从关系数据库学习 OWL 本体的方法

王 琦

(南京工程学院 计算机工程学院,江苏 南京 211167)

摘 要:提出了一种从关系数据库半自动学习 OWL 本体的方法。该方法在形式化表示关系数据库模式和 OWL 本体的基础上,遵循从关系数据库模式到 OWL 本体的一组通用映射方法和规则,并基于Java 2 平台实现了原型工具 OntoLearner。利用 OntoLearner 进行的典型案例研究表明了该方法的有效性。

关键词:本体学习;关系数据库模式;本体工程;OWL;语义 Web

中图分类号:TP311 **文献标识码**:A **文章编号**:1674-7720(2010)22-0058-04

Approach for learning OWL ontologies from relational databases

WANG Qi

(College of Computer Engineering, Nanjing Institute of Technology, Nanjing 211167, China)

Abstract: This paper presents a semi-automatic approach for learning OWL ontologies from relational database. Based on the formal definitions of relational database schema and OWL ontology, this approach follows a set of universal mapping methods and rules from a relational database schema to an OWL ontology, and implements a prototype tool OntoLearner based on Java 2 platform. A typical case study with OntoLearner validated the effectiveness of the proposed learning approach.

Key words: ontology learning; relational database schema; ontological engineering; OWL; semantic Web

本体是语义 Web 的关键使能技术,使用现有本体编辑器手工开发本体是一项冗长而繁琐的工作,极易导致知识获取的瓶颈,所以本体学习[1]技术应运而生。它极大地简化了本体的构建,满足了语义 Web 对于快速简便构造本体的需求,在语义 Web 中起到了杠杆的作用。关系数据库是基于 Web 的数据密集型应用的主要信息源,数据库模式中隐含着领域知识。因此,从关系数据库学习出的 OWL 本体更适合于数据密集型 Web 应用的需要。本文在形式化表示关系数据库模式和 OWL 本体的基础上,介绍了一种从关系数据库半自动学习 OWL 本体的方法,设计出一套通用的映射方法和规则,并基于 Java 2 平台实现了原型工具 OntoLearner。

1 关系数据库模式

关系数据库模型是以集合论中的关系(relation)概念为基础发展起来的数据模型[2]。为了便于形式化描述本体学习方法中的映射规则,这里给出了关系数据库模式的形式化定义(符合 3NF)。

定义 1:一个关系数据库模式 $S=(L, pkey, unique, notnull, fkey, subof, fdependency)$ 是七元组,其中:

(1)名集 $L=E\cup R\cup D$ 是一个有限集,由两两不相交的集合组成:

①一个实体关系(entity relation)名的集合 E;

②一个联系关系(relationship relation)名的集合 R;

③一个数据类型(data type)名的集合 D,每个数据类型名是 RDBMS 预定义的数据类型名。

(2)$\forall T\in E\cup R$,T 有一个非空的属性集合 $att(T)$,且每个属性 $A\in att(T)$ 有一个相关的预定义数据类型 $type(A)\in D$ 作为它的取值范围,其中 type(*)表示“*”的预定义数据类型。

(3)$\forall T\in E\cup R$,T 的属性集 $att(T)$ 中所有属性的一次取值的组合称为 T 的一个元组,其中,每个属性取值(称属性值)称为相应属性在此元组中的一个实例(instance),在某个时刻 T 的所有元组所组成的集合 $tup(T)$ 称为 T 的一个实例。

(4)$\forall T\in E\cup R$,T 有且仅有一个唯一决定其元组的属性或属性组称为 T 的主键 $pkey(T)$:要么 $pkey(T)$ 只包含一个属性(称 $pkey(T)$ 为单主键,此时 T 是实体关系),要么 $pkey(T)$ 包含多个属性(称 $pkey(T)$ 为复合主键,此

时T是联系关系)。

(5)$\forall T \in E \cup R$,若存在属性$A \in att(T)$,且A的所有元组在T中取值唯一,则称A为T的唯一属性,可表示为布尔函数 unique(A)=True;否则 unique(A)=False。

(6)$\forall T \in E \cup R$,若存在属性$A \in att(T)$,且A的所有元组在T中取值非空,则称A为T的非空属性,可表示为布尔函数 notnull(A)=True;否则 notnull(A)=False。

