摘要

信息时代到来已久，大量数据的产生使得数据筛选、数据分析、数据挖掘等成为信息时代大跨步发展前进的巨大障碍。在这日新月异的技术发展中，数据质量评价分析已成为上述技术商业化进程的最大绊脚石。本文利用Protege、Jena、Jess等工具，以数据质量评价为探究点展开探究。经过我对前辈研究成果的学习与思考，本文决定通过Protege用IDEF-5方法对数据质量本体进行建模，然后根据需求写出符合实际情况的SWRL规则，成功构建出可用于推理的数据质量本体。之后利用熵权法和层次分析法对于评价数据的计算，得出一个相对客观，又符合专家建议的综合评分，最后再通过Jess推理机经过逻辑推理获取到数据的质量等级，以此剔除数据系统中会降低我们数据分析结果的脏数据，在大多数情况下，都能提高大数据分析等技术的结果可靠性。

关键词：数据质量；本体；Jena；Jess；熵权法

Abstract

With the arrival of the information age, a great deal of data have made data screening, data analysis and data mining becoming a great obstacle to the development of the information age. In the rapid development of technology, data quality evaluation and analysis has become the biggest stumbling block in the process of commercialization of these technologies. This paper uses Protege, Jena, Jess and other tools to explore the data quality evaluation as the research point. After I study and Reflection on the research achievements of their predecessors, this paper determined by Protege using IDEF-5 modeling method for data quality ontology, then to write in line with the actual situation of the SWRL rule, is successfully constructed and can be used for data quality ontology reasoning. After using entropy method and AHP to calculate the evaluation data, draw a relatively objective, but also with the comprehensive score of expert advice, and then through the Jess inference engine through logical reasoning to obtain quality data, in order to eliminate data in the system will reduce our data dirty data analysis results, in most cases next, can improve the data analysis results of reliability technology.

**Keywords：Data Quality；ontology；Jena；Jess；Entropy weight method**

目录

[摘要 1](#_Toc729)

[Abstract 2](#_Toc5516)

[第1章 概述 3](#_Toc6113)

[1.1 选题背景及意义 4](#_Toc31947)

[1.2 国内外研究的现状 5](#_Toc30913)

[第2章 本体及相关技术 7](#_Toc31087)

[2.1 本体简介 7](#_Toc31655)

[2.2 OWL语言 7](#_Toc620)

[2.3 Protege本体建模工具 7](#_Toc14146)

[2.4 Apache Jena技术 8](#_Toc10492)

[2.5 Jess介绍 8](#_Toc8206)

[2.6 数据质量 9](#_Toc8006)

[2.6熵权法 9](#_Toc22724)

[2.7 Java 9](#_Toc6540)

[2.8 IntelliJ IDEA介绍 10](#_Toc5983)

[第3章 数据质量本体构建 11](#_Toc1427)

[3.1 本体的概念、关系以及约束公理 11](#_Toc19301)

[3.2 本体的构建准则 13](#_Toc12882)

[3.3 数据质量本体构建流程 13](#_Toc22352)

[3.4 数据质量本体类以及属性 15](#_Toc4665)

[3.5 数据质量规则对在本体中的描述 16](#_Toc32321)

[第4章 基于Jess的数据质量评价方法 18](#_Toc21645)

[4.1 基于SWRL的Jess推理 18](#_Toc28783)

[4.2 熵权法在本体中的设计实现 19](#_Toc13986)

[4.3 数据质量的Jess推理实现 22](#_Toc29681)

[第5章 系统实现 24](#_Toc13314)

[5.1 数据质量本体 24](#_Toc21700)

[5.2 熵权法与数据质量评价的实现 29](#_Toc13512)

[结论 32](#_Toc24145)

[参考文献 33](#_Toc5034)

第1章 概述

## 1.1 选题背景及意义

近些年来，计算机技术以及信息处理技术获得了跨越式的发展，也因此使企业、组织等产生的数据通过日积月累有一个庞大的数量。其中蕴含着巨大的信息量，如果能够及时的解析出其中的奥秘，经能够站在世界的颠端，引领世界的发展与走向。由此种种，标志着以数据分析为主要研究方向的大数据[1]时代粉墨登场。作为新时代的新希望，我们这些热血青年必然要紧跟世界发展的脚步，为未来、为世界、为祖国挥洒热血、肝脑涂地，不浪费一丝的青春。

