

建立本体的协作—挖掘方法^{*}

明 仲^{1,2}, 李师贤¹, 蔡树彬¹, 徐 晶¹

(1. 中山大学计算机科学系, 广东 广州 510275)

2. 深圳大学信息工程学院, 广东 深圳 518060)

摘 要: 提出建立本体的协作—挖掘方法, 即领域专家、知识工程师、领域终端用户通过互联网, 协作建立本体。利用网络爬虫从语义网搜索并收集 RDF 文档或片段, 自动分析初步建立 RDFDB。自动分析领域专家、领域终端用户使用半形式化语言 RDFL 书写的实例文档和它们使用引导程序输入的记录, 完善 RDFDB。检验清洗 RDFDB 数据, 并设计本体挖掘算法挖掘产生初始领域本体。挖掘算法使用红黑树建立索引, 最坏时间复杂度为 $O(n^3 \log^2 n)$ 。验证、评估初始本体产生领域本体, 并产生文档说明。建立新本体时, 可合并 RDFDB 集成现有本体。在实验系统中, 应用该方法建立计算机硬件信息领域本体。实验结果表明该方法是可行和高效的。

关键词: 本体建立; 协作; 挖掘; RDFDB

中图分类号: TP311 **文献标识码:** A **文章编号:** 0529-6579 (2005) 03-0015-05

本体是领域内共享概念化模型的形式化、显式规格说明。自 20 世纪 90 年代起, 本体成为人工智能、计算语言学和数据库理论领域的研究热点之一。近年来, 本体在知识工程、知识表示、知识管理与组织、语言工程、数据库设计、信息建模、信息获取、信息集成、面向对象分析、智能体系统设计等领域中的研究也越来越受到重视, 并且成为语义网的核心技术之一。建立本体方法的研究是本体研究应用的基础, 具有重要的理论与应用价值。

建立本体的方法、原则, 有 Gruber 提出的 5 条指导原则; Gruninger 和 Fox 等人在 TOVE 项目提出的基于场景的方法; Uschold 和 King 等人在 Enterprise 项目提出的“Skeletal Methodology”方法; Fernandez 和 Gomez 等人在 CHEMICALS 项目提出的 METHONOLGY 方法; KBSI 在 IICE 项目提出的 IDEF5 方法等。但这些方法还不成熟、不统一。例如, TOVE 非常强调的“能力问题”, Enterprise 却认为可能不需要考虑。陆汝钤院士、金芝研究员等^[1,2]在 Eagle 等项目中提出的基于生物种群进化的方法、文 [3] 介绍的 TEMPPLET 方法, 则从复用本体的观点出发建立本体, 这可以降低本体建立的困难, 提高本体质量。

建立相对完整的领域本体, 不但需要捕获大量的概念、术语, 而且必须处理这些概念、术语的语

义冲突、二义性等问题。这使建立本体的工作成为既单调乏味, 又困难棘手的任务, 成为知识获取的瓶颈^[4]。以本体学习来建立本体的方法, 自动化程度较高, 近年来备受研究人员关注^[4-6]。但这类方法建立本体的质量受学习资料限制, 必须进一步研究扩展以提高初始本体的质量。在本体建立过程中, 研究者们普遍认为多方协作对提高本体质量具有重要作用。多个领域专家、知识工程师协作开发本体, 有利于提高本体的共享程度和质量。

本文综合上述方法并结合语义网的发展趋势, 提出建立本体的协作—挖掘方法。即领域专家、知识工程师、领域终端用户在互联网上协作, 共同建立 RDF 数据库 (RDFDB), 再挖掘 RDFDB 自动建立初始本体, 最后验证求精初始本体得到领域本体。

文章的第 1 节介绍系统框架和特色; 第 2 节介绍 RDFDB 的建立; 第 3 节介绍本体表示语言的选择和支持度、置信度的定义和挖掘算法; 第 4 节是实例分析; 最后是小结和讨论。

1 系统框架和特色

本文提出的建立本体的协作—挖掘方法, 如图 1 系统框架图所示, 通过收集处理网络爬虫从语义网搜索的 RDF 文档或片段 (XML、HTML 文档的处

* 收稿日期: 2004-06-17

基金项目: 国家自然科学基金资助项目 (60373084); 广东省自然科学基金资助项目 (04011304); 博士学科点基金资助项目 (20030558004)

