第 35 卷 第 4 期 计 算 机 工 程 2009 年 2 月
Vol.35 No.4 Computer Engineering February 2009
·软件技术与数据库· 文章编号：1000—3428(2009)04—0075—03 文献标识码：A 中图分类号：TP301

大本体的分块与映射方法研究

徐德智，陶 克
(中南大学信息科学与工程学院，长沙 410083)

摘 要：在本体的映射研究中，现实本体或大本体之间的映射算法是研究的难点。该文提出一种针对大层次本体的映射方法。根据本体的结构和概念之间的语义距离，应用向量空间模式(VSM)将概念表示成多维空间中的点。在此基础上，应用聚类算法(CURD)对概念进行聚类，形成若干个语义上相对独立的块，在 2 个本体的块之间根据参考点建立映射关系。实验结果表明，该方法在测试数据集上能得到较好的映射结果

关键词：大本体；聚类；块；映射

Research on Large-scale Ontology Partition and Mapping Method

XU De-zhi, TAO Ke
(College of Information Science and Engineering, Central South University, Changsha 410083)

【Abstract】 In research of ontology mapping, the mapping between real ontology or large-scale ontology is difficult. This paper proposes a mapping method for large class hierarchies, which is one kind of the large-scale ontologies. Based on ontology structure and semantic distance between classes, the classes can be put in a multi-dimension space by using Vector Space Model(VSM). Two large class hierarchies are partitioned into small blocks respectively by applying CURD clustering algorithm, and then blocks from different hierarchies are matched by comparing the reference points. Experimental results show that the method performs well on the test cases.

【Key words】 large-scale ontology; clustering; block; mapping

1 概述

当前 Web 上不同领域的本体越来越多，如何重用现有的本体，通过一定方法对其进行展开和组合，以便集成不同本体是目前研究的热点。本体映射就是在不同的本体间搭建语义桥梁，以实现不同本体间的知识共享和信息交流。大本体是一种用来描述复杂现实世界的领域本体，具有概念数量庞大、相互之间关系复杂等特点。要实现它们之间的映射就要有针对其特点的映射方法。

本文提出一种针对类层次本体的基于聚类的本体分块与映射方法。类层次本体是一种常见的大本体。本文方法根据本体的结构和概念之间的语义距离，应用向量空间模式(VSM)将概念定位于多维空间中。并在此基础上，应用 CURD 聚类算法对概念进行聚类，形成若干个语义上相对独立的块。然后利用块中的参考点在 2 个本体的块之间建立映射关系。实验结果表明，该方法具有分块大小适中、无须人工介入、准确度高等优点。

2 相关研究

目前，大部分的映射方法如 QOA[1]，RiMOM[2]都是基于一般本体的映射算法。如果将上述方法应用于拥有超过几千个概念的本体上，会造成系统效率低下，甚至无法正常运行。即使能够运行，也无法得到预期的结果。并且这些方法都着重强调实现一对一的映射，不能做到多对多的映射。

关于大本体的分块与映射的方法，文献[3]中的分块解决方案使用 ε-connection，其核心思想是使具有包含关系的概念都能聚在同一块内；文献[4]使用“依赖图”和“island”算法来实现概念的分块；文献[5]先基于本体的结构和概念的相似度对概念进行分块，再利用预先定义的参考点和虚拟文档技术实现块与块之间的映射。但上述方法的结果都不理想，用 ε-connection 方法作本体分块的时候往往会产生很大的块，不利于进一步实现块之间的映射；文献[5]则须由本体专家预先指定要划分的块的个数。

3 本体分块

本体分块是本文映射方法的基础，分块策略的优劣将直接影响最终的映射结果。好的分块策略能使块与块之间的耦合性降到最低，而在块的内部，概念之间却有很好的内聚性。首先实现概念的向量化，将概念置于一个由所有概念信息架构起来的多维向量空间中，计算同一个本体内概念间的语义距离以及基于语义距离的语义传播算法，应用语义传播算法对概念向量进行迭代计算，最后用聚类的方法将本体分块。

3.1 概念的向量表示

在本体中的概念节点中，不仅概念的名字中包含了此概念的相关信息，而且本体的创建者在概念的标记和对概念的注释赋予更为丰富的信息。因此，本文将建立一个以从本体的概念名字(*local name*)、概念标记(*rdfs*: *label*)和概念注释(*rdfs*: *comment*)中解析出的单词作为维度的向量空间，概念节点将表示在这个向量空间中。