(7)$\forall T \in E \cup R$,T中一个引用其他实体关系$G \in E$主键$pkey(G)$的属性称为T的外键$fkey(T,G)$,满足:$fkey(T,G) \subseteq att(T)$且 value($fkey(T,G)$)$\subseteq$value($pkey(G)$)$\cup${null},其中 value(*)表示"*"的值域,"null"表示空值。T可能有$0 \cdot n(n \geqslant 0)$个外键。

(8)对$\forall T \in E$,若T引用$G \in E$主键的外键的同时又作为T的主键(即$fkey(T,G)=pkey(G) \in att(T)$),则称$T$和$G$之间存在"子/超关系",$T$为$G$的子实体关系,$G$为$T$的超实体关系,此时可表示为布尔函数$subof(T,G)$=Ture;否则$subof(T,G)$=False。若干个连续的子/超实体关系对构成关系数据库模式中的一个子/超实体关系层次。

(9)$\forall T \in E \cup R, \alpha, \beta \subseteq att(T), \alpha \neq \beta$,对$\forall t_1, t_2 \in tup(T)$,若有$t_1[\alpha]=t_2[\alpha]$,则必有$t_1[\beta]=t_2[\beta]$,则称$\alpha$和$\beta$之间存在函数依赖关系,$\beta$函数依赖于$\alpha$(或$\alpha$函数决定$\beta$),$\alpha$称为函数依赖的决定子,$\beta$称为函数依赖的被决定子,此时可表示为布尔函数 fdependency(α,β)=True(简记为 fdependency(α,β)),否则 fdependency(α,β)=False。T中所有函数依赖关系组成的集合称为T的函数依赖集,记为F_T。

定义2:$\forall T \in E \cup R$,α、$\beta \subseteq att(T)$,$\alpha \neq \beta$,$\beta \not\subset \alpha$,$\forall$ fdependency(α,β)$\in F_T$,满足下列条件之一:(1)α是超键;(2)β是主属性,则此关系T属于第三范式。若关系数据库模式中的所有关系均属于第三范式,则称其为规范化至第三范式(3NF)的关系数据库模式。

2 OWL本体

本体是一组描述某领域内概念及其属性以及概念间关系的词汇和公理的集合。W3C在2004年2月发布了标准化的Web本体语言OWL[3]。这里给出OWL DL本体的形式化定义。

定义3:一个OWL DL本体$O=(Cept, Axiom)$是二元组,其中:

(1)标识符集$Cept=CID \cup DPID \cup OPID \cup DTID$是一个有限集,由两两不相交的集合组成:

①一个类(class)标识符集CID;

②一个数据类型属性(datatype property)标识符集$DPID$;

③一个对象属性(object property)标识符集$OPID$;

④一个数据类型(data type)标识符集$DTID$,每个数据类型标识符是OWL本体中使用的预定义XML Schema数据类型标识符。

(2)公理集$Axiom=CAxiom \cup PAxiom$是一个有限集,由两两不相交的集合组成:

①一个类公理(class axiom)集$CAxiom$,包含本体中定义的所有类公理;

②一个属性公理(property axiom)集$PAxiom$,包含本体中定义的所有属性公理。

3 从关系数据库学习OWL本体的方法

从关系数据库学习OWL本体的可行性基于以下事实:运用数据库逆向工程方法可从关系数据库模式中提取ER模式[4];ER模式可语义保持地转换成OWL本体[5-6]。因此,本文针对现有本体学习方法和工具的不足,提出了一种从关系数据库学习OWL本体的方法,该方法分为关系数据库的逆向工程和从关系数据库模式到OWL本体的映射两部分。

3.1 关系数据库的逆向工程

数据库逆向工程DBRE(Database Reverse Engineering)指的是从物理数据库恢复数据库逻辑和概念模式,一般分成两个互相独立的阶段:数据结构的提取和概念化。在研究和分析数据库逆向工程现有理论和方法的基础上,本文制定了一套适合本体学习工程环境的较完备的逆向工程方法。

3.2 从关系数据库模式到OWL本体的映射

为了形式化表示从关系数据库模式到OWL本体的映射规则,需要预先定义以下辅助函数:

(1)$IS(x)$:布尔函数。若x(表达式)成立,则$IS(x)$=True;否则$IS(x)$=False。

(2)$idMap(ID)$:将关系数据库模式中的关系名和属性名映射为OWL本体中的同名标识符。即若ID是关系数据库模式中的关系名和属性名,则$idMap(ID)=ID \in CID \cup DPID \cup OPID$。

