

OWL 本体查询技术研究

倪 欢,许卓明

(河海大学计算机及信息工程学院,江苏 南京 210098)

摘要 通过基于本体工程 API 的本体查询技术的研究,给出了一种基于 Jena 的、支持 OWL Lite 本体查询的原型工具 OWL-QS 的设计与实现技术.该技术给出了本体解析、本体信息可视化、本体元素(类、属性和个体)查询等功能的处理流程和实现方法.应用研究表明:OWL-QS 实现的功能是有效的,基于当前开源本体 API(如 Jena)实现 OWL 本体查询是可行的.

关键词 本体查询;OWL;Jena;语义 Web

中图分类号 TP311 **文献标识码** A **文章编号** 1000-198X(2006)03-0333-04

语义 Web 是当前 Web 的一种扩展,它将信息表示为计算机能够理解和处理的形式,使得人和计算机能够更好地协同工作^[1].语义 Web 的实现需要 2 种关键技术的支持:一是语义 Web 内容的创建技术(例如基于本体的语义标注技术),本体(ontology)^[2]在其中发挥了关键作用;二是关于如何存取语义 Web 内容的相关技术(例如本体管理和查询技术),本体查询技术是它的基础^[3].OWL^①于 2004 年 2 月 10 日正式成为 W3C 推荐的标准 Web 本体语言,可以用来支持 Web 上本体的共享与发布.研究 OWL 本体查询技术、开发 OWL 本体查询工具,对研制各种基于 OWL 的本体工程工具以及开发各种基于本体的语义 Web 应用程序具有重要意义.

OWL 本体包含类、属性、个体及本体元数据(如本体作者和版本)等信息,各种构造子和公理用于描述类、属性和个体,以及它们之间的逻辑关系^[4](如类的包含关系、属性的等价关系、个体的类隶属关系等^[5]).在基于 OWL 的各种本体工具(如本体编辑器^[6]、映射工具^[7]、语义标注工具^[8]等)和实际应用中,均需要对本体信息进行查询.OWL 语言根据不同的应用目的、逻辑计算能力及约束的强弱依次分为 OWL Lite、OWL DL 以及 OWL Full 3 种子语言,其中 OWL Lite 易于实现,为用户提供了使用 OWL 最基本的功能子集,主要面向工具开发人员.

目前,一方面,W3C 对标准 OWL 本体查询语言(及其查询优化器)的研究还未开展,只集中于对 OWL 的基础——资源描述框架(RDF)的标准查询语言(SPARQL)^②的研究;另一方面,各种本体工程 API(如 Jena^③)正纷纷涌现,而基于这样的 API 可直接在本体工程工具中实现本体查询功能.

为了设计并实现一个基于 Jena API 的、支持 OWL Lite 本体查询的原型工具 OWL-QS,本文对基于本体工程 API 的本体查询技术进行了研究.

1 OWL-QS 体系结构与设计思想

OWL-QS(OWL Query System)的体系结构如图 1 所示.根据系统的设计目标,可将 OWL-QS 分为本体解析、可视化输出和查询处理 3 个功能模块.这 3 个功能模块彼此通过底层数据结构进行联系.OWL-QS 的主要设计思想如下:

“本体解析”部分的主要功能是获取本体信息,并将这些信息存储在预先定义好的数据结构中(主要包括类、属性、个体及本体元数据 4 种数据存储结构).本体解析功能是可视化输出与查询处理的基础,解析获取

收稿日期 2005-11-24

基金项目 国家自然科学基金资助项目(60573098)

作者简介 倪欢(1981—)男,江苏宿迁人,硕士研究生,主要从事语义 Web 技术研究.

① DEAN M, SCHREIBER G. OWL Web ontology language reference. 2004-02-10[2005-11-23]. <http://www.w3.org/TR/owl-ref/>.

② PRUD'HOMMEAUX E, SEABORNE A. SPARQL Query Language for RDF. 2005-07-21[2005-11-23]. <http://www.w3.org/TR/rdf-sparql-query/>.

