基于模块抽取的大本体分块与映射

王润梅 徐德智 赖 雅 姚学聪

(中南大学信息科学与工程学院 长沙 410083)

摘 要 大本体规模过大,使得本体间映射复杂。针对已有方法在分块上的不足,提出一种基于模块抽取的大本体分块映射方法。通过建立本体依赖图的拉普拉斯矩阵来抽取本体模块,计算模块之间的相似度,实现分块映射。实验结果表明,该方法能有效实现大本体分块,提高映射效率。

关键词 大本体,词干,模块抽取,模块主题相似度

中图法分类号 TP393 **文献标识码** A

Large Ontology Partition and Mapping Based on Module Extraction

WANG Run-mei XU De-zhi LAI Ya YAO Xue-cong

(School of Information and Engineering, Central South University, Changsha 410083, China)

Abstract The ontology mapping is complex because of the large size of ontology. Aiming at the defect of existing approaches, this paper presented a new way which is based on the module extractedion to realize the large ontology partition and mapping. We extracted the ontology modules by the Laplace Matrix of the dependency graph, and computed the module similarity, then got the mapped modules. The empirical results indicate that our method can achieve ontology partition and enhance the mapping efficiency.

Keywords Large ontology, Word stem, Module extraction, Similarity of module subject

1 引言

随着网络本体信息的不断增加,要实现人机协同处理,不可避免地要对这些分布式数据进行相互操作。通过定义相关条件规则和函数逻辑可以实现这些分布式本体间的相互关联。然而当前Web上本体规模越来越大,知识库中概念数量越来越多,概念之间关系错综复杂,要实现这些大本体之间的映射,必须对其进行分块。

本体模块化实质上是将软件工程中的模块技术应用到本体中,解决本体分块问题。其实现可通过建立标准的形式化本体模块构造本体和从已有本体中抽取本体片段,建立小规模子本体。前者处理代价过大,可操作性不强,没有得到充分应用。后者由于可降低大本体维护的复杂度,目前在一些具体研究领域[1]已经得到应用。

2 相关研究

目前,利用本体抽取方法实现大本体分块的方法主要可以分为以下3大类:1)基于查询的分块方法[2];2)基于描述逻辑的分块方法[3];3)基于网络理论的分块方法[4-6]。文献[2]是基于查询的分块方法,它通过SparQL的语法结构定义简单的可视化查询机制,获取SparQL信息,以完成分块任务,其缺点是在对本体概念间语义的利用率较低。文献[3]是基于描述逻辑的分块方法,它提出了本体签名和局部语法规则的概念,通过建立启发性规则从外部抽取相关模块,导致实现分块的复杂度高。基于网络理论进行分块的研究过程中,文献[4]先计算概念之间的相似度,然后利用层次分类法进行分块,借助V-Doc技术和GMO技术实现块间映射,其缺点是在分块时忽略了两个层次之间的一致性,映射查准率有待提高。文献[5]则把存在包含关系的概念聚合在同一本体块中,由于过于强调分块过程的严谨性,使得产生的本体模块太大,本体块间的映射受到限制。文献[6]通过本体依赖图和Island算法实现概念的分块,分块的质量取决于预先给定的参数。

针对以上方法的不足,在现有分块方法的基础上,本文提出一种基于有向带权图的模块抽取算法,以实现大本体之间的分块映射。本文第3节介绍本体预处理的方法;第4节提出一种基于模块抽取的本体分块映射算法;第5节是实验结果和分析;最后对本文工作进行了总结,同时提出了后续研究方向。