要做出正确的决策，离不开数据架构中更加统一、精确的大数据，也就是说，所拥有的数据越准确、正确，分析结果就越加准确，也就可以据此作出正确的决策。数据是信息的载体[2]，高准确度、高有效性的数据是数据分体、数据挖掘等数据信息提取技术能够得到无误信息与结果的基础。因此，各企业所拥有数据的质量也成为各企业关注的焦点。

在大数据时代，经常会听到人们谈论关于拥有大量数据，却缺少有用信息的问题。其中，数据分析技术不完善是最重要的问题。除此，保有数据质量较低也是重要因素之一，这就致使空有大量数据，而不能产出应有的数据效益。

目前，对于有效、完善管理数据采集、录入等各个环节，仍然没有一个统一、标准化的解决方案。虽然人类从很久之前就在研究数据质量，涉及范围也非常广泛，比如人工智能、知识工程[3]、数据仓库、统计学等，而大家也越来越依赖于数据，越来越关心数据分析、挖掘的结果，但是诸如数据的错误录入、错误拼写、数据空缺、不同来源数据冲突等一直无法有效解决问题的存在，致使我们获取到的数据通常是脏数据[4]。脏数据的存在对于大数据分析与研究的影响，就可能因为错误的结果导致错误的决策，导致错失良机还是小事，但是若因此蒙受巨大损失，就显得满盘皆输了。由此可见，提高数据质量、去除脏数据是多么重要的需求。大数据发展到现在，对于这两个问题的解决方案很多，但终究没有比较完美成体系的解决方案。

随着近几年人工智能和知识工程的飞速发展，本体在众多领域也受到了广泛的关注。由于大数据具有大数据量、数据应用范围广泛的性质，而本体却有着共享与复用的性质[5]。，因此，我们顺理成章的考虑用本体的技术方法来解决大数据所存在的问题。

数据质量本体是一种领域本体[6]，通过本体进行的数据质量评价，能够独立于具体的应用和所在的环境。这样，在数据评价领域，数据质量本体[7]就具有超乎想象的公用性，能够使得前有方法中关系模型进行的数据评估所存在的不具适用性的通病。关系模型进行的数据评估都是应用于具体的某一种应用和具体所处的环境，并且在处理数据质量评价时，必须分别对其中的复杂约束问题和简单约束问题应用不同的解决方案，使得无端增加了评估算法的空间复杂度和时间复杂度。最麻烦的问题还在于无法共享，更无法适用于不同源的数据和不同应用的数据。

数据质量本体是对数据评估领域知识的建模[8]，是对数据评估领域知识的规范化模式。主要是抽取了数据质量的评估规则，并且对这些规则进行抽象，提取来构建数据质量本体。通过对数据评估领域知识的规范化，形式数据评估领域的评估标准，使得在数据评估领域有一套可以通用，共享的数据评估标准。

数据质量本体构建及应用研究是针对数据质量出现的问题，利用本体的思想和方法来构建数据质量本体[9]。通过数据的评估规则约束，例如非空约束，值域约束，逻辑依赖约束，存在性依赖约束，等值约束，等值一致性约束，逻辑一致性约束等，抽取出数据质量本体中的属性。利用数据质量本体，对数据库中的数据进行质量评估，选出其中的不合格数据。所以数据质量的评估对于数据的分析以及研究具有很大的现实意义，能够选出不合格的数据，分析数据质量的高低。

## 1.2 国内外研究的现状

关于本体研究已经持续很多年了，在前期，都是由人手工的创建所需的本体架构。自然了，这种方法[10]存在在着很多的缺点。在手工创建之前，需要对本体所要解决的问题和所在的领域有一定的了解，然后通过人工的发掘其中的类（即概念），并需要理清楚这些类之间的附属关系和作用层次，之后，才可以根据整理好的思路，运用诸如Protege等可以进行本体编辑的工具创建所需的本体，最后再对创建的本体进行评价，以更好的修正和完善该本体。

