第35卷 第13期 计 算 机 工 程 2009年7月
Vol.35 No.13 Computer Engineering July 2009
·软件技术与数据库· 文章编号：1000—3428(2009)13—0066—03 文献标识码：A 中图分类号：TP393

一种基于语义网的本体映射改进算法

李 熙[1,2]，徐德智[2]，王建新[2]
(1. 永州职业技术学院信息中心，永州 425006；2. 中南大学信息科学与工程学院，长沙 410083)

摘 要：针对目前基于语义网的本体映射算法中背景本体搜索面少、本体收集不精确的问题，利用基于虚拟文档的映射技术提取在WordNet中与概念同义的同义词集，将对单个概念进行搜索转换成对同义概念集进行搜索，从而扩大本体搜索面，获取更多背景本体。提出基于语义环境的动态本体映射算法来排除错误背景本体，使本体收集更加精确。实验结果表明，该算法可有效提高映射的查全率和查准率。
关键词：本体；映射；语义网

Improved Algorithm of Ontology Mapping Based on Semantic Web

LI Xi[1,2], XU De-zhi[2], WANG Jian-xin[2]
(1. Information Center, Yongzhou Vocational Technical College, Yongzhou 425006;
2. School of Information Science and Engineering, Central South University, Changsha 410083)

【Abstract】This paper analyzes two existing problems in ontology matching algorithm based on semantic Web. It presents a new algorithm, which uses ontology matching based on virtual document to retrieval synset in the WordNet, and changes a single concept used by ontology searching into a concept set, so it can enlarge searching scope and obtain more background ontologies. It uses dynamic ontology matching based on semantic environment to exclude wrong background ontologies, so it can get more precision in ontologies collecting. Experimental result shows precision and recall of matching improved effectively by using the improved algorithm.
【Key words】ontology; mapping; semantic Web

基于语义网的本体映射方法[1-2]采用本体搜索引擎在网络中对包含待匹配概念对的本体文件进行搜索，收集这些本体文件，然后对这些本体文件进行语义挖掘，最后输出待匹配概念对的映射关系。该方法的主要思想是重用现有网络本体资源，在能描述语义关系的本体语言编写的本体中挖掘语义关系。这种基于语义的本体映射方法比传统的基于语言和结构的映射方法在映射结果上有更高的可信度，而且在词汇及结构不相似的情况下比传统的映射方法有更好的效果。然而该方法却存在某些不足，本文提出一种改进的基于语义网的本体映射算法。

1 基于语义网的本体映射方法及其不足

基于语义网本体映射的主要的思想是：通过本体搜索引擎动态的选取多个本体，利用推理工具挖掘单个本体或多个本体中所隐含待匹配概念对的语义关系。这种方法比基于人工选定背景本体映射方法有着可重用资源更加丰富的优点，因此，该方法具有更高的映射查全率。

基于语义网的本体映射的过程见图1。其中，概念 A, B 是待匹配概念对。

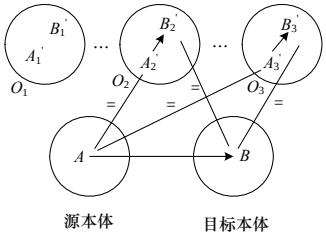

图1 基于语义网的本体映射过程

首先须在网络中搜集包含与待匹配概念对相同名称的 A', B'的本体(这个过程也称为定位过程)。然后在搜集后的本体中使用推理工具挖掘出 A, B 的语义关系。

然而在上述定位过程中却存在2个主要的问题：(1)由于未充分利用现有的网络本体资源而无法搜索到相关联的定位本体。(2)由于目标本体与搜索到的背景本体的语义环境不同而引起的本体收集不精确。

2 改进的本体搜索算法

为提高本体搜索面，在原先对于某个概念无法搜索到关联本体的情况下，通过同义词替换搜索来找到相关联的本体。本算法采用WordNet语义字典进行同义词提取。字典根据语义关系组织成一棵树的形式，其中树中的一个节点由一些同义词组成。一个词可能有多种意思，一个词可能在树中的多个节点上出现，因此，必须考虑本体中的概念词汇定位在WordNet语义树中的哪个节点上。这种定位过程本文通过映射来实现，将本体概念与WordNet存在映射关系的节点作为同义词集。