3 本体预处理

预处理关系到本体分块工作的效率,其处理时间一般占整个大本体分块映射时间的一半以上,算法通过数据元素的抽取和本体词干的提取对大本体进行预处理。

3.1 数据元素的抽取

为方便叙述,先给出数据元素的定义。

定义1 本体中的数据元素指的是本体形式化定义中的

到稿日期:2010-11-03 返修日期:2011-03-24 本文受国家自然科学基金项目(60970096)资助。

王润梅(1982-),男,硕士生,主要研究方向为大本体分块映射,E-mail:wrmfoolish@163.com;徐德智(1963-),男,博士后,教授,主要研究方向为Web计算、语义网;赖 雅(1984-),女,硕士生,主要研究方向为本体映射;姚学聪(1982-),女,主要研究方向为本体映射。

组成元素，主要包括概念、关系、实例等。

通过 Jena 建立本体模型、抽取本体数据元素，在抽取时综合考虑同义词、反义词、停用词以及一词多义的现象，来有效提高关键词的密度和抽取的执行效率。

3.2 词干提取

定义 2 词干是概念的核心部分，是指将抽取的核心概念的词缀删除后剩余的部分。

词干提取基于词缀后缀，统一用(condition)S1－>S2 表示，即当概念前面的主干部分满足给定条件，且以常见后缀 S1 结尾时，将 S1 用 S2 代替。如可通过 ate－>null，ation－>ate，s－>null，ing－>null 等操作将概念 indicate、indication、indications、indicating 用同一词干 indict 表示。这样可以把相同词干的核心概念集中到一起，从而降低处理的空间复杂度。

4 本体分块与映射

4.1 本体模块的相关概念

定义 3 本体模块指的是本体中关于某一核心主题的概念簇。这些概念和核心主题密切相关，而与模块之外的其它概念关联性不强。将其形式化表示为 $M=(ID,C,R,T)$，其中 M 表示本体模块，ID 对应模块的定位标识，C 表示模块中所定义的概念的集合，R 是模块中定义的关系的集合，T 指的是本体模块的主题，其描述的内容包括模块领域特征和其它非函数属性等。

如果一个大规模本体 O 可分解成 n 个本体模块，则大本体 O 也可表示为 $O=\{M_1,M_2,\cdots,M_n\}$，称之为模块化本体。

4.2 本体模块的抽取标准

(1) 内聚度是指模块内部概念之间的依赖程度，即模块内部概念之间关系的权值 $w(e)$ 之和为

$$cohension(M_i)=\sum_{e\in M_i} w(e) \tag{1}$$

(2) 耦合度是用来衡量模块之间相互关联和依赖的紧密程度，是指模块 M_i 和其它模块之间的有向边的权值总和。其计算公式为

$$couple(M_i)=\sum_{e\in(M_iM_j)} w(e) \tag{2}$$

式中，$w(e)$表示模块 M_i 与其它模块之间的权值。

(3) 规模是指本体模块中包含的概念数量。如果规模过大，则分块效果没有明显提高，规模较小则会降低本体映射的效率。本文采用的规模评价的标准为

$$Qsize(M_i)=\begin{cases}0.5-0.5\cos(\frac{\pi x}{25}), & 0<x\leqslant 50\\ 0, & x>50\end{cases} \tag{3}$$

式中，x 表示模块的规模，模块在满足规模适中的前提下，还要考虑模块内聚度和耦合度的有效调和。

4.3 模块抽取方法

本文在抽取模块前，先对本体进行图示化描述，再将相同词干的概念结点合并，优化本体依赖图，求出各概念结点之间的边的权值。

4.3.1 本体图示化

将本体模型定义的数据元素表示成本体依赖图 $G=\langle V,E,W\rangle$。V 表示本体中概念结点的集合，E 表示概念之间存在的边的集合，W 是边的权值。如果和某一结点 N 相连的其它结点数目越少，它们和 N 相连的边对于结点 N 就越重要，依赖性也越强，因此边的权值就越高。反之则越低。

4.3.2 优化本体依赖图

在本体图示化的过程中，常用的边和权值处理方法并未考虑相同词干的概念。本文对前面得到的本体依赖图进行优化，将本体依赖图中相同词干的概念结点之间的权值设为 1。保留一个有相同词干的结点；将其称为保留结点，将同词干结点清除，其关系边连接到保留结点上。