③ Jena-A Semantic Web Framework for Java. 2004-02-08[2005-11-23]. <http://jena.sourceforge.net/>.

的本体信息供可视化输出和查询处理功能模块读取.本体解析的主要处理流程是:首先读取需要解析的本体文件,然后调用Jena API中用来支持与本体进行底层交互的方法和接口将本体中类、属性、个体及本体元数据的各种信息提取出来,并将这些信息存入相应的数据结构中.

“可视化输出”部分的主要功能是将本体信息处理后以直观形式输出.该输出形式主要采用树型结构表示类层次关系,并在类层次树上设置事件监听机制.这样,可以根据用户的鼠标选择操作,自动获取结点相关信息后可视化输出.这种做法使得用户能直接获取本体信息,而不需要关心具体的本体语言和本体内部的复杂结构.可视化输出功能模块的主要处理流程是:首先从本体信息存储数据结构中读取类层次关系信息,通过类层次树生成算法,将本体中类之间的层次关系以树型结构图形化显示,并在类层次树上添加鼠标事件监听机制,当用户点击类树中的某个节点时,从存储类信息的数据结构中读取该结点的相关信息并可视化输出.

“查询处理”部分的主要功能是分析用户查询请求,根据查询条件执行相应的算法和查询处理逻辑,最后将查询结果返回并通过查询交互界面输出.查询处理功能模块的主要处理流程是:首先基于本体解析和可视化的处理结果构建用户查询交互界面,包括类查询、属性查询和个体查询3种查询交互界面,每种查询交互界面支持用户输入相应的查询条件.其中,类查询支持根据用户输入类层次关系和类的属性来查询本体中满足条件的类,属性查询支持根据属性的值域和定义域、属性类型、属性的全局基约束以及属性的逻辑特征来查询本体中的属性,个体查询可根据用户输入的个体属性的取值来查询本体中的个体.

2 OWL-QS 实现技术

2.1 数据结构定义

OWL 本体文档由本体头(ontology header)、类公理(class axiom)、属性公理(property axiom)和个体事实(fact about individual)等部分组成,其中本体头中定义了本体的各种元数据,包括本体的名空间、本体引入和本体的版本信息等.依据OWL语言的标准规范及本体文档的组成结构,OWL-QS主要定义了类、属性、个体及本体元数据4种数据结构,这4种数据结构分别用来存储本体中类、属性、个体及本体元数据的相关信息.OWL-QS是基于Java语言编程来实现的,表1列出了每种数据结构定义的详细信息.

表 1 OWL-QS 数据结构定义

Table 1 Data structure definition of OWL-QS

数据结构类型	本体元数据数据结构	类数据结构和个体数据结构	属性数据结构
数据结构内容定义	<pre>class OntoMetaData{ String basemodelURI = null ; //本体 URI ArrayList modelnamespace ; //本体名空间 ArrayList importedmodelURI ; //引入本体 ArrayList labels , comments ; //标签及注释 ArrayList allDifferentInds ; //相异个体 }</pre>	<pre>class OntoClass { //类数据结构定义 String className , classURI //类名及 URI boolean isrestriction //标记类是否为匿名类 ArrayList labels , comments //标签及注释 ArrayList subclass , superclass //子类和超类 } class OntoIndividual { //个体数据结构定义 String indName , indID //个体名及 URI ArrayList sameAsList //等价个体 ArrayList differentFromList //相异个体 ArrayList indProperty //个体属性 ArrayList labels , comments //标签及注释 }</pre>	<pre>class OntoProperty { String proName , proURI //属性名及 URI boolean isobjectpro //标记属性是否为对象属性 ArrayList subproperty , superproperty ; //子属性及父属性 ArrayList inverseproperty , eqproperty ; //逆属性及等价属性 ArrayList range , domain //值域及定义域 Boolean isFunctionalPro , isInverseFunctionalPro ; //全局基约束 boolean isTransitivePro , isSymmetricPro //逻辑特征 }</pre>

