基于参考点的大规模本体分块与映射*

赖 雅，王润梅，徐德智
（中南大学 信息科学与工程学院，长沙 410083）

摘 要：针对大规模本体映射中存在查全率和查准率不高的问题，提出了一种新的基于参考点的大规模本体分块与映射的方法。该方法的主要思想是用参考点来指导分块，并同时对待映射的两个大规模本体同时分块，即联合分块。首先对大规模本体进行预处理，将本体中的实体名称归一化并将其表示成本体树的形式，然后采用一些简便的方法找到参考点，最后以参考点为聚类中心对两个本体树的概念进行聚类，并同时实现块映射。理论分析和实验结果表明，该方法能够有效地解决大规模本体映射问题，并能获得较好的查全率和查准率。

关键词：大规模本体；本体映射；参考点；联合分块；块映射
中图分类号：TP391 **文献标志码**：A **文章编号**：1001-3695(2013)02-0469-03
doi:10.3969/j.issn.1001-3695.2013.02.041

Anchor-based large-scale ontologies partitioning and mapping

LAI Ya, WANG Run-mei, XU De-zhi
(*School of Information Science & Engineering*, *Central South University*, *Changsha* 410083, *China*)

Abstract: In order to solve the problem of low precision and low recall of large-scale ontology partitioning and mapping, this paper proposed a new anchor-based large-scale ontology partitioning and mapping method. This method used anchors to guide partitioning, and partitioned the two ontologies at the same time, which called co-clustering. Firstly, it preprocessed the two ontologies in order to normalize the entities's name and turn them into tree structure, then used some simple methods to find anchors. At last, the anchors acted cluster centers to cluster the concepts in both ontology trees, and found block mappings at the same time. Theoretical analysis and experimental results show that this method both solves the large-scale ontologeis mapping problem and achieves good precision and recall.

Key words: large-scale ontology; ontology mapping; anchors; co-clustering; block mapping

语义 Web 被称为下一代互联网，是对现有互联网的扩展的延伸，其基本思想是构建一个巨大的知识网络，网络中的信息都是具有语义的，从而能够被计算机理解并处理，提高计算机处理信息的自动化和智能化，更好地实现人机之间的合作和交互。

本体是实现语义网的关键技术，由于不同的本体创建者在创建本体时，所采用的描述语言、描述方式、创建意图可能不同，即使是对同一个领域的建模，也会构造出不一样的本体，因此不可避免地存在本体异构问题。本体映射是解决本体异构问题的最好方法。但是随着语义 Web 的发展，本体的规模越来越大，越来越多的应用需要在大规模本体之间进行匹配。由于大规模本体自身所具有的概念数目庞大、概念之间关系复杂的特性，采用普通本体的映射方法[1,2]往往会造成效率低下，有的甚至因为缺乏内存而无法运行，因此，应该针对大规模本体采取不同的映射方法，本文着重对此进行研究。

1 相关工作

目前对本体映射的研究很多，方法也比较成熟，但是在大规模本体映射方面，国内外研究都比较少，主要的方法都是先对大规模本体进行分块，然后对各个块进行映射，最后在匹配的块之间进行实体映射。文献[3]基于本体结构相似度和概念语言级相似度对本体中的概念层次进行划分，然后通过预定义的参考点和虚拟文档技术进行块匹配，该方法只对本体中的概念进行划分，忽略了本体中其他元素，如属性、实例，而且需要人工给定分块个数，分块的自动化程度不高。文献[4]考虑实体语言学和结构上的信息，利用虚拟文档技术计算实体间的相似度，然后采用层次的对分算法进行分块。文献[5]提出了一种基于结构的划分方法，首先计算实体的结构相似度，并采用凝聚的层次聚类算法对实体进行聚类，再通过 RDF 句子将类构建成块，通过参考点在各个块中的分布找到块映射，最后采用 V-Doc 和 GMO 进行块内映射，进而输出最后结果。但是在结构相似度的计算方面忽略了实体的细分程度，因而分块结果不是很精确，且在参考点寻找过程中需要人工参与。文献[6]能够处理任意大小的本体映射问题，算法开始于一对预先找到的映射对，然后通过不断地搜索其邻居节点找到更多的映射对，从而完成整个本体的匹配。不足之处在于需要人工给出参考点，而且对于在结构上不是很对称的本体映射任务，映射质量不是很理想。