(3)$dtMap(DT)$:将关系数据库模式中的数据类型名映射为OWL本体中使用的数据类型(XML Schema数据类型)标识符。即若DT是关系数据库模式中的数据类型名,则$dtMap(DT)=DType \in DTID$。

3.2.1 属性公理的生成规则

规则1 将关系数据库模式中关系的非外键属性及其相应的预定义数据类型映射为OWL本体中以关系对应类为定义域的数据类型属性及其相应的预定义XML Schema数据类型。形式化表示为:

$\forall T, G \in E \cup R \wedge \forall A \in att(T) \wedge \neg IS(A=fkey(T,G)) \rightarrow$ DatatypeProperty($idMap(T_A)$domain($idMap(T)$)range($dtMap$(type(A))))。

规则2 将关系数据库模式中关系T引用实体关系G主键的外键(且不是T的主键也不属于T的主键)作如下映射:生成一个新的OWL本体中的类(联系类),一对分别以T对应类和联系类为定义域的互逆的对象属性,一对分别以G对应类和联系类为定义域的互逆的对象属性。形式化表示为:

$\forall T \in E \cup R \wedge \exists G \in E \wedge \forall A \in att(T) \wedge \neg IS(A=pkey$

$(T)) \wedge \neg IS(A \in pkey(T)) \wedge IS(B=pkey(G)) \wedge IS(A=fkey(T,G))$ →ObjectProperty(*idMap*(*T_A*) domain(*idMap*(*T*)) range(*dtMap*(*T_G*))), ObjectProperty(*idMap*(*G_B*) domain(*idMap*(*G*)) range(*idMap*(*T_G*))), ObjectProperty(*idMap*(*T_G_A*) domain(*idMap*(*T_G*)) range(*idMap*(*T*)) inverseOf(*idMap*(*T_A*))), ObjectProperty(*idMap*(*T_G_B*) domain(*idMap*(*T_G*)) range(*idMap*(*G*)) inverseOf(*idMap*(*G_B*)))。

规则 3 将关系数据库模式中关系 *T* 引用实体关系 *G* 主键的外键(且同时又是 *T* 的主键或属于 *T* 的主键)映射为 OWL 本体中一对分别以 *T* 对应类和 *G* 对应类为定义域的互逆的对象属性。形式化表示为:

$\forall T \in E \cup R \wedge \exists G \in E \wedge \forall A \in att(T) \wedge (IS(A=pkey(T)) \vee IS(A \in pkey(T))) \wedge IS(B=pkey(G)) \wedge IS(A=fkey(T,G))$ →ObjectProperty(*idMap*(*T_A*) domain(*idMap*(*T*)) range(*dtMap*(*G*))), ObjectProperty(*idMap*(*G_B*) domain(*idMap*(*G*)) range(*idMap*(*T_G*)) inverseOf(*idMap*(*T_A*)))。

规则 4 将关系数据库模式中实体关系的主键(且非外键)映射的 OWL 本体的数据类型属性声明为函数属性。形式化表示为:

$\forall T, G \in E \wedge \forall A \in att(T) \wedge IS(A=pkey(T)) \wedge \neg IS(A \in fkey(T,G))$→DatatypeProperty(*idMap*(*T_A*) Functional)。

规则 5 将关系数据库模式中关系取值唯一的非外键属性映射的 OWL 本体的数据类型属性声明为函数属性。形式化表示为:

$\forall T, G \in E \cup R \wedge \forall A \in att(T) \wedge \neg IS(A=fkey(T,G)) \wedge unique(A)$→DatatypeProperty(*idMap*(*T_A*) Functional)。

3.2.2 类公理的生成规则

规则 6 将关系数据库模式中关系的非外键属性映射为 OWL 本体中的类公理。形式化表示为:

$\forall T, G \in E \cup R \wedge \forall A \in att(T) \wedge \neg IS(A=fkey(T,G))$→Class(*idMap*(*T*) partial restriction(*idMap*(*T_A*) allValueFrom(*dtMap*(type(*A*))) minCardinality(0) maxCardinality(1)))。

规则 7 将关系数据库模式中关系 *T* 引用实体关系 *G* 主键的外键(且不是 *T* 的主键也不属于 *T* 的主键)映射为 OWL 本体中的类公理。形式化表示为:

