

## 一个基于 SWRL 的查询系统框架\*

郑 鹏

(东南大学 计算机科学与工程学院,江苏 南京 210096)

**摘要:**基于本体的查询是当前的研究热点,但本体的语义表达能力局限于描述逻辑,不能表达一般形式的规则,因此人们提出了语义 Web 规则语言(SWRL)。分析了 SWRL 的研究现状及主要特点,然后基于 SWRL 解析器构建了一个基于 SWRL 的查询系统框架。

**关键词:**本体;规则;SWRL;查询系统

**中图分类号:**TP311.12 **文献标识码:**A **文章编号:**1671-5322(2007)02-0027-05

Web 自出现以来,为人们提供了一个极其丰富而有价值的信息资源库。然而,由于 Web 环境下的数据量日渐庞大,所以迅速并准确地定位有效信息变得越来越困难。用户在基于关键字方法搜索信息时经常会得到过多不相关结果或者根本就没有符合搜索要求的结果。其原因为同一个单词在不同的上下文中可能有不同的含义,或搜索使用的关键字在网页中根本没有出现。尽管可以通过包含更多的关键字或者同义词得到改进,但人们经常会搜到一些不是自己想要的信息。这些缺陷的主要原因是现在的 Web 并不是为计算机处理而设计的,计算机不能理解 Web 页面内容。为了使计算机能够处理网页内容, Tim Berners - Lee 提出了 Semantic Web 的构想,其核心技术之一是本体技术。

但是,本体的语义表达能力还是有局限性的,其表达能力局限于描述逻辑,不能表示一般形式的规则,更不能表示涉及时空关系的连续变化的事件流或是基于统计数据的不确定知识(容许矛盾冲突)。因此,以上基于本体的查询系统的推理能力仅仅局限于以类别为基础加上关联性的推理,而对于 If... Then... 这样很普通的关系则推理能力不足。例如,你可以用一阶逻辑来定义概念  $Uncle: Parent(?x, ?y) \wedge Brother(?y, ?z) \Rightarrow Uncle(?x, ?z)$ , 而 OWL 则不能定义这样

的关系。因此,我们相信规则的表示将在语义 Web 中发挥重要的作用,一方面,规则可以表达更多的知识,范围从隐含在人们头脑中的常识到公司的商业策略;另一方面,规则表达能力可以大大节省知识的存储空间,因为一些事实可以通过知识库中已有的事实和规则动态地推导出来,而不必事先将它们存储起来。近年来,很多研究致力于在 OWL 提供的本体层上结合不同类别的规则语言,以提供更强大的知识表示和推理能力,这些工作已经提供了一些扩展 OWL 语言的途径。2003 年 11 月 DARPA 组织提出了一套语义网规则语言 SWRL (Semantic Web Rule Language), 其目的就是为了驱使 Horn-like 规则可与 OWL 知识库产生结合。因此,本文的主要工作就是利用 SWRL 所提供的强大的语义表达能力,来设计与实现一个具有本体和规则推理能力的查询系统<sup>[1]</sup>。

本文首先介绍 SWRL 相关研究;然后提出了一个基于 SWRL 的查询系统框;最后分析了查询过程分析,并提出了查询系统实现框架。

### 1 相关研究现状

SWRL 相关研究主要包括 3 个方面:

第 1 是关于 SWRL 语言本身方面的研究。SWRL 是由 OWL 子语言 OWL DL 与 OWL Lite, 以

\* 收稿日期:2006-09-21

作者简介:郑 鹏(1976-),男,江苏东台市人,硕士研究生,主要研究方向为数据库挖掘。

及 Unary/Binary Datalog RuleML 为基础的规则描述语言,其目的是为了驱使 Horn-like 规则可与 OWL 知识库产生结合。SWRL 是由 RuleML 演变而来,一样是 XML Based 的规则格式,具有人机可读的优点,可较为清楚地表现规则之间的关系,此外 SWRL 可以引用本体中的元素来编辑规则,这是和 RuleML 的不同之处<sup>[1]</sup>。

第 2 是关于如何对 SWRL 进行推理解析方面的研究,这也是研究环节中最重要的一环,它将是构建相关应用系统的基础。SWRL 的规范没有限制应该如何来推理 SWRL 规则<sup>[2]</sup>。因此用户可以自由地选择多种类型的推理引擎来通过 SWRL 规则推理存储在 1 个 OWL 知识库中的知识,同样,也可以自由实现自己的编辑工具来创建 SWRL 规则。

当前国内外对于如何来解析 SWRL 规则的研究主要方向是将 SWRL 的编辑器与第 3 方的规则引擎(如 Jess、Algernon 和 SweetRules)加以整合以实现 SWRL 的推理解析。例如,文献[3]中提出了 1 个将运行于 Protégé OWL Plug-in 中的 SWRL 编辑器与 Jess 引擎整合以实现 SWRL 推理的方式。