作者简介: 明仲 (1967 年生), 男, 教授, 博士; 通讯联系人: 李师贤; E-mail: lnslx@zsu.edu.cn

理尚待进一步实现)；自动分析处理领域专家、领域终端用户使用半形式化语言书写的实例文档和他们使用引导程序输入的记录，建立 RDFDB。检验清洗 RDFDB 数据，降低噪音，再挖掘产生初始本体。最后验证、评估初始本体产生领域本体，并产生文档说明。同时，维护 RDFDB，供建立新本体时合并 RDFDB 以集成现有本体。如图 2 所示，与其他本体建立方法比较，本文介绍方法具有如下 2 个显著特点：

1.1 领域终端用户的协作

领域资料是领域内相对成熟、静态的知识；领域专家具有领域的整体认识。通过分析领域资料，可以获取领域内的大部分概念、术语。同时，对于领域细节，领域终端用户比领域专家具有更清楚的认识。通过请求领域终端用户的协作，能获取更多概念、术语。

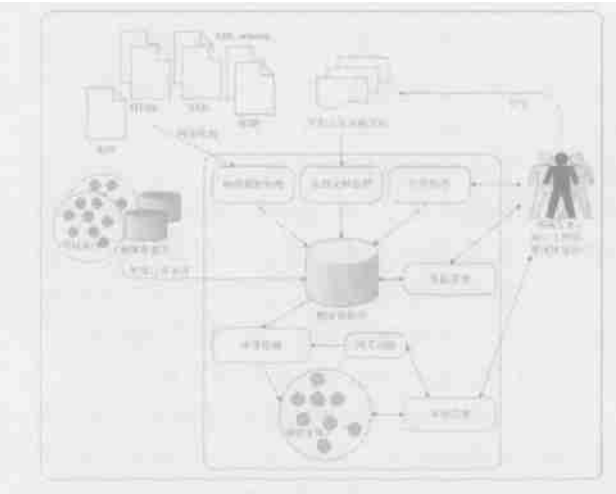


图 1 系统框架图

Fig 1 System architecture

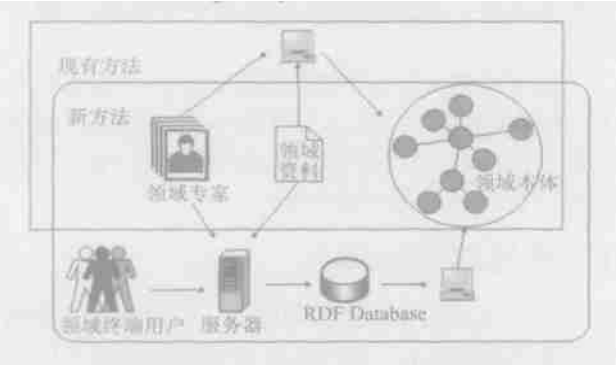


图 2 现有及本文建立本体方法对比

Fig 2 Existing ontology building approach vs collaborative-mining approach

1.2 RDFDB 的建立

RDF（Resource Description Framework）是语义网

两大基础技术之一，已经应用在许多领域中^[4,7]。本文使用爬虫程序搜索互连网的 RDF 片段，按深度优先从 www.semanticweb.org 开始分析超链接的网页，得到结果如表 1 所示：

表 1 爬虫程序运行结果

Tah 1 Result of crawler

运行时间	10 min	4~24 h
结果记录数	2 千多	2 万多
基本术语定义 (如 url、title 等)	85%	10%
网页资源描述 (如网页创建者是谁等)	10%	5%
概念术语定义 (如数码相机是相机子类等)	—	60%
非网页资源描述 (如谁的相机是数码相机等)	—	5%
无意义的例子片段及其他	5%	20%

本文将收集的资料转化为 RDF 表示，形成 RDFDB。结合语义网发展趋势可知，该 RDFDB 具有良好扩充前景。RDF 具有简单语义，更可以用来表示概念的内涵和外延^[8]。我们将可以非常方便地选取适当文档，将它们自动转换并存储进 RDFDB，使概念术语的定义更精确。概念术语的语义冲突，是阻碍本体集成、重用的主要因素之一。精确的概念术语定义，使 RDFDB 在本体集成、重用方面将具有良好的应用前景。

文 [8] 提出从 RDF annotation 进行本体学习的增量式方法。但它没有建立相应 RDFDB；也没有使用本体表示语言表示本体，使建立的本体难以共享；并且增量式算法最坏时间复杂度达到 $O(n^n)$ 。本文采用先建立 RDFDB，再挖掘 RDFDB 建立初始本体的方法。该方法建立本体的最坏时间复杂度是 $O(n^3 \log^2(n))$ ，并使用 OWL 表示本体。