$\forall T \in E \cup R \wedge \exists G \in E \wedge \forall A \in att(T) \wedge \neg IS(A=pkey(T)) \wedge \neg IS(A \in pkey(T)) \wedge IS(B=pkey(G)) \wedge IS(A=fkey(T,G))$ →Class(*idMap*(*T*) partial restriction(*idMap*(*T_A*) allValueFrom(*idMap*(*T_G*)))), Class(*idMap*(*G*) partial restriction(*idMap*(*G_B*)) allValueFrom(*idMap*(*T_G*)))), Class(*idMap*(*T_G*) partial restriction(*idMap*(*T_G_A*)) allValueFrom(*idMap*(*T*)))), Class(*idMap*(*T_G*) partial restriction(*idMap*(*T_G_B*)) allValueFrom(*idMap*(*G*))))。

规则 8 将关系数据库模式中关系 *T* 引用实体关系 *G* 主键的外键(且同时又是 *T* 的主键或属于 *T* 的主键)映射为 OWL 本体中的类公理。形式化表示为:

$\forall T \in E \cup R \wedge \exists G \in E \wedge \forall A \in att(T) \wedge (IS(A=pkey(T)) \vee IS(A \in pkey(T))) \wedge IS(B=pkey(G)) \wedge IS(A=fkey(T,G))$ →Class(*idMap*(*T*) partial restriction(*idMap*(*T_A*) allValueFrom(*idMap*(*G*)))), Class(*idMap*(*G*) partial restriction(*idMap*(*G_B*) allValueFrom(*idMap*(*T*))))。

规则 9 将关系数据库模式中实体关系的主键属性(且非外键属性)映射为 OWL 本体中的类公理。形式化表示为:

$\forall T, G \in E \wedge \forall A \in att(T) \wedge IS(A=pkey(T)) \wedge \neg IS(A=fkey(T,G))$ →Class(*idMap*(*T*) partial restriction(*idMap*(*T_A*) Cardinality(1)))。

规则 10 将关系数据库模式中关系取值不为空的属性(且非外键属性)映射为 OWL 本体中的类公理。形式化表示为:

$\forall T, G \in E \cup R \wedge \forall A \in att(T) \wedge \neg IS(A=fkey(T,G)) \wedge notnull(A)$→Class(*idMap*(*T*) partial restriction(*idMap*(*T_A*) Cardinality(1)))。

3.2.3 类关系的启发式规则

规则 11 如果两个实体关系的主键之间存在包含依赖关系,则这两个关系对应的 OWL 本体中的两个类之间存在父类和子类的关系。形式化表示为:

$\forall T, G \in E \wedge$ idependency(T,G)→SubClassOf(*idMap*(*T*) *idMap*(*G*))。

规则 12 关系数据库模式中的每个联系关系映射的类与 OWL 本体中的其他类之间是互不相交的。形式化表示为:

$\forall T \in R \wedge \forall W \in CID \wedge idMap(T) \neq W$→DisjointClassers(*idMap*(*T*) *W*)。

规则 13 关系数据库模式中关系 *T* 引用实体关系 *G* 主键的外键(且不是 *T* 的主键也不属于 *T* 的主键))映射生成的类(联系类)与 OWL 本体中的其他类之间是互不相交的。形式化表示为:

$\forall T \in E \cup R \wedge \exists G \in E \wedge \forall A \in att(T) \wedge \neg IS(A=pkey(T)) \wedge \neg IS(A \in pkey(T)) \wedge IS(A \in fkey(T,G)) \wedge \forall W \in CID \wedge idMap(T\_G) \neq W$→DisjointClassers(*idMap*(*T_G*) *W*)。

3.3 OntoLearner 设计思想

本体学习工具 OntoLearner 的体系结构如图 1 所示。基于 OntoLearner,本体学习的过程由下面三个子过程组成:

(1)关系数据库的逆向工程。使用逆向工程的方法从关系数据库中获取规范化至 3NF 的数据库模式信息,明确数据源的语义;

(2)从关系数据库模式到 OWL 本体的映射。以提取出的关系数据库模式(3NF)作为输入,按照一组通用的启发式规则实现从关系数据库模式到 OWL 本体的映射,并以可视化的方式显示;

