)}{P(a_i\cup b_j)}=\frac{P(a_i,b_j)}{P(a_i,\bar{b_j})+P(a_i,b_j)+P(\bar{a_i},b_j)}$$

这样,问题就被转化为计算 4 个联合概率,同样采用机器学习的方法来计算,只不过这里所用到的实例是这些属性的取值,可以得到:

$$P(a_i,b_j)=\frac{N_1(a_i,b_j)+N_2(a_i,b_j)}{N(U_1)+N(U_2)}$$

$$P(a_i,\bar{b_j})=\frac{N_1(a_i,\bar{b_j})+N_2(a_i,\bar{b_j})}{N(U_1)+N(U_2)}$$

$$P(\bar{a_i},b_j)=\frac{N_1(\bar{a_i},b_j)+N_2(\bar{a_i},b_j)}{N(U_1)+N(U_2)}$$

设概念 A 和概念 B 之间共计算出 m 个 $ASim(a_i,b_j)$,并设置相应的权值 $w^k_{attribute}$,则概念 A 和概念 B 基于属性的相似度计算公式为:

$$ASim(A,B)=\frac{\sum_{k=1}^{m} w^k_{attribute}ASim(a_i,b_j)}{\sum_{k=1}^{m} w^k_{attribute}}$$

3.2.3 结构相似度计算

结构特征的相似度计算是匹配的关键技术。如果两个元素的子概念和父概念存在映射关系的话,那么这两个元素也往往存在映射关系;如果两个给定元素有相同或相似的上下文结构,则这两个元素也可能存在映射关系。采用基于距离的语义相似度模型来计算结构相似度,则节点 A 和节点 B 之间的距离满足:

$$D(A,B)=c^p\cdot dist(A,B)+c^i\cdot dist(A,B)+c^c\cdot dist(A,B)$$

其中,$c^p+c^i+c^c=1$,$dist(A,B)$为两个结点间的基距离(basic distance)。则:

$$SSim(A,B)=\frac{\alpha}{D(A,B)+\alpha},\alpha\text{ 为可调节参数}$$

3.2.4 综合的概念相似度

把基于实例、基于属性和基于结构的概念相似度合并,得到综合的概念相似度,计算公式为:

$$Sim(A,B)=w^i\cdot ISim(A,B)+w^a\cdot ASim(A,B)+w^s\cdot SSim(A,B)$$

其中,$w^i+w^a+w^s=1$。

4 实验与分析

4.1 实验数据

采用 2007 年国际本体匹配大赛[5]所用的数据集。该数据集包括 54 组本体数据,描述了书籍信息。编号 101 的数据是它的参考本体,也是最完整的本体,其中包括 33 个概念,40 个属性,24 个关系。编号 223 的数据体现了本体的扩展层次关系,考虑到相似度计算是在本体分类树的基础上进行的,选取 101 和 223 两个本体来进行匹配。

4.2 评估标准

采用 TempLoadRunner 工具来测试算法是否减少了计算量,用信息检索领域里常用的查全率和查准率来验证算法的精确度。

$$\text{查全率 Recall}=\frac{\text{正确匹配个数}}{\text{参考匹配结果数量}}$$

$$\text{查准率 Precision}=\frac{\text{正确匹配个数}}{\text{实际总匹配数量}}$$

4.3 实验设计及结果

为了验证基于分类的本体匹配方法的高效性和精确性,在数据集上进行了两组对比实验,都采用综合相似度计算方法,第一组实验未进行分类,对于每一对概念都要进行相似度的计算;第二组实验采用了分类的方法,由于不同的分类方法对时间复杂度有一定的影响,因此在实验中分别选择了子树中含有 5~7 个、10~12 个、15~17 个概念节点来对比时间复杂度。其结果如表 1 所示:

表 1 执行时间比较

实验方法	未分类	分类法		
		5~7 个节点	10~12 个节点	15~17 个节点
执行时间/s	26	21	18	20

从比较结果可以看出,当概念子树中包含 10~12 个概念节点时有较少的执行时间,因此采用这种分类方法,分别比较了综合概念相似度计算方法与基于概念实例、基于概念定义、基于概念结构方法的查全率和查准率。结果如图 2:

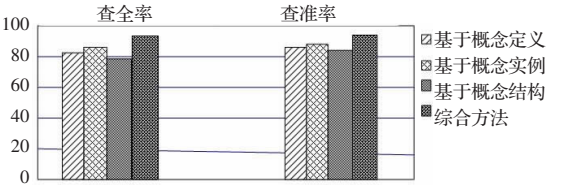

图 2 实验结果比较

从实验结果可以得出:综合的相似度计算方法具有较高的查全率和查准率,其次分别为基于概念实例的方法、基于概念定义的方法和基于概念结构的方法。

4.4 实验分析

从实验结果可以看出:相对于传统的算法,该算法具有较

(下转 192 页)