定义 1 设 C 为本体中一概念节点，则 C 的向量表示为

基金项目：国家自然科学基金资助重点项目(60433020)
作者简介：徐德智(1963－)，男，教授，主研方向：Web 计算；陶 克，硕士研究生
收稿日期：2008-04-21 **E-mail：**taoke2216@126.com

$$\boldsymbol{V}_C = \alpha_1 \times collection\ of\ words\ in\ the\ local\ name + \alpha_2 \times collection\ of\ words\ in\ the\ rdfs:label + \alpha_3 \times collection\ of\ words\ in\ the\ rdfs:comment \quad (1)$$

其中， *collection* 类似于集合，但允许元素重复出现；*collection of words* 是指从概念名字或概念标记和注释中解析出来的单词组成的集合；其中，α_1, α_2, α_3 为[0, 1]之间的有理数。

3.2 基于结构的语义距离

语义距离也是语言学中经常提到的一个概念，它指 2 个概念的相近程度。一般 2 个概念间的语义距离越小，其语义越相近，反之越远。

在类层次本体中，所有的概念根据语义被表示在一个有向图中，图 1 为部分概念层次图，用概念在图中的最短路径来表示它们的语义距离。

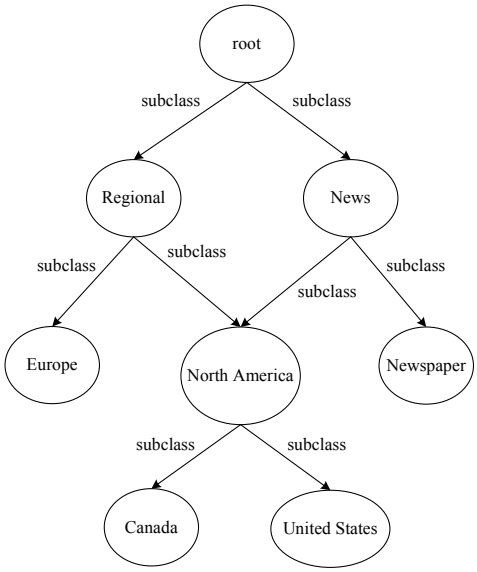

图 1 部分概念层次图

3.2.1 概念权值

所有从概念 C 引出的边具有相同的权值。本文将从 C 引出的边的权值简称为概念 C 的权值，记为 $weight(C)$。由图 1 中的 1 根节点 root 可见：自根节点开始，概念的分类由大到小，大类间的语义距离一般小于小类间。因此，离根节点越远，概念的语义距离权值越小。概念的子节点越多，表示它分得越细，权值越小。

定义 2 在概念层次图中，概念 C 的权值可表示为

$$weight(C) = \frac{2^{dep(C)}}{wid(C)} \quad (2)$$

其中，$dep(C)$表示从根节点到概念 C 的所经过的节点数；$wid(C)$表示概念 C 的宽度，即孩子节点的数目。

3.2.2 语义距离

概念层次图是有向图，它表示概念之间的包含关系。但在计算语义距离时，将把它看作无向图，因为可认为一个概念到它的子概念的语义距离等于他的概念到它本身的语义距离。

定义 3 在概念层次图中，概念 C_1 和概念 C_2 间的语义距离 $Dist(C_1, C_2)$为连接它们的最短路径上 n 条边的权值的总和，即

$$Dist(C_1, C_2) = \sum_{i=1}^{n} weight_i \quad (3)$$

3.3 语义传播算法

在本体中父子节点之间，兄弟节点之间，甚至祖先节点和子孙节点之间都存在一定的相似度。为基于概念间的语义距离来衡量概念之间的相似度，本文提出语义传播的概念。语义传播是指语义距离较近的 2 个概念在语义上具有互包含性。将这个互包含性定量表示就是语义传播　系数。