(3)利用现有本体工程工具对生成的 OWL 本体进行精炼、评估和验证。

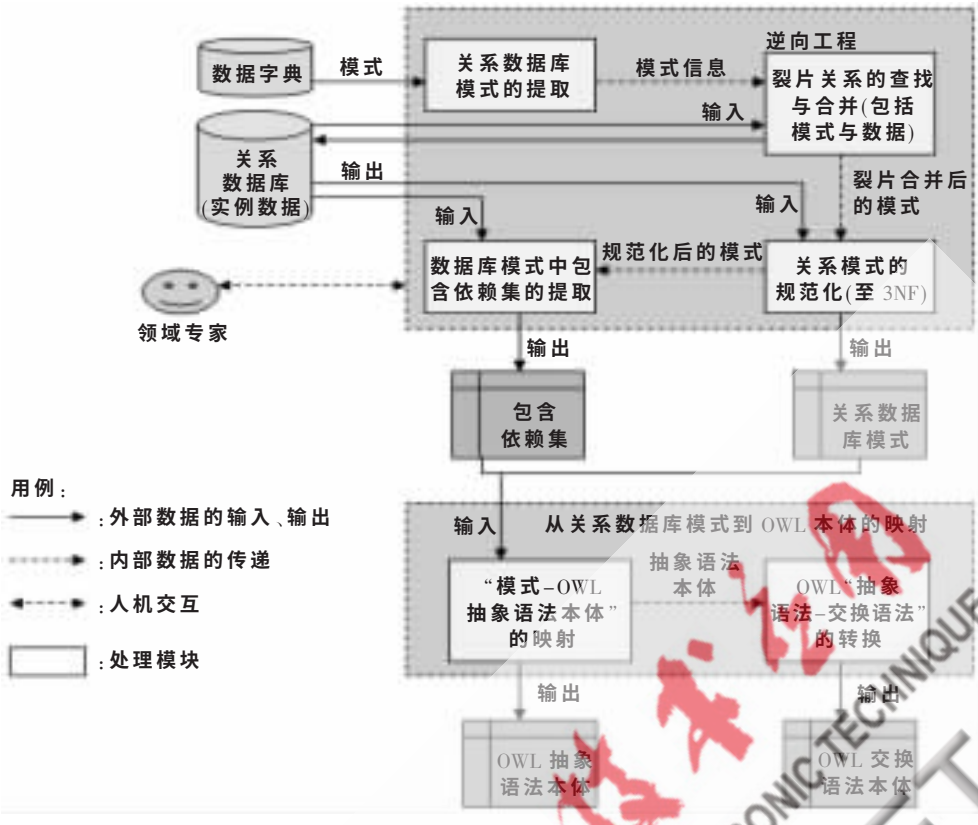

图 1 OntoLearner 体系结构

4 实例

使用 MS SQL Server 2000 创建一个包含大学基本信息情况的关系数据库 University，并插入实例数据。利用 OntoLearner 对该数据库进行本体学习，运行结果如图2 所示。

图 2 本体学习的源模式与目标模式的信息图

语义 Web 研究的重点就是如何把信息表示为计算机能够理解和处理的形式，即带有语义。本体在创建这种机器可理解和处理的 Web 内容中扮演着关键的角色。由于关系数据库是本体学习重要的知识源，所以研究从关系数据库学习 OWL 本体的方法，对数据密集型 Web 站点向语义 Web 迁移、动态 Web 页语义标注、构建新一代信息管理基础结构等均具有重要的现实意义。

参考文献

[1] STAAB S, MAEDCHE A. Ontology learning for the semantic Web[J/OL]. IEEE Intelligent Systems, 2001, 16(2): 72-79.

[2] 王能斌. 数据库系统教程(上册)[M].北京: 电子工业出版社, 2002:22-238.

[3] MICHAEL K S, CHRIS W, DEBORAH L.OWL Web ontology language guide(W3C Recommendation)[J/OL].(2004-02-10).http://www.w3.org/TR/2004/REC-owl-guide-20040210/.

[4] CHIANG R HL, BARRON T M, STOREY VC. Reverse engineering of relational databases: extraction of an EER model from a relational database [C].Data & Knowledge Engineering, 1994, 12(2): 107-142.

[5] XU Zhuo Ming, CAO Xiao, DONG Yi Sheng, et al.Formal approach and automated tool for translating ER schemata into OWL ontologies[J/OL].Advances in Knowledge Discovery and Data Mining, 2004(3056):464-475.

[6] 王琦.计算机与信息技术[M].安徽:安徽省计算机学会.2009.

(收稿日期:2010-06-17)

作者简介:

王琦，女，1980年生，硕士，主要研究方向：数据库、信息检索及语义 Web 技术。