(1)边的优化处理

本体依赖图中的边 E 用 OWL 语法表示，将其分成 4 类。A 类：等价边，边的两个顶点是对同一实体的不同描述。B 类：继承、部分与整体聚合关系，此类边联系紧密，边的重要度值也比较高。C 类：交、并、补等连接关系的边。D 类：其它限制、约束等关系。各类边重要度值设置为

$$\varphi_{ij}=\varphi(E(i,j))=\begin{cases}1, & E\in A\text{类边}\\ 0.7, & E\in B\text{类边}\\ 0.5, & E\in C\text{类边}\\ 0.3, & E\in D\text{类边}\end{cases} \tag{4}$$

(2)权值优化处理

引入上文设定的边的重要度函数 φ_{ij} 来修改结点 i 到结点 j 之间的边的权值 W_{ij}，具体为

$$W_{ij}=\frac{w_{ij}\varphi_{ij}+w_{ji}\varphi_{ji}}{\sum_k(\varphi_{ik}w_{ik}+\varphi_{ki}w_{ki})} \tag{5}$$

图 1 表示某本体片段的带权图。由式(5)计算可得，从结点 mammal 到结点 human 的边的权值为

$$W_{ij}=\frac{0.33*1+0.5*1}{(0.33*1+0.33*0.7*2)+(0.5*1+0.5*0.7*2)}=0.418$$

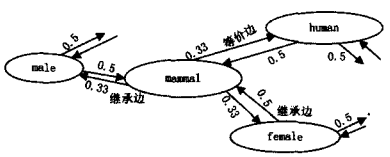

图 1 本体带权图

(3) 生成本体模块

首先由本体依赖图构造对应的拉普拉斯矩阵，求出其费德勒向量。若图连通，则可以根据费德勒向量中元素的值大于零或小于零将依赖图分解为两个子图；若图不连通，则利用零特征值对应的特征向量中的两个值将图进行分块。然后，以上述分块方法作为一次迭代，对有向带权图进行迭代分解，直至子图的独立性低于父图。由此得到一系列的独立子图，对应为本体模块。

与文献[7]类似，本文从模块规模、内聚度和耦合度对抽取的子图进行独立性度量，计算公式为

$$Independence(G_i)=\frac{cohension(G_i)}{couple(G_i)*sizeof(G_i)} \tag{6}$$

本体模块抽取算法如下。

算法 1 BiPartitionGraph(G)

输入：本体依赖图 G

输出：子图 G1、G2

LapMatrix=CreateLaplacianMatrix(G)；

λ_2=GetFiedler(G)；//求拉普拉斯矩阵的费德勒值

VectorFiedler=GetFiedlerVector(G)；//求出费德勒向量

If (λ_2) G[1,2]=SubGraph(G,VectorFiedler)

Else G[1,2]=SubGraph(G,λ_2)；

算法 2 ExtractModules(G)

输入：大本体 O

输出：本体模块 Modules(M_1,M_2,…,M_n)

Modules=0；

BipartitionGraph(G)；

If(Indepedence(G[1]) ≥Independene(G))

ExtractModules(G[1])；

Modules=AddModule()；}

If(Indepedence(G[2]≥Independene(G))

ExtractModules(G[2])；

Modules=AddModule()；

OutputModule(Modules)；

4.4 块间映射

将子本体描述为 $M=\langle ID,C,R,T\rangle$ 的本体模块形式。利用源本体和目标本体之间模块概念相似度和模块主题的相似度加权求和，计算本体模块的相似度。当模块间相似度大于某一阈值时，认为两模块相似，建立块间映射。

(1)概念相似度

概念相似度的计算，主要是从概念的结构着手，利用字符串的相似性进行判断，其计算公式为

$$\mathrm{Sim}_{\mathrm{concept}}(N_1,N_2)=\max\left(0,\frac{\min(|N_1|,|N_2|)-ed(N_1,N_2)}{\min(|N_1|,|N_2|)}\right) \tag{7}$$