最后是有关于基于 SWRL 的应用系统方面的研究。由于 SWRL 的规范还在进一步的发展当中,目前还没有该语言的正式版本,因此这一方面的研究还比较少。文献[4]中提出了一个基于 OWL DL 与 SWRL 的大学排课系统的设计与实现,但是该文虽然提出了一个可行的思路,却没有真正地解决大学排课中的资源冲突问题,没有能够充分体现 SWRL 强大的知识表示与推理能力。

本文的工作主要的思路就是借鉴以上这些已有的工作,特别是关于 SWRL 推理解析方面的工作,寻找相关的应用场景来建立 1 个基于 SWRL 的查询系统。

## 2 基于 SWRL 的查询系统框架

为使基于 SWRL 的查询系统可在 Intranet 上运行,本文提出的基于 SWRL 的查询系统框架是基于 C/S(客户端/服务器端)模式的。客户端中主要包含查询需求输入界面和查询结果输出界面,用户可输入查询条件和获得查询结果。系统的核心部分放在服务器端,其中包含 3 个层次:接口层、数据层和推理层。图 1 是基于 SWRL 的查接受用户从输入界面上提交的查询条件并启动相

询系统体系结构图。下面我们逐一对其中的功能模块作解释说明。

### 2.1 客户端

客户端包含查询需求输入界面和查询结果输出界面。

#### 2.1.1 查询需求输入界面

输入界面的设计决定了能生成什么样的查询需求。查询需求的表达方式主要有如下几种:一是以自然语言方式表示,最典型的例子是关键字的方式;二是以特定的查询语言表示,如 SQL、RQL、SPARQL 等;三是以可视化的方式表示,例如通过菜单选择查询条件的方式。这些方式对用户的要求也是不同的,前两种尤其是第 2 种方式要求用户对需要查询的领域信息了解得比较清楚,才能清晰地表达查询需求,而第 3 种方式则对用户要求较低。

本文设计查询需求输入界面的目标是使用户能够确切表达对真正所需信息的需求,同时尽量降低对用户的技术要求。由于本体能够清晰地表达领域内事物之间的内在联系,因此,为使用户更好地利用查询信息,本文采用本体引导查询需求输入界面的生成。

查询需求输入界面生成时应首先从服务器端载入领域本体的 Schema,将其解析为直观的树形结构,用户只需根据自己认知的要求浏览领域本体来确定相应的查询内容,而不需要主动提供关键字。例如,用户可以在这样的引导方式下查询属于某个类的所有实例信息。如果用户需要描述更复杂的查询要求,便可通过另一页面让用户选择查询条件与查询范围。例如,在大学排课系统中用户想要表达调课的需求,则需另一页面来引导用户生成查询条件。

本文在具体实现原型系统的查询需求输入时一方面是使用本体引导下的菜单选择方式,同时另一方面也充分利用了 Protégé 的可视化本体编辑界面来实现部分复杂查询需求的输入。