本体编辑工具归根结底也只是一个工具，只不过提供了一些便捷和友好的UI界面。而本体类的构思，理清明晰的类关系和层次，仍然是由对应领域专家来完成。伴随着对本体的研究逐渐趋于成熟，创建本体产生了半自动化的方案。它能利用已存在的资源去降低人在本体创建过程中的工作量。目前，创建本体的半自动化研究在国内外都取得了一定的成果。应用二维的方式来记录类和关系，这是对本体创建的一次重大突破。眼前，分析关系模式的语义信息来抽取本体构建时赖以基础的类和关系。综上所述，在没有一个比较完美的系统开发出来之前，应用领域本体的技术是数据质量评估的最好选择。

关于数据质量的研究，相对国内，国外要开始的早得多，并取得了不菲的成果。而国内借助于后起优势，奋赶直前，与国际的距离已越来越小。在信息爆炸的今天，对数据精确处理的需求一天比一天紧迫。在大量社会资源投入的情况下，数据质量的研究必将成指数型爆发，相信不许很久，数据质量评估与分析将成为各个企业找到企业问题，发现商机的有利工具。本文也将基于Jess[11]对数据质量进行评价研究。

第2章 本体及相关技术

## 2.1 本体简介

本体英文对应的是ontology，该词最早是出现在哲学领域的文献中的，最初是对事物本身的探究。目前，该词以北延伸，成为很多领域中的术语。在人工智能界，Neches等人给出了本体最早的概念。他们认为：本体是对某一领域广泛认同术语的集合，这个定义显然使得本体看起来并没有什么太大的实际作用，缺少联系和逻辑。随着越来越多得人对本体进行研究，其他人也根据自己的理解给出了他们自己的理解，其中得到普遍认同的是Gruber提出的。他的意思就是，本体是各个概念在逻辑理论上的关系形成的一种分类方法。在具体的一个领域中，本体就是领域术语和它们之间的关系和层次。本体也可以成为软件开发的对象模式和类的定义。本文采用本体为数据质量进行建模，以此为其他探究提供基础的支持。

## 2.2 OWL语言

OWL：Web Ontology Language，网络本体语言，用于对本体语义进行描述。OWL是以DAML+0IL为基础开发的，因此也保留了DAML+0IL的部分特性。W3C定义了三种不同的OWL语言，包括OWL Lite、OWL DL和OWL Full，用以应对不同的需求来选择不同的语言类型。

OWL Lite具有最弱的表达能力。它牺牲了部分公理约束，从而提高了推理的效率。OWL Lite是仅需要一个分类层次和简单约束的用户的最佳选择。

OWL DL：Description Logic，描述逻辑，它注重于可判定的推理能力，具有较强的表达能力。而不关注对于RDF的相性。如果需要逻辑的推理能力，那么就可以选择OWL DL。

OWL Full没有任何限制，持有OWL语言的全部成分，但不能保证是可进行判定推理的。对于不要求计算，需要强大的表达和自由的需求，可以考虑OWL Full。

## 2.3 Protege本体建模工具

Protege是一个开源的，并被广泛应用的本体创建和编辑的软件。它也能够实现一些其他的功能比如推理，排列等。它具有以下特征[13]：

（1）允许用户自定义的知识集合；

（2）强大的本体导入导出功能；

（3）具有友好的开发界面；

（4）优异的可扩展功能，庞大的插件库使得非常方便的增加所需的功能；

（5）实现了本体的半自动合并和排列；

（6）Protege软件支持基于框架和应用于语义网两种类型的本体建模方式。

由于该软件易于上手、性能优越、开源免费、可扩展、持续更新等诸多优点而受到广泛的使用。

## 2.4 Apache Jena技术

Jena是Java应用程序对RDFS类文件进行解析的一个优秀的Java框架类库。最开始是由HP公司开发维护，后来转由Apache维护。其实，Protege也是在Jena基础上开发的。本文Jena除了处理owl文件以外，还可以处理RDF，这是Jena最基本的功能，除此之外，就是实现了基于规则的推理。有了Jena，我们就可以实现应用程序对本体文件的解析，而不用我们每个人都去写一套类库或者方法库去解析本体文件。就好比我们需要用电脑去完成我们的各种工作，但我们不必每个人都去造自己的电脑，甚至我们也不必知道如何去造一台电脑，只需别人造好，我们拿来用就可以了。在Jena中，处理owl文件是与处理一般RDFS文件有着很大的区别的。OntModel是专门处理本体的，它具有父接口Model许多没有的方法，比如listClasses()方法是列出所有的类，listObjectProperties()方法是列出所有的Object属性，这两个概念是本体中才有的。另外，Jena还提供了OntDocumentManager（本体文档管理器）用来协助管理Ontology中的class。