2.1 虚拟文档的建立

虚拟文档[3]是为了描述概念特点而建立起来的文档，每个概念都可建造自身的虚拟文档。虚拟文档主要包括本节点的描述信息和邻近节点的描述信息组成，由于本体和

基金项目：国家自然科学基金资助重点项目(60433020)
作者简介：李 熙(1975－)，男，讲师、硕士研究生，主研方向：语义网，本体映射，网络安全；徐德智，教授；王建新，教授、博士生导师
收稿日期：2008-11-04 **E-mail：**netmail123@163.com

WordNet[4]建模的不同，因此所提供的描述信息种类也不同。

定义 1(本体概念描述信息) 其描述信息与一些和概念有关的评论、注释、标签、属性组成定义为

$$Des(e)=\alpha_1\times CWC(e)+\alpha_2\times CAC(e)+\alpha_3\times CLC(e)+\alpha_4\times CPC(e)$$

其中，$\alpha_1, \alpha_2, \alpha_3, \alpha_4$ 是 4 个代表权值的有理数；$CWC(e), CAC(e), CLC(e), CPC(e)$分别代表概念 e 的评论、注释、标签和属性的词汇集合。值得注意是，这里的描述信息未包括概念名称，这主要是映射的目的是排除同名异义，概念的名称对这种映射目的并没有帮助。

定义 2(WordNet 概念描述信息) 其描述信息是与实体有关的评论和注释组成，定义为

$$Des(e)=\alpha_1\times CWC(e) + \alpha_2\times CAC(e)$$

其中，α_1, α_2 是 2 个代表权值的有理数。

定义 3(虚拟文档) 虚拟文档由实体的描述信息和实体子父概念集节点组成，定义为

$$VD(e) = Des(e)+r_1\times \sum_{e'\in FN(e)} Des(e')+r_2\times \sum_{e'\in CN(e)} Des(e')$$

其中，$FN(e)$代表关于实体 e 的父概念集合；$CN(e)$代表关于实体 e 的子概念集合；r_1, r_2 是 2 个有理数权值。

2.2 相似度计算

虚拟文档建立后，文档中的每一项是由唯一词汇所对应的权值组成，权值表示对应词汇和文档的联系程度。为能更精确地表达词汇和文档的关系，还须进行停用词删除，这些广泛使用的词汇不能代表文档的特性，只会对后续的相似度计算提供噪音信息。然后将文档变换成向量的形式，向量的长度由双方的虚拟文档唯一词汇数决定，最后通过余弦函数来计算相似度。

2.3 映射发现

相似度计算产生了本体中的一个概念与 WordNet 中多个节点的多个相对应相似度值，选取相似度值大于阈值且在所有相似度中最大的作为映射对：

$$Sim(A,\ A_j) = \max \sum_{k=1,2,\cdots,n} sim(A,\ A_k) \tag{2}$$

即选择满足式(2)的$(A,\ A_j)$作为映射对。其中，A 是本体的概念；A_j 是 WordNet 中的第 j 个相对应节点。

3 改进的本体收集算法

在本体搜索后，将多个背景本体存放在本地网络上，然而如果对这些本体不加以处理，直接进行概念间语义挖掘，必然会由于词义的模糊性收集了错误的背景本体，最终导致挖掘出错误的映射关系。因此，必须对收集后的本体进行筛选，排除错误背景本体。排除方法的基本思想与第 2 节相似，但这里采用了基于语义环境的动态本体映射的方法是因为映射对象是 2 个规模较小的本体，时间效率不是其主要的瓶颈，而且本体与本体之间映射比本体与 WordNet 之间映射可利用的信息更加全面(如属性信息、实例信息)。

3.1 基于语义环境的动态本体映射

基于语义环境的动态本体映射的主要思想是：相似度计算公式的确定和权值的设置是一个动态的过程，它不仅要考虑各个策略的自身特征，还须考虑各个策略所处的语义环境。例如注释信息的丰富程度对基于注释策略的影响，在名称相同情况下对策略组合方式以及策略自身变化的影响等。该映射方法比传统的映射方法能更有效的利用语义信息，特别是针对在对排除同名异义为主要映射目的的映射上有更好的效果。下文介绍几种具体策略，为简单描述起见，假设$<A, A'>$为待匹配概念对。

3.1.1 基于名称的策略

该策略采用 lenvenshtein's 编辑距离来计算概念间的相似度。

3.1.2 基于实例的策略

在概念同名的情况下，实例是判断 2 个概念是否相关最直接最有效的依据。因为实例是概念的实例化结果，概念的属性都包含在实例中，所以如果存在 2 个相同的实例，则说明 2 个概念也有许多属性相同，这也说明两者概念是有一定的相关性的。在现实中也很难找出 2 个同名异义的概念，会有相同的实例。在概念同名的情况下基于该策略的相似度计算公式为

$$Sim_{ins}(A,\ A') = \begin{cases} 1 & Ins(A)\cap Ins(A')\notin\varnothing \\ 0 & Ins(A)\cap Ins(A')\in\varnothing \end{cases}$$