2 建立 RDFDB

组织和定义项目是本体开发项目的首个活动。本体开发的目的和角度决定最终获得的本体。一般从多个角度出发建立领域本体，如建立人的顶层本体就可以从心理、社会、生理和病理 4 个角度^[9]进行。确定目的和角度后，如图 1 系统框架图所示，从 3 方面收集领域资料并按下列步骤建立 RDFDB。

2.1 领域专家、知识工程师协作建立原始领域模型

原始领域模型 DM（Domain Model）由 3 部分构成：领域关键概念集合 C （Concept）；概念的多继承分类层次结构 H_c （Concept Hierarchy）和概念的整体部分关系层次结构 WP_c （Concept Whole-Part

Hierarchy)。

知识工程师通过访问领域专家，或分析百科全书、字典等资料，或根据网络数据处理从语义网 RDF 片段产生的概念术语定义建立 DM。领域专家在 C/S 模式的引导程序辅助下，通过互联网协作建立并验证 DM。

2.2 领域终端用户验证、完善 DM 并书写实例文档

领域终端用户参与验证、完善 DM。每位领域终端用户在引导程序辅助下，验证并完善它们熟悉的部分领域模型。引导程序同时跟踪那些被反复修改的概念术语 MC。这些术语可能存在语义二义等问题，需要加以区别处理。

实例文档主要由 3 部分构成：领域概念实例和数据实例集合 I ；与领域概念 C 或数据类型（使用 XML Schema 定义）的实例化关系 T_i ；实例间关系 R_i 。实例文档是领域终端用户用半形式化的自然语言详细描述他们熟悉的事物、现象。如事物的组成、属性等。受文 [10] 启发，我们设计了一种半形式化语言 RDFL (Resource Description Framework Language)。实例文档处理程序自动分析使用 RDFL 书写的实例文档，产生 RDFDB 大部分数据。

2.3 设置同义词组并确认 DM

该阶段首先处理被反复修改的概念术语集 MC。即对二义性术语的使用达成一致，并设置同义词组。其次，确认 DM。对 H_c 和 W_{P_c} ，要求领域专家、领域终端用户验证确认。

2.4 采样测试

采用随机采样测试的方法，对 RDFDB 中的实例数据进行检验以计算数据的正确率。选取几个共享概念，计算它们出现的最小概率。置信度阈值一般略小于正确率，而支持度阈值略小于最小概率。

3 领域本体的建立

本体表示语言的选择直接影响到挖掘算法的设计。本体表示语言很多，有 OKBC、OCML、Flogic、CYCL、LOOM、OXL、SHOE、OML、OIL、DAML+OIL 等。NKI^[11] 则使用 COS、CAF 表示本体。OWL Web Ontology Language 是 W3C 推荐的本体语言，它可以明确表示概念和概念间关系的语义，便于机器理解并自动处理信息。OWL 与 RDF 存在共通点，我们采用 OWL 作为领域本体的表示语言。

OWL 包含 3 个子语言：OWL Lite、OWL DL 和 OWL Full。我们先挖掘产生 OWL Lite 的本体，再提供辅助工具建立 OWL DL 的本体。由于 OWL Full 并不保证计算的完备性，我们不提供额外工具支持产生 OWL Full 的本体。

3.1 支持度、置信度

支持度和置信度是关联规则挖掘两个重要的兴趣度度量。支持度是出现规则模式的任务相关元组所占的百分比，用以度量模式的潜在有用性；置信度是规则的蕴涵强度的估计，用以度量模式的有效性。

在挖掘产生的初始领域本体中，定义支持度是出现规则模式的元组占可能出现该规则模式元组的百分比。置信度是满足约束条件的规则模式的元组数占该规则模式元组的百分比。结合 OWL 语法，定义各种约束的支持度、置信度，设计算法挖掘概念间的关系及约束，产生初始领域本体。

3.2 挖掘算法

记号说明： Φ ：RDF 三元组组成的集合
 $subClassOf = http://www.w3.org/2000/01/rdf-schema \#subClassOf$
 $subPropertyOf = http://www.w3.org/2000/01/rdf-schema \#subPropertyOf$
 $type = http://purl.org/dc/elements/1.1/type$

定义 1: $Parents(x) = \{y \mid (x, subClassOf, y) \in \Phi \vee (x, subPropertyOf, y) \in \Phi\}$;
 $Ancestors(x) = Parents(x) \cup (Ancestors(y), \text{each } y \in Parents(x))$ 。

即 $Parents(x)$ 是 H_c 声明的 x 的超类的集合， $Ancestors(x)$ 是 x 所有超类（显式、隐式的）的集合。

定义 2: $ExplicitType(x) = \{y \mid (x, type, y) \in \Phi\}$;