## 2.5 Jess介绍

Jess是[Java](http://baike.baidu.com/item/Java/85979" \t "http://baike.baidu.com/_blank)平台上的[规则引擎](http://baike.baidu.com/item/%E8%A7%84%E5%88%99%E5%BC%95%E6%93%8E" \t "http://baike.baidu.com/_blank)，它是CLIPS[程序设计语言](http://baike.baidu.com/item/%E7%A8%8B%E5%BA%8F%E8%AE%BE%E8%AE%A1%E8%AF%AD%E8%A8%80" \t "http://baike.baidu.com/_blank)的超集，由Sandia国家实验室的Ernest Friedman-Hill开发。

Jess是一个体积最小、运行速度最快的推理引擎。它通过模式匹配，读取一条或几条规则，通过规则中的事实和关系，推理出一个新的事实。它是使用一种解决复杂的多对多模式匹配很高效的Rete算法来处理规则，这是它快的重要原因。

Jess有许多特性：后向[链表](http://baike.baidu.com/item/%E9%93%BE%E8%A1%A8" \t "http://baike.baidu.com/_blank)、[运行内存](http://baike.baidu.com/item/%E8%BF%90%E8%A1%8C%E5%86%85%E5%AD%98" \t "http://baike.baidu.com/_blank)查询以及操作和直接推理Java对象的能力。Jess也可以在不编译Java代码的情况下，直接创建对象并调用Java中的方法。

## 2.6 数据质量

在大数据时代，经常会听到人们谈论关于拥有大量数据，却缺少有用信息的问题。其中，数据分析技术不完善是最重要的问题。除此，保有数据质量较低也是重要因素之一，这就致使空有大量数据，而不能产出应有的数据效益。

在一个企业中，用户数据是最具价值的无形资产。高质量的数据通过分析就能够让公司及时做出决策，把即将发生的危机扼杀在摇篮之中，即将呈现的商机提前研发布局。

具有较高一致性、完整性、准确性的数据，才能称得上是高质量的数据。一个企业中数据质量的高低是根据数据分析后结果作出正确决策的决定因素。如何提高企业数据的数据质量，就是本文所探究的重中之重。本文将结合本体知识、熵权算法、Jess推理机等就数据质量评价的方案作出探究，对数据质量评估方法进行切实的介绍、实现与分析。

## 2.6熵权法

熵是一个系统无序程度的度量[13]。熵值过大，呈现出的，便是这个体系足够混乱。在信息系统中，熵值越小，则说明该组数据指标所提供的有效信息量越大，也因此该指标应该占有更大的权值，这就是熵权法的基本思路。层次分析法受到人为因素影响太大，结果太过于主观，结合熵权法，可以在一定程度上降低主观因素造成大影响，是数据质量评价的结果更加贴近于事实。在我们的信息系统中，自然是希望我们的数据越精确越好，所以通过熵权法来计算个指标的综合评分，用以去除信息系统中的垃圾数据。

## 2.7 Java

Java是一种拥有跨平台、面向对象、泛型编程的特性，被广泛应用的一种编程语言。Java是以C++为基础开发的一种开源语言，既吸收了C++面向对象的优秀思想，摒弃了C++中难以理解的多重继承和地址操作，又实现了扩平台，泛型约束，分布式，嵌入式和对web的支持等优秀的功能，使得Java从众多优秀的编程语言中异军突起，如今已成为全球最火、使用量最多的计算机编程语言。本文选择Java的一个重要原因是因为Java语言有很多的第三方类库，免去了在前人已解决问题上浪费时间，提高了开发的效率。

## 2.8 IntelliJ IDEA介绍

IDEA全称IntelliJ IDEA，是Java语言开发的IDE（集成开发工具）。在Eclipse一家独大，傲视群雄的时候，这个新的IDE悄然诞生。由于其强大的功能完爆Eclipse，使得IDEA迅速在编程社区刷屏。但目前其取代Eclipse的最大障碍就是完整版是收费的，且比较贵。不过对于一般情况而言，社区免费版已经可以满足用户使用了。