#### 2.1.2 查询结果输出界面

将查询结果以表格的形式显示给用户。

### 2.2 服务器端

服务器端包含 3 个层次:接口层、数据层和推理层。

#### 2.2.1 接口层

接口层的组件主要是查询控制器,其功能应是查询过程;将查询结果传给输出界面。关于

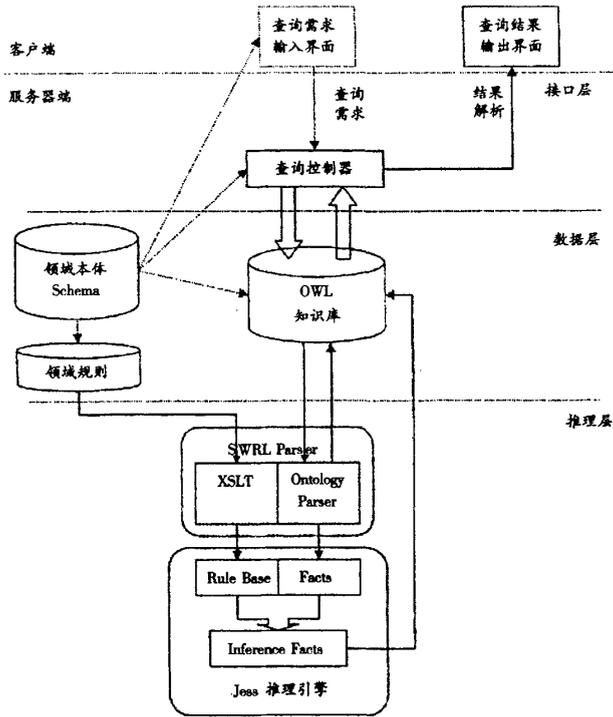


图 1 基于 SWRL 的查询系统体系结构图

Fig. 1 Architecture of query system based on SWRL

查询过程我们在下一节将会进行详细分析。

### 2.2.2 数据层

数据层包含领域本体 Schema、领域规则和 OWL 知识库 3 部分。其中前两部分是系统建立时的基础数据,对其的建立和修改可通过后台维护模块来实现,本文所建立的查询系统框架假设这两部分内容应由具体的领域专家来建立和维护,暂不考虑这方面的维护问题。OWL 知识库则是用户查询时的数据源,其中包含领域本体 Schema 的镜像和领域本体的实例信息。另外,领域本体 Schema 或领域规则的发生变动会影响到 OWL 知识库中的内容,因此,如果发生这样的变动则需要对 OWL 知识库重新初始化,这部分工作也应该通过后台维护模块来完成。

**领域 Schema:**领域本体的 Schema 是整个系统的概念基础,它提供了领域知识的类别层次结构,是定义领域规则的基础,是 OWL 知识库中信息的结构基础,同时也是引导查询生成的内容依据<sup>[5]</sup>。

**领域规则:**领域规则是在领域本体的基础上定义的,领域规则以 SWRL 语法描述,是进行基

于推理的查询的基础。

**OWL 知识库:**用 OWL 描述的本体知识,是本文查询系统的数据源,它不仅包含了实例化的领域本体本身,还包含了领域本体经过描述逻辑推理(DL reasoning)分类后的隐含信息,同时包含通过领域规则和领域本体经过 Jess 推理所获得的信息。

### 2.2.3 推理模块

推理模块包含 SWRL Parser 和 Jess 推理引擎两部分。

**SWRL Parser:**上一章所建立的 SWRL 解析器,其主要功能是将 OWL 知识和 SWRL 规则解析为 Jess 引擎所能接受的格式,同时其中的 DL Reasoner 也具有基于描述逻辑的推理能力可以对 OWL 知识库进行隐含信息挖掘以使 OWL 知识库中的信息完整<sup>[6]</sup>。

**Jess 推理引擎:**在启动基于推理的查询时,根据事实库中的已有事实,通过规则库中的规则推理出新的事实,并将新的事实经过格式转换后加入到 OWL 知识库中,以达到对 OWL 知识库内容的更新与扩充。

### 3 查询过程分析

查询过程可分为两种:简单查询和复杂查询。简单查询指的是不需要经过推理引擎的推理可以直接在 OWL 知识库中找到所需信息的查询;复杂查询指的是由于查询需求会要求改变 OWL 知识库中的内容而触发推理引擎对 OWL 知识库进行整理和更新的查询。下面我们根据以上基于 SWRL 的查询系统框架,分别对这两种查询进行分析。

#### 3.1 简单查询

经过查询控制器分析用户输入的查询条件后,如果用户的查询需求只是要查询 OWL 知识库中的已有信息,而不涉及到对 OWL 知识库中信息的修改,那么就无需启动推理模块进行推理,需要的工作只是在 OWL 知识库中的遍历匹配操作。例如在大学排课系统中,OWL 知识库中已有排好的课程信息,用户需要查询所有与某教师有关的课表信息,这时我们可以直接在 OWL 知识库中寻找属性中包含该教师名称的实例信息,然后将这些实例信息以表格的形式显示给用户。

#### 3.2 复杂查询

经过查询控制器分析用户输入的查询条件后,如果用户的查询需求需要对 OWL 知识库中的实例信息进行某些修改或限定,为了满足这些修改和限定,需要对知识库重新推理以发现可能的冲突,如果不出现冲突说明这些修改或限制可以满足。

例如,在大学排课系统中,OWL 知识库中已有排好的课程信息,假设某教师提交了这样的查询要求:能否将其星期二上午的某门课程调到星期五下午。此时查询需求可理解为用户对其课程时间上的限定,因此需要推理。

此外,有可能会因为用户查询需求因为限制过多而导致其本身就包含了冲突,此时将因为不能满足相关的 OWL 约束而无法启动推理或推出来的新事实无法写回 OWL 知识库中,系统应提示用户描述的查询需求有误。

### 4 查询系统实现框架

在建立本文的原型系统时,我们将使用一些

工具来快速地构建,这些工具主要是 Protégé、Racer 和 Protégé 的一些插件。因此,这些工具将分别对应上面查询系统结构的一个或几个功能模块。

Protégé 提供了可视化的本体编辑界面,用户可以方便的编辑本体的类、属性和实例,并且推理后的结果也可以通过相应的接口返回给 Protégé,因此我们可以使用 Protégé 的 OWL 插件来编辑本体以及做后期本体的后台维护工作。