式中，$ed(N_1,N_2)$表示概念 N_1 和 N_2 中不同字符个数的平均值。$|N_1|$表示概念 N_1 中所含字符的个数。

(2)模块主题相似度

定义 4 模块主题是指某一具体领域内的一个概念集群。利用主题的词干信息，基于模式级别计算其相似度。假设模块 M_i 有 m 个词干，模块 M_j 有 n 个词干，词干之间相似度采用 humming distance 计算方法为

$$\mathrm{Sim}(N_1,N_2)=1-\frac{\left[\sum_{i=1}^{\min(|N_1|,|N_2|)}f(i)\right]+||N_1|-|N_2||}{\max(|N_1|,|N_2|)} \tag{8}$$

当词干 $c1$ 和词干 $c2$ 中的第 i 个字符相同时，$f(i)$取值为0，不同时取值为1。通过计算得到模块 M_i 和模块 M_j 的 $m*n$ 个词干相似度，分别设置相应的权值 $w_1,w_2,\cdots,w_{m*n}$，则模块 M_i 和模块 M_j 的模块主题相似度为

$$\mathrm{Sim}_{\mathrm{subject}}(M_i,M_j)=\sum_{p=1}^{m*n}w_p\mathrm{Sim}_p(c1,c2) \tag{9}$$

合并模块概念相似度和模块主题相似度，得模块相似度计算方法为

$$\mathrm{Sim}(M_i,M_j)=W_{\mathrm{subject}}\mathrm{Sim}_{\mathrm{subject}}(M_i,M_j)+W_{\mathrm{concept}}\mathrm{Sim}_{\mathrm{concept}}(M_i,M_j)$$

式中，权值系数 $W_{\mathrm{subject}}+W_{\mathrm{concept}}=1$，当 $\mathrm{Sim}(M_i,M_j)$大于给定阈值时，建立本体模块之间的映射。

5 实验结果和分析

5.1 实验数据集

本文采用两个 Medical 本体(Medical A 和 Medical B)和两个 Tourism 本体(Tourism A 和 Tourism B)作为实验对象。这些本体规模适中，既考虑了数据集过小对分块造成的影响，也考虑到数据集太大超出硬件处理能力等因素。同时给定一个参考映射文档，可对实验结果做出合理评价。具体的数据信息如表 1 所列。

表 1 本体数据元素(个)

本体名称	概念数目	属性数目	实例数目	参考映射
MedicalA	398	175	163	217
MedicalB	443	219	247	
TourismA	340	97	103	226
TourismB	474	100	92	

5.2 评价指标

(1) 对于分块质量的评价，本文主要从模块的信息熵和独立性来考虑。

信息熵指的是将本体分块后，各本体模块中概念分布的概率。本体模块中概念的分布越有序，则其信息熵值就越低，分块的质量也越好。信息熵的计算公式为

$$Entropy(M_i)=\frac{-1}{\log m}*\sum_{j=1}^{m}P_j*\log P_j \tag{10}$$

$$Entropy(O)=\sum_{i=1}^{n}Entropy(M_i)*|M_i|/\sum_{i=1}^{n}|M_i| \tag{11}$$

式中，M_i 表示第 i 个本体模块，n 表示抽取的本体模块的数目，m 表示专家给定的参考分块数目，j 则表示第 j 个参考分块，P_j 表示本体的查准率。

由于信息熵的计算需要综合考虑实验环境(如运行的软件设置、硬件配置等)等前置条件，因此所求结果受到其它因素的影响。在此基础上，本文引入本体的模块独立性来评价分块质量。独立性的计算采用式(6)，计算出的独立性值越大，表示本体模块内部概念越靠近于同一概念主题。