IDEA也是一款可扩展的软件。其在智能代码提示，重构，调试，主流框架的支持等方面远超Eclipse等其他IDE，对于版本控制工具也有很好的支持。

IDEA所提倡的是智能编码，是减少程序员的工作，IDEA有如下特色功能：

（1）完美的自动代码完成

（2）丰富的导航模式

（3）JUnit的完美支持

（4）对重构的优越支持

（5）智能提示

（6）XML的完美支持

（7）代码检查

（8）历史记录功能

（9）对JSP的完全支持

（10）可自定义的排版功能

（11）智能编辑

（12）EJB支持

（13）多点编辑模式

（14）预置模板

（15）JavaDoc预览支持

（16）动态语法检测

（17）版本控制完美支持

（18）未使用代码的检查

（19）智能选取

（20）智能生成

（21）正则表达式的查找和替换功能

（22）程序员意图支持

第3章 数据质量本体构建

## 3.1 本体的概念、关系以及约束公理

3.1.1本体的概念

本体在经过如此长久地发展后，有了很多的定义。被普遍接受的定义是本体具有以下四层含义[14]：

① 概念模型，从具体情况事物抽离出独立于该事物的概念得到的模型；

② 明确性，对于概念的类型、关系、约束等进行明确的定义；

③ 形式化，能够被计算机理解；

④ 共享，在领域内，概念应该被广泛认同。

本体相当于一个领域术语收录集，记录了在领域中被广泛认同的概念，以及他们之间的关系、层次，使之清晰地呈现在使用本体的人们眼前。

3.1.2本体之间的关系

在前辈的研究积累下，已经确定本体中存在四种最基本的关系。它们分别是：部分与整体的包含关系，父类与子类的继承关系，对象与实例的映射关系，类与类之间的属性关系。详细如下：

1. 部分与整体的包含关系：part-of，表示一个类包含另一个类，却又独立于这个类。比如：门是一个房间的一部分，但门又是一个独立的类；

（2）父类与子类的继承关系：kind-of，表示一个类是另一个类的子类，子类会继承父类的全部属性、关系和约束。比如：草是植物的子类，草就具有生命周期的约束，被动物吃的关系，和光合作用的属性；

（3）对象与实例的映射关系：instance-of，表示类与类的实例之间的关系。例如人类有一个实例是张晓，则张晓与人类就是类的实例与类的关系；

（4）类与类之间的属性关系：attribute-of，表示的是一个类与另一个类的某种关系，对应于OWL就是Properties，包括Object Properties和Data type Properties两种。比如：Object Properties相当于动物类吃植物类，而Data type Properties就相当于张晓喜欢吃茄子。

3.1.3本体的约束公理

本体中最重要的部分，就是约束公理。一个约束公理只在一个具体的情况下有效。在没有特别声明的情况下，一个约束公理对应于其中一个概念的一个属性，在其他概念的该属性上也是不存在该约束的。当然了，对于其他情况，我们这里也不做过多的介绍，有兴趣的朋友可以到论坛或者社区去寻找自己的答案。被约束的类是一个实体类，而约束是一个匿名类，被约束的类通过继承获得该约束。约束公理的这种特性符合最小本体承诺规则，这就表示只定义最通用的公理，就可以在大多数情况下适用。约束公理分为两种，分别为值域约束和基数约束[15]。