除此之外，还有一些专门对本体进行分块的方法，如网络划分法[7]、ε-connection[8]法等，但它们均存在一些缺点与不

收稿日期：2012-05-28；**修回日期**：2012-07-20 **基金项目**：国家自然科学基金资助项目(60970096)；中南大学研究生重点课程建设资助项目(1960-71131430019)

作者简介：赖雅(1986-)，女，硕士研究生，主要研究方向为本体映射(laiya_csu@126.com)；王润梅(1982-)，男，硕士研究生，主要研究方向为语义网、本体映射；徐德智(1963-)，男，教授，博士(后)，主要研究方向为 Web 计算、语义网等.

足。网络分块法只考虑具有直接父子关系的节点之间的联系且认为每种联系的重要程度都相等,因此分块结果不理想;ε-connection 法将具有包含关系的所有概念都划分到同一个块中,往往会产生过大的块。

2 基于参考点的大规模本体分块与映射

2.1 传统方法的缺点

从第 1 章的介绍可知,传统的大规模分块与映射方法一般采用分而治之的思想,先将待映射的两个大规模本体单独分块,再进行块映射。这种方法虽然能够大大降低大规模本体之间的映射复杂度,但是会造成两个方面的问题:a)在本体分块阶段,可能映射的两个实体未被分到可能形成块映射的两个块中;b)在块映射阶段,可能映射的两个实体所在的块未形成块映射。这两种原因都有可能导致最后实体映射的查全率和查准率不高,会造成很多错误的匹配和失配的情况。单独对两个本体分块,由于各个本体分块时所依据的原则、所要考虑的因素不同,因此会造成它们之间许多关联的语义信息的缺失。即使分块质量很高(即块内高内聚,块间低耦合),如果块映射质量不高,也会造成查全率和查准率的缺失。另外在块映射阶段,由于本体存在不同形式的异构,而且精确地计算块与块之间的相似度花费的代价很高,一般采用启发式的方法,例如根据参考点的分布来确定块映射,只利用了本体的语言学方面的信息,忽略了本体中的其他语义信息,对于不同的映射任务块映射的质量也不一样。由于传统的方法从分块到块映射要经过两个主要的步骤,这中间含有许多变化的因素,即使两个环节的质量都高,也无法保证最后映射发现的查全率和查准率。因此,对于大规模本体分块与映射,本体考虑采用一种新的方法,用参考点来指导分块,将分块和映射作为一个步骤来完成,即联合分块。这种方法克服了传统方法造成的不便,同时也能够大大地提高分块与块映射的质量,为实体之间的映射发现打下基础。

2.2 本体预处理

由于不同本体所采用的描述方法和描述语言不同,在对它们进行分块与映射之前,首先要对其进行归一化处理,这里称为预处理。预处理的目的是消除名称冲突,并将本体转换成树状结构。

算法 1 本体预处理。

输入:待映射的大规模本体。

输出:本体树。

a)解析本体,获取本体中的实体,包括概念、属性和实例,以及子类,instance-of、attribute-of 等关系。

b)对所有的实体名称进行预处理,包括分词、缩写词扩展、词干抽取、名称规范化和命名实体。

c)根据概念的子类关系构建本体树。

(a)如果树为空,将根概念加入树中作为根节点。

(b)按从左到右的顺序,依次处理当前深度层的概念节点,将它们的直接子概念加入到树中,树的深度加 1,重复执行该步骤,直到所有的概念都处理完毕。

(c)根据 instance-of 和 attribute-of 关系将所有的实例和属性与其对应的概念关联起来。

对本体进行预处理之后,待映射的两个本体被表示成两棵本体树。

2.3 本体分块与映射

2.3.1 参考点的获取

在对本体进行预处理之后,将要寻找两棵树之间的参考点信息。首先给出参考点的定义。

定义 1 具有很高相似度且很容易被找到的实体映射对称为参考点。

参考点一般很容易找到,本文采用如下方法来寻找参考点:

a)采用基于实体名称的字符串匹配方法,找到完全匹配的实体对。

b)利用 WordNet 同义词词典找到互为近义词或者在 WordNet 中具有相等关系的实体对。

c)利用本体可视化工具,如 Protégé,人工识别匹配的实体对。

对于第三种方法,由于需要人工参与,大大降低了系统的自动化程度。在实际实验中,本文考虑采用一种折中的策略,仅在前两种方法找到的参考点比较少的时候,本文才使用第三种方法。