定义 4 对于同一个本体中的 2 个概念，如果它们之间的语义距离为 $Dist$，则它们之间的语义传播系数 e 为

$$e = \frac{1}{2^{\sqrt{Dist}}} \quad (4)$$

由式(4)可见，语义距离越小，则语义传播系数越大，反之亦然。

本文基于语义传播系数构造了语义传播算法。该算法根据语义传播系数对本地概念向量和邻近节点的概念向量进行迭代运算，使概念向量包含尽可能多的语义信息。

算法 1 语义传播算法

输入 本体中的概念向量

输出 经过计算后的概念向量

(1)计算所有概念间的语义距离；

(2)对任一概念 C，查找与之语义距离小于 D 的所有概念节点 p；

(3)按式(5)计算概念 C 的新的向量表示：

$$\mathbf{V'}_C = \mathbf{V}_C + \sum_{Dist(c,p)<D} e(c,p)\mathbf{V}_n \quad (5)$$

(4)重复 step2 和 step3，直到所有的概念都计算完毕。

为得到更好的概念表示结果，须反复调用语义传播算法。实验结果表明，5 次迭代即可得到比较理想的结果。

3.4 分块聚类算法

将本体中的概念转换为多维空间中的节点后，对这些节点的聚类，本文应用一种基于参考点和密度的聚类算法 CURD(Clustering Using References and Density)[6]。该算法适合对大规模数据的挖掘，它通过参考点来准确地反映数据的空间几何特征，然后基于参考点对数据进行分析处理。

定义 5 对空间中任意点 p 和距离 $Radius$，以 p 点为中心，，半径为 $Radius$ 的空间内的点的个数称为点 p 基于距离 $Radius$ 的密度，记作 $Density(p, Radius)$。

定义 6 对空间中任意点 p, 距离 $Radius$ 和阈值 t，如果满足 $Density(p, Radius) \geq t$，则称 p 为参考点，同时称 t 为密度阈值。参考点不是实际输入数据中的点，而是虚拟点或称为假想点。

定义 7 每个参考点代表了以该点为中心、半径为 $Radius$ 的空间区域，该区域称为参考点的代表区域。

定义 8 给定距离 $Radius$ 和阈值 t，如果参考点 p, q 满足 $Dist(p, q) \leq 2Radius$，即 p, q 之间的距离小于或等于 2 倍于 $Radius$，则称 p, q 为邻接参考点。

算法 2 分块聚类算法

输入 多维向量空间中的概念节点

输出 完成聚类后的若干概念集

(1)寻找满足条件的参考点；

(2)建立参考点与其所代表区域内点的映射；

(3)对参考点进行分类,符合条件的邻接参考点构成了一个聚类的基本信息；

(4)属于同一类的参考点代表区域内的点的集合，最终构成一个聚类。

4 本体间块映射

2 个本体间块的相似度可通过计算块中参考点的相似度来衡量：设 R_1, R_2 分别是本体 O_1, O_2 中的参考点，则其相似

度为

$$S(R_i,R_j)=\frac{\sum_{k=1}^{M}n_{ik}n_{jk}}{\sqrt{\sum_{k=1}^{M}n_{ik}^2\sum_{k=1}^{M}n_{jk}^2}} \tag{6}$$

其中，M 为向量空间的维度，$n_{ik}(n_{jk})$是向量在第 k 维上的坐标；设 B_i, B_j 分别是本体 O_1 和 O_2 中的分块，R_i, R_j 分别为其中的参考点，若其相似度 $S(R_i, R_j)>u$，则称 R_i 和 R_j 为一对锚点。其中 u 为参考阈值。

通过计算锚点，得出不同本体间块 B_i 和 B_j 的相似度计算公式为

$$S_B(B_i,B_j)=\frac{2\cdot anchors(B_i,B_j)}{ref(B_i)+ref(B_j)} \tag{7}$$

其中，$anchors(B_i, B_j)$返回锚点对数；$ref(B_i)$是 B_i 中的参考点个数。

5 实验测试

5.1 测试数据集

采用 OAEI 中“directory”测试数据集。该数据集中的数据来源于 Google 和 Yahoo 等 Web 目录，它由 2 265 对本体文件对构成，每一个本体文件均由一个类层次树构成。 本文实验先将这 2 265 个本体文件对合并成 2 个大的类层次本体：源本体和目标本体，实验将在这 2 个本体上验证本文的分块聚类与映射算法。