2.2 本体解析和可视化的实现

OWL-QS 中的本体解析功能借助于 HP 公司语义 Web 研究组开发的、基于 Java 的、用于建立语义 Web 应用程序的开源应用编程接口 Jena v2.1 来实现.OWL-QS 的本体解析和可视化程序通过 Jena 提供的方法将本体中的类、属性、个体及元数据信息读取出来并存储于相应的数据结构中.解析功能模块所调用的主要 Jena 方法如表 2 所示.可视化功能模块从类数据结构中读取子类和超类的信息,通过一个类层次树生成算法构建

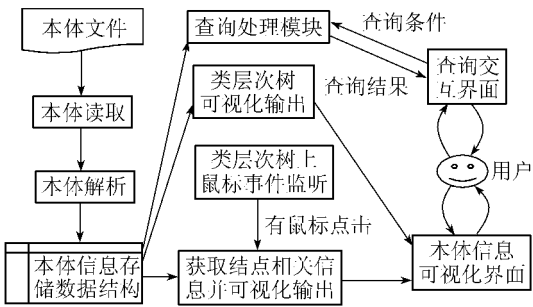


图 1 OWL-QS 体系结构与处理流程

Fig.1 Architecture and disposal procedure of OWL-QS

类树并输出.该算法的主要思想是,以 owl :Thing 类(所有类的超类)作为类树的根节点,通过 listSubClasses()方法的调用,获取 owl :Thing 的所有直接子类并将它们作为根节点的子节点添加到类树中,然后通过递归调用的方法,对 owl :Thing 的每个直接子类采用相同的处理过程,最后生成一棵完整的类层次树并输出.可视化模块通过 getDomain()和 listInstances()方法的调用,确定本体中属性及个体所关联的类,采用 Java 中的事件监听机制监听类层次树上的鼠标事件,根据用户节点选择操作显示类的相关信息.

2.3 本体查询的实现

类查询实现方法:首先处理类层次关系的查询,根据用户选择的类,调用 Jena API 中的 getSubClass()或 getSuperClass()方法,得到该类的子类或超类的集合,将结果存入动态数组中;然后处理类的属性查询,根据用户选择的属性,调用 getDomain()方法,得到该属性所关联的类,将结果存入另一动态数组中;最后将 2 个动态数组的交集存入结果集并在查询交互界面输出.

属性查询实现方法:根据用户输入的属性值域、定义域、属性的类型、全局基约束以及属性的逻辑特征,分别通过 getDomain(), getRange(), isDatatypeProperty(), isObjectProperty(), isFunctionalProperty(), isInverseFunctionalProperty(), isSymmetricProperty(), isTransitiveProperty()方法来查询本体中满足条件的属性,将查询结果存入结果集并在查询交互界面输出.

个体查询实现方法:根据用户输入的类、属性以及属性的取值,首先采用 isObjectProperty()和 isDatatypeProperty()方法判断个体属性的类型,然后根据个体属性的类型执行下一步的查询处理.如果是对象属性,则根据用户输入的类及属性名,调用 listInstances()方法,得到该类的所有个体后再用循环处理的方法依次判断每个个体对象属性的取值是否包含用户输入的个体,将查询结果存入结果集并输出;如果是数据类型属性,则根据用户输入的类及属性名,调用 listInstances()方法,得到该类的所有个体后再用循环处理方法依次判断每个个体的数据类型属性取值是否与用户输入的数值相等,将满足条件的个体存入结果集并输出.

3 实 例

本文通过对 person.owl 本体的解析、可视化和查询的运行结果(图 2 和图 3)来示意 OWL-QS 的功能实现情况.图 2 左部面板的树型结构用于显示该本体中定义的类及类层次关系,鼠标点击类树中某个节点之后(图 2 中为 Organization 类),右部面板显示该类的详细信息,包括类的所有属性和个体.图 3 是类查询交互界面,用户首先输入类的查询条件(图 3 中查询以 PhdStudent 类为子类且包含属性 first_name 的类),然后用户通过点击“search”按钮提交查询,系统执行查询处理操作,并将查询结果以列表形式输出.