2.3.2 分块与块映射算法

在给出具体的方法之前,先介绍以下两个定义。

定义 2 将大规模本体分解后的一个个本体片段,称为块(block)。

定义 3 来自两个不同的本体的块之间的映射关系称为块映射(block mapping)。

块映射与实体映射有区别,而本体映射的最终任务是发现实体与实体之间的映射,因此块映射和分块都是为实体映射服务的。

本文方法是基于这样的思想,根据参考点多少以及在本体树中的分布,选定一个或多个参考点作为聚类中心,然后指定一个初始的语义半径,通过不断地缩小语义半径的大小来对本体中的概念进行聚类,即同时对两个本体树进行分解,分解的同时也实现了块映射。

算法 2 基于参考点的大规模本体分块与映射算法

输入:待映射本体的有向无环图,参考点以及同一个本体中概念的语义相似度矩阵 W_T 和 W_S。

输出:分块和块映射结果。

a)将每一个参考点作为初始的块中心,并对其进行分别编号,如 $T_{b1}, T_{b2}, T_{b3}, \cdots, T_{bk}$ 以及 $S_{b1}, S_{b2}, S_{b3}, \cdots, S_{bk}$,其中 k 为参考点的个数,也即分块数。

b)给定初始语义半径,将与各参考点语义距离为 1 的概念加入到参考点所在的块中,并将该概念与所有参考点的相似度值置为 0。

c)根据语义相似度的大小依次进行以下处理:

(a)对于矩阵 W_T,寻找最大的语义相似度值,将其所对应的概念加入到该参考点所在的块中,并将该概念与所有参考点的相似度值置为 0。

(b)对于矩阵 W_S,寻找最大的语义相似度值,将其所对应的概念加入到该参考点所在的块中,并将该概念与所有参考点的相似度值置为 0。

d)重复执行步骤 c),直到所有的概念都处理完毕。

e)根据参考点来识别块映射,即参考点中的两个概念分

别所在的块形成块映射。

f)输出分块及块映射结果。

g)算法结束。

对算法 2 的补充说明如下:由于算法中只用到概念与参考点之间的语义相似度,在计算语义相似度的时候只需计算参考点中的概念和其他概念的语义相似度,大大节省了计算时间。在进行聚类的过程中,每个概念只加入到一个块中,所以分块结果不会有重复;在进行块映射时,由于是根据参考点来确定,所以任何一个块都只存在于一个块映射中。

3　实验结果及分析

3.1　实验环境

SNAX(sematic native XML database)系统是一个语义原生 XML 数据库,被视为一个实验性质的微型语义 Web,主要由原生 XML 数据存取管理器、本体管理器、多功能查询器和 Web 数据采集器几部分组成。SNAX_Map[9] 在 SNAX 系统的基础上加入本体映射的模块。

本文在 SNAX_Map 系统的基础上加入了三个模块,分别为本体预处理模块、参考点获取模块、分块与块映射模块,并将分块和块映射的结果作为该系统的输入。

3.2　实验数据

本文采用本体集 Russia12 作为实验数据集,该数据集包含 Russia1 和 Russia2 两个本体,都是用来描述俄罗斯的旅游产业,均能够从网站上获得。选择它们的原因有:其规模适中,不是太大也不是太小,既能够用来说明大规模本体映射问题,又便于对实验结果进行分析。另外其参考文件中包含了参考映射对,其中,Russia1 包含了 151 个类和 76 个属性;Russia2 包含了 162 个类和 81 个属性。

3.3　评价标准

由于本体分块以及块映射最终是为本体映射服务的,本文不对分块质量和块映射质量进行评价,只对最后的映射结果进行评价。本文采用信息检索领域的查全率和查准率作为实体映射的评价标准,其公式分别如式(1)(2)所示。