(2) 对于本体映射效率的评价，通常利用查准率(Prec)和查全率(Rec)。查准率指的是算法算法找到的正确的本体映射对数目相对于给定参考映射文档中的映射对数目的比值。查全率指的是找到的正确的映射对数目与所检索到的映射对的比值。两者具有逆向相互依赖关系，查准率的提升往往会以查全率的降低为代价，反之亦同。

5.3 实验结果及数据分析

本文算法命名为 MPM(Module_based Partititon and Mapping)。

(1)分块性能(Entropy 值)如表 2 所列。

表 2 分块性能比较

	MedicalA	MedicalB	TourismA	TourismB
PBM	0.21	0.23	0.16	0.16
CBM	0.20	0.18	0.13	0.15
MPM	0.17	0.21	0.17	0.11

本文的分块算法对本体进行模块化聚合。从实验结果分析，MPM 算法在 TourismB 本体中的分块性能最佳，比另外两种算法有一定改进，主要是因为 MPM 算法能高效地从实例数目较少而概念数目较多的本体中抽取内聚性能高的模块。在 TourismA 中，由于其实例数目较多，而且概念数目较少，因此不能体现出本文算法的优越性。

(2) 映射效率如图 2 所示。

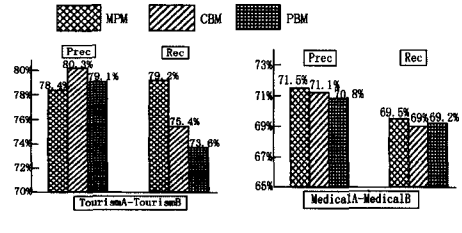

图 2 映射效率

与常用的分块映射模型 PBM 和 CBM 比较，本文算法在 TourismA 和 TourismB 中查全率有较大提高，查准率有小幅下降；而在 MedicalA 和 MedicalB 中查准率和查全率都有小幅提升，这是由于本文算法通过对相同或相近本体概念元素的聚合，使得本体间用于进行匹配比较的概念对数目增加，从而能有效提高查找的准确率。

结束语 在大规模本体中，为更好地实现分块映射，本文提出了一种基于模块抽取的方法。本文方法能抽取出独立性高的本体模块，有效实现大本体的分块映射。但目前对于分块质量的衡量还缺少一个统一的评价体系，无法对提出的分块算法进行精确地度量。下一步，本文将把研究重点放在抽取模块的质量好坏的评价上，如模块内部纯语义的评价等，同时将该评价方法和其它通用方法相结合，建立一个比较完备的评价体系。

参 考 文 献

[1] 方俊，郭雷，王晓东. 基于推理信息的本体模块化方法[J]. 计算机科学，2008，35(7)：177-180

[2] Seidenberg J, Rector A. Web ontology segmentation: analysis, classification and use[C]// Proceedings of the 15th International Conference on World Wide Web. 2006：13-22

[3] Grau B C, Horrocks I, Kazakov Y, et al. Extracting Modules from Onotlogies: A Logic-based Approach[J]. Modular Ontologies, 2009, 5445：159-186

[4] Hu Wei, Qu Yu-zhong, Cheng Gong. Matching large ontologies: A divide-and-conquer approach[J]. Data & Knowledge Engineering, 2008, 67：140-160

[5] 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. 2005：231-238

[6] Jin Long-fei, Liu Lei. An Ontology Slicing Method Based on Ontology Definition Metamodel[J]. Business Information Systems, 2007, 4439：209-219

[7] 罗景，赵伟，秦涛，等. 基于有向带权图迭代的面向对象系统分解方法[J]. 软件学报，2004，15(9)：1292-1300

(上接第 235 页)

表 2 最优解对比

Algorithm	Avg-BestFitness	Avg-SpendTime
CE1	49.3	43.4
CE2	62.7	35.1
CE3	62.4	34.8
CE-HOA	71.9	21.4