图 2 本体信息可视化界面

Fig.2 Interface for ontology information visualization

表 2 本体解析调用的主要 Jena 方法

Table 2 Main Jena methods invoked by ontology parsing

方法名	描 述
getmodel()	获取本体模型
getclassList()	获取本体中所有类列表
listInstances()	获取指定类的所有个体
getDomain(), getRange()	获取属性定义域和值域
getSubClass(), getSuperClass()	获取子类和超类的方法



图 3 类查询交互界面

Fig.3 Interactive interface for class query

4 结 语

通过基于本体工程 API 的本体查询技术的研究 ,给出了一种基于 Jena 的、支持 OWL Lite 本体查询的原型工具 OWL-QS 的设计与实现技术 .OWL-QS 不仅实现了对 W3C 标准 Web 本体语言 OWL 表示的本体的查询 ,而且集成了本体解析和可视化功能 ,并可根据用户的选择 ,动态显示本体信息 .OWL-QS 的各部分功能模块均基于 Java 2 平台实现 ,具有可移植性强的特点 ,因此可以集成到其他本体工具和语义 Web 应用中 .SPARQL 是 W3C 最新发布的 RDF 本体查询语言 ,虽然该语言标准尚处于研究阶段 ,但已有相关工程 API 对 SPARQL 现有的规范标准提供相应的支持(如 Jena v2.3).随着 W3C 关于标准本体查询语言的研究和制定 ,对相应本体查询系统的研发将成为 Web 本体查询技术研究的主要方向 .

参考文献 :

[1] BERNERS-LEE T , HENDLER J , LASSILA O . The semantic Web[J] . Scientific American , 2001 , 284(5) : 34-43 .
[2] HENDLER J . Ontologies on the semantic Web[J] . IEEE Intelligent Systems , 2002 , 17(2) : 73-74 .
[3] 韩亚洪 ,刘永革 . 本体的查询与推理机制研究 [J] . 计算机工程与应用 , 2005 , 41(9) : 82-85 .
[4] 许卓明 ,王琦 . 一种从关系数据库学习 OWL 本体的方法 [J] . 河海大学学报 :自然科学版 , 2006 , 34(2) : 208-211 .
[5] 许卓明 ,黄永菁 . 从 OWL 本体到关系数据库模式的转换 [J] . 河海大学学报 :自然科学版 , 2006 , 34(1) : 95-99 .
[6] 李景 . 主要本体构建工具比较研究(上) [J] . 情报理论与实践 , 2006 , 29(1) : 109-111 .
[7] XU Zhuo-ming , CAO Xiao , DONG Yi-sheng , et al . Formal approach and automated tool for translating ER schemata into OWL ontologies [C] // DAI Hong-hua , SRIKANT R , ZHANG Cheng-qi . Advances in Knowledge Discovery and Data Mining : PAKDD2004 Proceedings . Berlin : Springer , 2004 : 464-475 .
[8] 凌海云 ,左志宏 ,陈兰 ,等 . 语义标注元数据及其抽取技术 [J] . 计算机应用研究 , 2004 , 21(7) : 147-149 .

Research on OWL ontology query technique

NI Huan , XU Zhuo-ming

(College of Computer & Information Engineering , Hohai University , Nanjing 210098 , China)

Abstract :The design idea and implementation technique of a Jena-based prototype tool OWL-QS for supporting OWL Lite ontology query were presented based on API of ontology query technique , with focus placed on the process flow and realization method for ontology parsing , ontology information visualization and ontology elements(classes , properties , and individuals) query . Case studies show that the functions of OWL-QS are effective , and technique for OWL ontology query based on current open-source ontology APIs , such as Jena , is feasible .

Key words :ontology query ; OWL ; Jena ; Semantic Web