其中，$Ins(A), Ins(A')$分别为元素 A, A'的实例集合。

在概念不同名的情况下，传统的方法是通过计算 2 个概念的相同实例集与共有实例集的比值来作为相似度值。然而这种方法却忽略了实例信息之间的差异，一般来说实例越丰富，语义信息越多，基于该策略的相似度计算也就越重要、越可信。给出如下定义：

定义 4(丰富度) 丰富度公式为

$$richness(A) = k\times|Ins(A)|,\quad k\in Q^+$$

其中，$Ins(A)$表示概念 A 所对应的实例集；$|Ins(A)|$表示实例集中实例的数目。丰富度是实例数目的正比例函数，$|Ins(A)|$越大，基于实例的匹配策略所得到的结论越重要、越可信。

经过调整后的在概念不同名的情况下基于实例策略的相似度计算公式为

$$Sim_{ann}(A,\ A') = richness(A)\times richness(A')\times\frac{Ins(A)\cap Ins(A')}{Ins(A)\cup Ins(A')}$$

3.1.3 基于注释的策略

该策略将注释的词汇看成实例，所以该策略与概念不同名的情况下的基于实例策略类似，也引用了实例丰富度。然而除了上述改进外，还须对注释信息在概念同名与不同名情况下进行不同的处理。在概念不同名的情况下须对注释进行停用词删除，而在概念同名的情况下除了删除停用词外还须删除与概念相同的词汇。因为这些词汇都是基于该策略的噪音信息，影响基于该策略相似度计算的正确性。

3.1.4 基于结构的策略

该策略的主要思想是：如果给定 2 个元素的有相同或相似的上下文结构，则可能存在映射关系。例如：2 个元素的父类和子类都存在映射关系，那么这 2 个元素也往往存在映射关系。结构的上下文包括父类、子类、属性和关系等。目前，在程序中只实现了直接与给定元素相连接的元素为上下文元素。

3.1.5 多策略合并

须将多策略的合并分为概念同名与不同名 2 种情况分析：在概念同名的情况下，若能判定概念间相关，则可判定概念是同义的，相似度值定为 1，否则说明出现同名异义的情况。判断 2 个概念是否相关，一种有效的方法就是判断 2 个概念的实例集中是否有相同的实例，如果有则说明相关。另一种方法是通过基于注释和结构的相似度值取为 CL_{sim} 这个数值反映了概念间的相关性，如果该数值大于给定阈值的话，则可认为是相关的；在概念同名且不相关的情况下说明出现同名异义，因此，基于概念名称的策略不能采用，其相

似度的值即为 CL_{sim}。固在概念同名的情况下相似度合并公式为其合并公式为

$$Sim(A,\ A^{'})=\begin{cases}1 & I_A\cap I_{A'}\neq\varnothing \text{ or } CL_{\text{sim}}\geqslant Lim_{sim}\\ CL_{sim} & CL_{sim}<Lim_{sim}\end{cases}$$

其中，I_A 表示概念 A 的实例集和；Lim_{sim} 是一个给定的阈值。

$$CL_{sim}=\sum_{k=1,2}w_k\sigma(Sim_k(A,\ A^{'}))/\sum_{k=1,2}w_k$$

其中，W_k 是某个策略的权值；σ 是一个 sigmoid 函数，sigmoid 是一个平滑函数，它使合并结果偏向于预测值高的策略。

在概念不同名的情况下，则须通过基于名称、实例、结构、注释相结合来计算相似度。其合并公式为

$$Sim(A,\ A^{'})=\sum_{k=1,2,\cdots,n}w_k\sigma(Sim_k(A,\ A^{'}))/\sum_{k=1,2,\cdots,n}w_k$$

3.2 错误背景本体排除

通过基于语义环境的动态本体映射的相似度计算可知在概念对同义的情况下，其相似度值为 1。错误背景本体排除的过程如下:

(1)提取与待匹配概念对<A, B>在背景本体中对应的概念对<A', B'>;

(2)分别对<A, A'>, <B, B'>进行基于语义环境的动态本体映射的相似度计算，获取 $Sim_{bs}(A, A')$, $Sim_{bs}(B, B')$;