5.2 测试结果

在实验中，式(1)中的 α_1, α_2, α_3 分别取 1.0, 0.5, 0.25；式(5)中的 D 为 3.5；聚类算法中的 *Radius* 为 4.2，密度阈值 t 为 17；定义 10 中的 u 为 0.5。

经合并之后的源本体，包含 1 067 个概念和 1 313 个 *rdfs*: *subclass* 关系，最大深度为 10；而目标本体则包含 1 560 个概念和 2 331 个 *rdfs*: *subclass* 关系，最大深度为 9。

源本体经过分块聚类后被分为 6 块，最大的块包含 211 个概念，最小的包含 161 个概念，共产生了 34 个参考点；而目标本体则被分为 7 块，最大的块包含 423 个概念，最小的包含 207 个概念，共产生 47 个参考点。

源本体和目标本体间的块映射结果如表 1 所示，块的名字则取其中的最大根节点。实验共找到 8 个块映射对，其中 5 个正确，1 个映射对未找到(Sports vs. Sport)。查准率 *Precision*= 5/8=0.63，查全率 *Recall*=5/6=0.83，结果较为理想。

表 1 块映射结果

source	target						
	United	Recipes	Music	Games	Software	Diseases and	Sport
United	0.31	0.16					
Cooking		0.31	0.18				
Video			0.23	0.21			
Software					0.30		
Health						0.25	
Sports							

6 结束语

本文就大规模本体之间的映射问题，提出一种针对类层次本体的分块与映射方法。该方法能对大规模的类层次本体的概念进行比较合理地聚类分块，并能在块之间建立起比较好的映射关系，但其查准率还有待进一步提高。

目前该方法仅局限于类层次本体，下一步研究方向是将本方法应用于其他类型的本体，如包含属性和更复杂的关系的大规模本体，同时将本方法与其他映射方法相结合，以实现更为精确的概念对之间的映射。

参考文献

[1] Ehrig M, Staab S. QOM——Quick Ontology Mapping[C]//Proc. of the 3rd International Semantic Web Conference. Hiroshima, Japan: [s. n.], 2004: 683-696.

[2] Li Yi, Li Juanzi, Zhang Duo, et al. Result of Alignment with RiMO-M at OAEI'06[D].Beiying: Tsinghua University, 2006.

[3] Grau B, Parsia B, Sirin E, et al. Automatic Partitioning of OWL Ontologies Using ε-connection[C]//Proc. of the International Workshop on Description Logics. Edinburgh, Scotland, UK: [s. n.], 2005: 231-238.

[4] Stuckenschmidt H, Klein M. Structure-based Partitioning of Large Concept Hierarchies[C]//Proc. of the 3rd International Semantic Web Conference. Hiroshima, Japan: [s. n.], 2004: 289-303.

[5] Hu Wei, Zhao Yuanyuan, Qu Yuzhong. Partition-based Block Matching of Large Class Hierarchies[C]//Proc. of the 1st Asian Semantic Web Conference. Beijing, China: [s. n.], 2006: 72-83.

[6] 马 帅, 王腾蛟, 唐世渭, 等. 一种基于参考点和密度的快速聚类算法[J]. 软件学报, 2003, 14(6): 1089-1095.

(上接第 74 页)

5 结束语

本文设计并实现了一种基于 CIM 模型的资源信息模型，根据这种模型设计和实现了能够提供集中、统一的资源信息服务的资源信息服务器，并对这种资源信息服务器对整个系统性能的影响进行了测试和分析，得出了在一定规模和发生足够多次调度事件的情况下，统一的资源信息服务机制能够改善资源管理系统的性能的结论。下一步将研究在大规模分布式环境下集中、统一的资源信息服务机制的实现和对整个系统性能的影响。

参考文献

[1] Distributed Management Task Force Inc.. Common Information Model(CIM) Core Model Version 2.4(DMTF)[Z]. 2000.

[2] 曾 琼, 卢宇彤, 沈志宇. 基于CIM的机群系统资源信息模型[J]. 计算机工程, 2004, 30(13): 64-66.

[3] Strassner J. 活动目录网络——智能网络的基础[M]. 北京: 北京希望电子出版社, 2000.

[4] Neumann H. High Performace Computing(HPC) Cluster Manager[Z]. IBM Linux Technology Cernter. 2002.