推理引擎使用的是 JessTab,它是整合在 Protégé 中的 Jess 引擎插件,采用 JessTab 的主要原因是其对本体的支持比单纯得 Jess 引擎更好,除了 Jess 原本的功能外,还有专门针对与本体进行信息交互的一些应用编程接口(API)。

Ontology Parser 中的 DL Reasoner 我们则使用外挂的 Racer,它可以通过推理将本体重新分类并将重新分类后的本体通过相应的接口传回给 Protégé,然后可以通过 JessTab 中的 mapclass 指令和 facts 指令将经过重新分类后的本体知识映射成为 Jess 事实。

SWRL 规则的编辑与转换则使用 Protégé 的插件 SWRL Editor。

图 2 是本文的查询系统实现框架。

### 5 结束语

本文分析了当前基于本体的信息查询技术的优点与缺点,然后针对本体推理能力不足的缺陷,提出了一个基于 SWRL 的查询系统框架。SWRL 相比较本体而言一个非常重要的优势在于引入了规则,所以 SWRL 能够表达本体所不能表达的一般形式的规则,从而具有更强的推理能力。

本体与规则的结合是目前语义 Web 领域研究的热点,同时也未完全成熟。本文所基于的 SWRL 语言本身还处于规范制定阶段,目前还没有提供正式的版本。同时,当前已有学者研究通过使用一阶逻辑、模糊逻辑等手段来进一步扩展 SWRL 的语义表达与逻辑推理能力。本文主要是讨论 SWRL 的应用框架,属于探索性的工作,还有很多实际的验证工作需要研究。

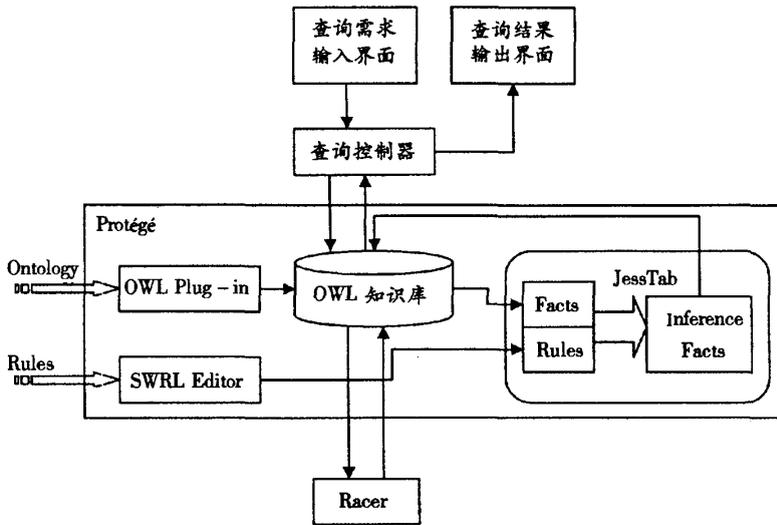


图2 查询系统实现框架图

Fig. 2 Application Architecture of query system

## 参考文献:

- [1] 廖明宏,程光明,吴翔虎. 一个 WWW 智能搜索引擎[J]. 计算机应用研究,2001(5):29-31.
- [2] 武成岗,焦文品,田启家,等. 基于本体论和多主体的信息检索服务器[J]. 计算机研究与发展,2001(6):641-647.
- [3] Rabarijaona A, Dieng R, Corby O. Building and searching an XML - based corporate memory[J]. IEEE Intelligent Systems, 2000,15(3):56-63.
- [4] Jerome Euzenat. Eight Questions about Semantic Web Annotations[J]. IEEE Intelligent SYSTEMS, 2002(1):55-62.
- [5] Michael Erdmann, Rudi Studer. How to structure and access XML documents with ontologies[J]. Data&Knowledge Engineering,2001, 36:317-335.
- [6] Martin O'Connor, Holger Knublauch, Samson Tu, etal. Stanford Medical Informatics[C]. Stanford; Stanford University School of Medicine, 2005.

## A Query System Framework Based on SWRL

ZHENG Peng

(School of Computer Science and Engineering, Southeast University, Jiangsu Nanjing 2210096, China)

**Abstract:** Research on ontology - based query has caused current research concern. Since the semantic expressive ability of ontology is limited to the Description Logic and unable to express the regulative rules, the Semantic Web Rule Language (SWRL) is proposed. The paper has first analysed the current research development of SWRL and its main characteristics. Then a query system framework based on SWRL is established.

**Keywords:** ontology; rule; SWRL; query system