1. 值域约束公理

值域约束公理是本体中对于属性值域range的约束，包含三种，分别为：全值实例约束、存在实例约束、含有约束。

（1）全值实例约束：allValuesFrom，就如字面意思，该属性的全部取值必须为全值实例约束指定的类的实例，可为空；

（2）存在实例约束：someValuesFrom，该属性的全部取值中至少有一个是存在实例约束指定的类的实例，不可为空；

（3）含有约束：hasValue，该属性的全部取值中至少有一个是含有约束指定的值，不可为空。

2.基数约束公理

基数约束公理是对属性取值的个数的约束，它有三种，分别为最多取值个数约束、最少取值约束、范围约束。

（1）最多取值个数约束：maxCardinality，该属性的取值个数不能超过最多取值个数约束约束的个数；

（2）最少取值个数约束：minCardinality，该属性的取值个数不能低于最少取值个数约束约束的个数；当该约束值为1时，就相当于非空约束。

（3）范围约束：cardinality，范围约束顾名思义时对属性取值个数的所在范围的约束。它相当于是由一个最多取值个数约束和一个最少取值个数约束组合起来。

OWL 约束公理的应用主要在以下3个方面：

（1） 定义类的必要约束。就是必然会满足的约束，该类的全部实例必然满足这个约束，但满足这个约束的不一定就是该类的实例。

（2）定义类的充要约束。就是一种可逆的必然关系的约束，该类的全部实例必然满足这个约束，反过来，满足这个约束的也一定就是该类的实例。

（3）定义类的轮廓，是一种封锁性的约束。通过约束组合对某个属性在值域上的范围进行约束，有两种形式。第一种：用全值实例约束封锁存在实例约束；第二种：用范围约束封锁全值实例约束。

## 3.2 本体的构建准则

对于不同的领域和环境而言，本体的构建准则也会有一定的差异，目前较为得到较为普遍接受的是Gruber创造的五条准则[16]。

（1）清晰性、客观性与明确性：即要求一目了然，能够明确、清晰地展现其中的内涵；

（2）完全性：即要求本体中的术语的定义要尽可能的完整，不能莫能两可。

（3）一致性：即要求通过已定义的概念进行推理得到的结论应该在转化为自然语言时与已用定义、公理保持大体的一致，不产生冲突；

（4）最大单调可扩展性：没有扩展性就很难复用、升级，本体要能够在不用修改现有基础上添加要扩展的项；

（5）最小承诺和最小编码：原创者对本体定义的约束要尽可能的小，这样才能使本体能够很方便地复用与共享，复用者只需根据自己的需求添加自己的约束规则即可。

后人又在研究之后给出了另外几条现在相对仍表现出色的几条规则：

（1）概念命名标准化

（2）概念层次多样化

（3）语义距离最小化

概念命名标准化使得使用者能够见名知意，进一步的提高本体的共享性；概念层次多样化使得本体的集成能力变得高效；语义距离最小化指将近义词进行合并，降低本体的冗余度，使本体能够更加简洁明了。

这些创建规则都是由前人在长期的工作中总结的，对于我们的指导作用自然是不言而喻的。虽然现在并没有足够成熟的本体构建技术，但我们在创建本体的过程中，可以以此为依据，加上自己的理解，所构建的本体也足以满足我们当前的需求。

## 3.3 数据质量本体构建流程

在前文中我们也提到了，仍然没有足够成熟的本体构建技术。目前的各种构建机制各异，方法也可谓百家齐放，但这些方法中的自动化、半自动化都不够健全。即便如此，瘸子里面也要挑个将军，下面本文介绍几个在学术界中比较被普遍接受的本体构建方法[17]。

（1）IDEF-5方法

本文中采用的就是IDEF-5方法，它是在图表语言和细化说明语言的基础上，经过结构分析和术语提取等，获得构建本体所需要的类、属性、联系、约束等，再创建本体的方法。它的创建步骤如下图3-1所示：

图3-1 IDEF-5方法

定义课题、组织队伍

收集数据

本体优化与验证

本体初步开发

分析数据

（2） Skeletal Methodolody骨架法

骨架法是专门用来创建企业本体的方法。它常用来充当商业企业间术语和定义的集合。它的创建步骤如下图所示：

图3-2 Skeletal Methodolody骨架法

不符合

确定本体的应用目的和范围

本体分析

本体表示

评价

本体评价

建立本体

符合

（3）TOVE企业建模法

TOVE法是加拿大Toronto大学企业集成实验室开发的应用于商业过程和活动建模领域的本体开发模式。

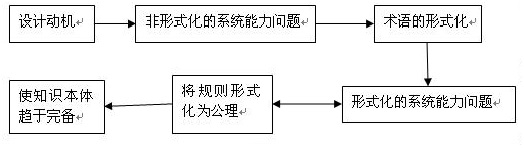


图3-3 TOVE企业建模法

(4) Methontology方法

这是一种组合的方法，具有更加普遍的使用环境。他结合GOMEZ-PEREZ法和传统骨架法，抽取他们的优点，用两种不同的技术，支持了两各方面——开发进程和生命周期的本体构建方式。它的创建步骤如下：