$Type(x) = ExplicitType(x) \cup Ancestors(y), \text{each } y \in ExplicitType(x)$ 。

即 $ExplicitType(x)$ 是 T_l 声明的 x 的类型（实例化的概念）的集合， $Type(x)$ 是 x 继承的所有类型的集合。

定义 3: $ExplicitHasPart(x) = \{y \mid (x, hasPart, y) \in \Phi\}$;

$HasPart(x) = ExplicitHasPart(x) \cup HasPart(y), \text{each } y \in ExplicitHasPart(x)$ 。

即 $ExplicitHasPart(x)$ 是 W_{P_c} 声明的 x 包含的部分的集合， $HasPart(x)$ 是 x 包含的所有部分的集合。

$Type(x)$ 和 $HasPart(x)$ 和同义词组一起作为挖掘算法的背景知识。

挖掘算法：从 RDFDB 挖掘出初始领域本体。
输入：RDF 三元组集 Φ ，支持度和置信度阈值 $S、C$

输出：OWL Lite 表示的初始领域本体

1. 依次读取 Φ 中元组 (r, p, v) ，利用红黑树建立

(r, p, v) 和 (p, r, v) 的二次索引树
 (r, t) 和 (t, r) 的一次索引树，对每个 $t \in \text{Type}(r)$

累计属性 p, r 的类型，即 $\text{Type}(r)$ 元素的出现次数。

2. 对每个属性 p ，若支持度小于 C ，则删除 p ，否则

对二次索引树 (p, r, v) 检索 p 得到 p 的一次索引树 (r, v)

产生 (r, v) 的传递闭包并计算 p 是传递属性的支持度、置信度

计算属性 p 满足泛函属性约束、对称属性约束、domain、range 约束、是属性 p 相反属性、反泛函属性约束的置信度

若上述支持度、置信度的值不小于对应阈值，输出属性 p 对应约束的 OWL 语法定义。

3. 对每个类型 t ，若支持度小于 C ，则删除 t ，否则

对一次索引树 (t, r) 检索 t 得到 (r) 。
对 each $r \in (r)$ ，从 (r, p, v) 索引得到 (p, v)

累计类 t 具有属性 p 的元组个数
累计类 t 具有属性 p 的最小、最大势等于 1 的元组个数

计算类 t 具有属性 p 的任意大小的势。
累计类 t 具有属性 p 满足部分取值约束和所有取值约束的元组个数。

计算各种置信度并输出不小于阈值的类 t 的相关定义。

上述算法按 OWL Lite 的语法要求，输出支持度、置信度不小于阈值的概念术语的定义。

若捕获的概念术语数量太少，可适当降低支持度或置信度阈值；反之则增加阈值。我们实验设置的支持度和置信度阈值分别是 2% 和 70%。

由于红黑树插入、删除和查找节点均只需要 $\log(n)$ 时间，上述算法利用红黑树建立的二级索引进行插入、删除、查找节点的操作需要 $\log^2(n)$ 时间。而计算所有取值约束使用 3 个元组的连接，因此上述算法的最坏时间复杂度为 $n^3 \log^2(n)$ 。

4 实例分析

本文应用上述方法建立计算机硬件领域本体。建立本体的目的是为了提高计算机硬件信息检索的查全率和查准率，并用于自动设计符合用户要求的



图 3 辅助工具

Fig 3 CASE tool

计算机装配解决方案。34 位计算机专业学生为领域终端用户，由他们协作建立 RDFDB。实验分析 34 份实例文档，产生 4270 条 RDF 陈述。辅助工具环境使用情况如图 3 所示。再运行挖掘算法，得到初始领域本体。最后验证求精，建立领域本体。建立该本体共有领域专家 1 人，知识工程师 2 人，领域终端用户 34 人参与，共耗时 20 天完成，详细耗时情况见表 2。共捕获本体概念术语 1185 个。领域专家对挖掘得到的初始本体的概念术语的评估见表 3。

表 2 时间分布情况

Tab 2 Cost-time list

阶段名称	耗时（天）	参与人员 (括号内是人数)
建立 DM	2	DE (1)、KE (2)
实例文档写作、收集	7	DEU (34)
设同义词组及确认 DM	2	DE(1)、KE(1)、DEU(34)
数据采样测试及清理	2	DE (1)、KE (2)
挖掘产生初始领域本体	0	KE (1)
验证求精初始领域本体	4	DE (1)、KE (1)
文档化	3	KE (1)