(3)排除不满足 $Sim_{bs}(A,\ A')=Sim_{bs}(B,\ B')=1$ 此条件的错误背景本体。

4 改进的基于语义网的本体映射

改进后的映射系统的基本步骤如下:

输入 2 个待匹配本体

输出 本体中的概念对的匹配关系

(1)在本体中提取待匹配概念对。

(2)使用本体搜索引擎在对包含待匹配概念对的本体进行搜索。

(3)对搜索后的背景本体进行筛选，排除错误误景本体。

(4)计算经过筛选后的本体文件数目，如果 $O_{num}\leqslant O_{\min}$ 进入(4)，否则进入(5)。其中，O_{num} 是筛选后的背景本体文件数目；$O_{\min}$ 是一个固定的整数阈值。

(5)对概念进行同义词替换后，进入(2)。如果同义词替换结束，进入(1)。

(6)通过推理工具对筛选后的背景本体提取待匹配概念对的语义关系。

5 实验结果及分析

5.1 实验数据

考虑到基于语义网本体映射时间效率较低，在没有影响到结果的正确性基础上，本文的实验选用了规模较小的 university 数据集，该数据集分别包括 SWRC 和 LUBC，描述的是大学本体。其中，SWRC 含有 56 个概念；LUBC 含有 43 个概念。

本文采用查全率和查准率作为评价标准对实验结果进行评估，手工建立测试数据集的映射关系，将其作为测评标准。

5.2 实验结果及分析

本文先针对改进前的算法进行实验，查全率和查准率分别达到 56%和 78%；然后针对改进后的算法，Lim_{sim} 不同时其查全率和查准率见表 1。

表 1 改进后算法的的查全率和查准率

	Lim_{sim}				
	0.24	0.28	0.32	0.36	0.40
查全率/(%)	81	85	91	97	99
查准率/(%)	73	62	57	32	15

可见，当 Lim_{sim} 较低时，改进后的映射算法能有效提高查全率。当 Lim_{sim}=0.24 时，查全率达到 73%，比改进前上升了 17%，然而随着 Lim_{sim} 值的不断增加，查全率下降明显，这主要由于 Lim_{sim} 在偏高的情况下，错误地排除了正确的映射对。在查准率上，当 Lim_{sim} 增大时，查准率提高较为明显。

6 结束语

本文主要针对现有基于语义网本体映射过程中的背景本体搜索面窄和收集精度不高等问题，提出基于虚拟文档的本体映射技术提取 WordNet 同义词集，提高搜索面；通过基于语义环境的动态本体映射对收集后的本体进行筛选，提高收集精度。实验分析结果表明，改进后的算法在映射效率上优于原先的算法。下一步研究工作是降低改进后基于语义网本体映射算法的时间复杂度，并将基于语义网本体映射算法与传统的映射方法相结合，提高映射效果。

参考文献

[1] Aleksovski Z, Klein M, Kate W T. Matching Unstructured Vocabularies Using a Backgound Ontology[C]//Proc. of EKAW'06. Podebrady, Czech Republic: [s. n.], 2006.

[2] Gracial J, Lopez V, Aquin M, et al. Solving Semantic Ambiguity to Improve Semantic Web Based Ontology Matching[C]//Proc. of OAEI'07. Busan, Korea: [s. n.], 2007.

[3] Qu Yuzhong, Hu Wei, Cheng Gong. Constructing Virtual Documents for Ontology Matching[C]//Proceedings of the 15th International Conference on World Wide Web. Edinburgh, Scotland: [s. n.], 2006.

[4] Fellbaum C. WordNet——An Electronic Lexical Database[M]. Cambridge, MA, USA: MIT Press, 1998.

编辑 金胡考

（上接第 22 页）

参考文献

[1] 王子健, 张 军, 罗喜伶. 基于 TFFS 的嵌入式系统在线升级设计与实现[J]. 计算机工程, 2006, 32(13): 257-259.

[2] WindRiver Inc.. VxWorks Programmer's Guide5.5[EB/OL]. (2002-01-01). http:// www.windriver.com.

[3] Felser M, Kapitza R, Kleinöder J, et al. Dynamic Software Update of Resource-constrained Distributed Embedded Systems[DB/OL]. (2007-07-04). http://www4.infomatik.uni-erlangen.de/publications/2007/felser_07_iess.pdf.

[4] Levine J R. Linkers and Loaders[M]. San Francisco, USA: Morgan-Kauffman, 1999.

[5] 程敬原. VxWorks 软件开发项目实例完全解析[M]. 北京: 中国电力出版社, 2005.

编辑 顾姣健