1) 查准率(precision)

$$P = A/(A + B) \tag{1}$$

2) 查全率(recall)

$$R = A/(A + C) \tag{2}$$

其中:A 表示找到的正确的映射,B 表示找到的错误的映射,C 表示未找到的正确的映射。

3.4　实验结果

通过算法 1 对 Russia12 中的两个本体进行预处理后,其中 Russia1 的最大深度为 9,Russia2 的最大深度为 6,找到的参考点有 15 对,因此最后的块映射个数也为 15 个。由于篇幅限制,在这里就不列出详细的参考点以及分块和块映射结果。表 1 列出了本文算法的查全率和查准率,并将其与另外一个著名的大规模本体分块和映射系统 PBM(partition-based block matching)[5] 进行比较。

表 1　查全率和查准率比较

system	P	R
SNAX_Map	0.67	0.53
PBM	0.65	0.59

从表 1 中可以看出,本文系统的查准率稍高于 PBM,但是查全率不及 PBM。通过仔细分析得知,PBM 采用的是分治法的思想,先对大规模本体进行分块,然后利用参考点来进行块映射,刚好与本文方法的步骤相反。而在进行块映射的时候,本文采用一一对应的关系,而 PBM 采用的是多对多的方式,可能找到的实体映射对会比本文的方法多,因此,实体映射的查全率要高于本文的方法。另外系统的查准率比 PBM 高,这是因为本文采用参考点来指导分块,并利用参考点来识别块映射,避免了先分块后映射的盲目性,所以映射的准确率会很高。综上所述,从这点可以看出,本文的方法虽然保证了实体映射的查准率,但是在查全率上有所缺失。

另外在实验过程中发现,依然存在具有实体映射关系的两个概念未被分到一个块中的情况,这是因为本文的分块算法为了避免块与块之间实体的交叉,只将每一个概念划分到一个块中。在后续工作中可以考虑允许块中实体重叠的情况,可以有效地提高查全率。由于参考点在块中分布不是很均匀,算法 2 中也没有对块的大小作任何限制,所以形成的块的大小不一,本文会继续探究由于这方面原因对映射结果造成的影响。

4　结束语

针对大规模本体映射问题,本文提出了一种基于参考点的大规模本体分块与映射方法,用参考点来指导分块过程,对两个大规模本体进行同时分块,并根据参考点来实现块映射。实验证明,该方法能够有效地解决大规模本体映射问题,同时也能获得较好的查全率和查准率。

本文的下一步工作是对分块算法进一步改进,考虑允许块内元素部分交叉的情况,从而进一步提高实体映射的查全率和查准率。

参考文献:

[1] MAO Ming, PENG Ye-fei, SPRING M. Ontology mapping: as a binary classification problem[J]. Concurency and Computation: Practice and Experience, 2011, 23(9): 1010-1025.

[2] LEVY R, HENRIKSSON J, LYELL M, *et al*. Ontology matching across domains[C]//Proc of Agent-Based Technologies and Applications for Enterprise Interoperability. Berlin: Springer, 2012: 95-113.

[3] HU Wei, ZHAO Yuan-yuan, QU Yu-zhong. Partition-based block matching of large class hierarchies[C]//Proc of the 1st Science Asian Semantic Web Conference. Berlin: Springer, 2006: 72-83.

[4] HU Wei, QU Yu-zhong. Block matching for ontologies[C]//Proc of the 5th International Semantic Web Conference. Berlin: Springer, 2006: 300-313.

[5] HU Wei, QU Yu-zhong, CHENG Gong. Matching large ontologies: a divide-and-conquer approach[J]. Data & Knowledge Engineering, 2008, 67(1): 140-160.

[6] Md SEDDIQUI H, AONO M. An efficient and scalable algorithm for segmented alignment of ontologies of arbitrary size[J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2009, 7(4): 344-356.

[7] STUCKENSHMIDT H, KLEIN M. Structure-based partitioning of large concept hierarchies[C]//Proc of the 3rd International Semantic Web Conference. Berlin: Springer, 2004: 289-303.

[8] GRAU B C, PARSIA B, SIRIN E, *et al*. Automatic partitioning of OWL ontologies using E-connections [C]//Proc of International Workshop on Description Logics. 2005: 1-22.

[9] ZHANG Zhi-wei, XU De-zhi, ZHANG Tian. Ontology mapping based on conditional information quantity[C]//Proc of IEEE International Conference on Networking, Sensing and Control. 2008: 587-591.