注：DE= Domain Expert KE= Knowledge Engineer DEU= Domain Ended User

表 3 领域专家对初始领域本体术语评估意见

Tab 3 Evaluation of primitive domain ontology by domain expert

个数	百分比/%	熟悉程度	认可程度	访问 KE 可得
785	66.2	非常熟悉	赞同	可以
161	13.6	熟悉	赞同	可以
136	11.5	有印象	可以接受	大部分可以
68	5.7	比较陌生	勉强接受	小部分可以
21	1.8	陌生	难以确定	不可以
14	1.2	—	错误的	不可以

5 结 语

本体建立的困难严重阻碍本体研究应用的高速发展。本文综合国内外本体建立方法,并结合语义网发展趋势,提出了一种请求领域终端用户协作建立 RDFDB,再挖掘 RDFDB 建立初始领域本体,最后求精验证初始领域本体形成领域本体的方法。该方法要求领域终端用户协作开发本体,使本体的可预期质量更高。伴随语义网的发展将涌现的 RDF 文档、片段,将使 RDFDB 的建立更加简便容易,使该方法具有良好的发展前景。由于使用内涵、外延定义概念术语,使概念术语的表示更精确,有利于以后本体的集成、二次开发。我们采用该方法指导建立计算机硬件信息领域本体的过程表明,该方法是可行和高效的。

参考文献:

- [1] CHEN G, LU R Q, JIN Z. Constructing virtual domain ontologies based on domain knowledge reuse[J]. Journal of Software 2003, 14(3): 350—355.
- [2] LU R Q, JIN Z. Domain Modeling—Based Software Engineering[M]. Boston: Kluwer Academic Publishers, 2000.
- [3] YING D. TEMPLET: A New Method for Domain—Specific Ontology Design[J]. EDCIS 2002, LNCS 2480[C]. 2002: 90—103.
- [4] ALEXANDER M, STEFFEN S. Ontology Learning for the Semantic Web[J]. IEEE Intelligent Systems, 2001(3, 4): 72—79.
- [5] CHRISTIAN B, ALFONSO V. Automatic Ontology Construction from the Literature[J]. Genome Informatics 2002(13): 201—213.
- [6] HOLGER W, UBBO V, THORSTEN S. Ontology construction—An iterative and dynamic task[C]. The 15th International FLAIRS Conference, Pensacola, Florida, May 14—16, 2002.
- [7] ALEXANDER M, STEFFEN S. Ontology Learning[A]. STEFFEN S, RUDI S(Eds.): Handbook on Ontologies[C]. Springer 2004: 173—190.
- [8] ALEXANDRE D, CATHERINE F Z, ROSE D. Learning Ontologies from RDF annotations[C]. Proceeding of the IJCAT'01 Workshop on Ontology Learning (OL—2001), Seattle, USA, Aug. 2001.
- [9] TIAN W, GU F, CAO C G. Designing a Top—Level Ontology of Human Beings: A Multi-Perspective Approach[J]. Journal of Computer Science and technology, 2002, 17(5): 636—656.
- [10] CAO C G. Extracting and Sharing Knowledge from Medical Texts[J]. Journal of Computer Science and Technology, 2002, 17(3): 294—303.
- [11] CAO C G, FENG Q Z. Progress in the Development of National Knowledge Infrastructure[J]. Journal of Computer Science and Technology, 2002, 17(5): 522—534.

A Collaborative Mining Approach to Building Ontology

MING Zhong^{1,2}, LI Shi xian¹, CAI Shu bin¹, XU Jin¹

(1. Department of Computer Science, Sun Yat-Sen University, Guangzhou 510275, China;

2. Faculty of Information Engineering, Shenzhen University, Shenzhen 518060, China)

Abstract: A collaborative mining approach to building ontology is proposed. DEs (Domain Experts), KEs (Knowledge Engineers) and DEUs (Domain End Users) work on internet corporately in order to build ontology. Documents fragments of RDF collected by Crawler serve as primitive resources. Instance documents written by DEs/DEUs in Resource Description Framework Language, which is a pseudo formal language to describe instances, act as complementary resources. And records input using Wizard are optional resources. All these resources are analyzed automatically to establish the RDFDB. Data in RDFDB are refined before primitive domain ontology is mined. The worst time complexity of ontology mining algorithm is $O(n^3 \log^2 n)$, because its index service is based on Red Black tree. Primitive ontology evolves into domain ontology after verification and evaluation. Explanatory documents are generated automatically. When building new ontology, RDFDB of existing ontologies can be integrated into RDFDB of the new one. The approach is adopted in building computer hardware ontology. The experiment results show that this approach is feasible and efficient.

Key words: ontology building; collaborative; mining; RDFDB