由表 2 可知，利用 CE-HOA 算法不仅适应值高，而且能够有效缩短寻优时间，提高算法效率，使协商更为高效。

(2)多议题让步对收益影响实验

分别对比多议题与单议题让步时的收益情况。选取指标为平均收益(Avg-U)和平均协商成功率(Avg-S)。

由表 3 可见，两种方法的协商成功率差别不大，但是获得的收益差别明显。这是由于考虑多议题让步后，决策者对偏好小议题采取较大让步，而偏好大议题采取较小让步。这样既能够保证协商成功率，又能够尽可能提高自身利益。

表 3 对比结果

Issue Type	Avg-U	Avg-S
Single-issue	62.3	89%
Multi-issue	81.5	92%

结束语 本文提出一种竞争及合作环境影响下的动态协商模型，并基于种群自生长原理构建了环境分析模型；为提高决策灵活性，提出一种引入战略特征的决策模型；为平衡各议题的让步值，利用一种混合优化算法进行最优组合的搜索。仿真实验结果表明，本文提出的模型达到了较好的效果。

在未来工作中，将针对其它影响因素进行研究，以进一步提高协商模型的实际应用效果。

参 考 文 献

[1] Byde P A, Bartolini C. Economic dynamics of agents in multiple auctions[C]// The 4th Int'l Conf on Autonomous Agents. Montreal, Camda, 2001

[2] Byde C P, Jennings N R. Decision procedures for multiple auctions[C]// Proc. of the 1st Int'I Joint Conf on Autonomous Agents and Multi agent Systems (AAMAS'02). New York: ACM Press, 2002

[3] Nguyen T D, Jennings N. A heuristic model for concurrent bilateral negotiations in incomplete information settings[C]// Int'l Joint Conf on Artificial Intelligence. Mexico, 2003

[4] Li C, Giampapa J A, Sycara K. Bilateral negotiation decisions with uncertain dynamic outside options[J]. IEEE Trans on Systems, Man, and Cybernetics, Part C: Special Issue on Game-theoretic Analysis and Stochastic Simulation of Negotiation Agents, 2006, 36(1)：31-44

[5] Sim K M, Choi C Y. Agents that react to changing market situations [J]. IEEE Tram on Systems, Man and Cybernetics, Part B: Cybernetics, 2003, 33(2)：188-201

[6] Sim K M, Wang S Y. Flexible negotiation agent with relaxed decision rules [J]. IEEE Trans on Systems, Man and Cybernetics, Part B: Cybernetics, 2004, 34(3)：1602-08

[7] 刘文俊，王天江，等. 不确定外部竞争和选择下的动态双边协商决策模型[J]. 计算机研究与发展，2006，43(suppl)：89-95

[8] Ren Feng-hui, et al. Adaptive conceding strategies for automated trading agents in dynamic, open markets [J]. Decision Support Systems, 2009：704-716

[9] Meckenzie A, Ball A S, Vire-dd S R. 生态学[M]. 北京：科学出版社，2000

[10] Miozzo M, Sgorbissa A, Zaccaria R. The Artificial Ecosystem: A Multi-agent Architecture[C]// DEAL. Lecture Notes in Computer Science, 2003, Vol. 2690. 2003：52-59

[11] Friedrich R. Simulation of aquatic food Web and species interactions by adaptive agents embodied with evolutionary computation: a conceptual framework[J]. Ecological Modeling, 2003, 170

[12] 程昱，等. 基于机器学习的自动协商决策模型[J]. 软件学报，2009，20(8)：2160-2169

[13] Potter M, de Jong K. Cooperative Coevolution: An architecture for evolving coadapted subcomponents[J]. Evolutionary Computation, 2000, 8(1)：1-29

[14] 吕艳萍，等. 自适应扩散混合变异机制微粒群算法[J]. 软件学报，2007，18(11)：2740-2751

[15] García-Pedrajas N, et al. A cooperative coevolu- tionary algorithm for instance selection for instance based learning[J]. Mach Learn, 2010, 78：381-420

[16] Rajagopalan P, et al. Emergence of Competitive and Cooperative Behavior using Coevolution[C]// Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation(GECCO'10). 2010