管理阶段：用于规划任务进度，确立所需资源，以及如何保证质量等问题

开发阶段：在此阶段将着手进行本体的开发，分为五个步骤，分别为规范说明、概念化、形式化、执行以及维护

维护阶段：本体开发结束，进入本体的维护阶段，也分为五个步骤，包括知识获取、系统集成、评价、文档说明和配置管理

(5) 七步法

七步法的创建中的是否可以复用现有本体可以再很大程度上减少重复工作的投入，一般可以降低开发成本和时间。该方法由七个步骤组成：确定本体的专业领域和范畴；是否可以复用现有本体；列出本体中的重要术语；定义类和类的等级关系；定义类的属性；定义属性的分面；创建实例[18]。

当然了，数据质量本体的构建方法并不局限于此。不管选用哪种本体构建方法，可扩展性、术语清晰、推理一致、最小编码、本体约定最小都是领域专家在构建本体时应遵循的基本规则。具体情况下，也要根据自己所在的领域，追求的目的等各方面因素，选择或者开发最有效的本体构建方法。不过，考虑到成本的问题，在有时虽不是最有效但不存在较大问题的情况下，可以复用现有本体。只是一种极大节约成本且行之有效的方法。

## 3.4 数据质量本体类以及属性

通过我们对所拥有数据的分析，已经能够大致呈现出数据质量本体的类、属性、约束、联系等基本项。我们定义的初步本体属性包含项如表3-1所示：

表3-1 初步本体属性表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名称 | 对象属性 | 数据类型 | 定义域 | 值域 |
| HasMorethan | 是 | String | DATACOLUMN | DATACOLUMN |
| HasLessthan | 是 | String | DATACOLUMN | DATACOLUMN |
| HasExist | 是 | String | DATACOLUMN | DATACOLUMN |
| Equals | 是 | String | AssessedGoal | AssessedGoal |
| GreaterThan | 是 | String | AssessedGoal | AssessedGoal |
| NotEquals | 是 | String | AssessedGoal | AssessedGoal |
| LessThan | 是 | String | AssessedGoal | AssessedGoal |
| NOTNULL | 否 | Boolean | DATACOLUMN | Boolean |
| TABLE\_ID | 否 | String | DATATABLE | String |
| COLUMN\_ID | 否 | String | DATACOLUMN | String |
| COLUMN\_NAME | 否 | String | DATACOLUMN | String |
| DATA\_TYPE | 否 | String | DATACOLUMN | String |

## 3.5 数据质量规则对在本体中的描述

数据质量本体是对数据质量的评估规则构建一个本体，数据质量的具体评估方法还没有系统的评估框架。所以本文在进行数据质量本体构建时，将抽取数据质量评估规则中具体代表性的数据质量评估规则，具体的数据质量评估规则如下所示：

1. 非空约束：非空约束是完整性约束的一个部分，非空约束要求数据表中某个字段的取值不能为空。该约束如表3-2所示：

表3-2 非空约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 数据属性 | 定义域 | 值域 |
| 1 | NOTNULL | DataColumn | Boolean |

1. 值域约束：值域约束要求数据表中某个字段的取值必须在一定的范围内，如最大值，最小值约束。该约束如表3-3所示：

表3-3 值域约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 数据属性 | 定义域 | 值域 |
| 1 | MIX\_VALUE | DataColumn | Float |
| 2 | MAX\_VALUE | DataColumn | Float |

1. 等值函数依赖约束：指统一记录上，该数据项的数据和其他某些数据项的数据具有函数式上的等值关系。该规则与等值一致性依赖约束的不同体现在数据的范围上。该约束如表3-4所示：

表3-4 等值函数依赖约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象属性 | 定义域 | 值域 |
| 1 | Equals | AssessedGoal | AssessedGoal |

1. 逻辑依赖约束：逻辑依赖约束指在统一记录上，该数据项与其他某些数据项存在逻辑上的关系。如大于、小于、不等等。该约束如表3-5所示：

表3-5 逻辑依赖约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象属性 | 定义域 | 值域 |
| 1 | LessThan | AssessedGoal | AssessedGoal |
| 2 | GreaterThan | AssessedGoal | AssessedGoal |
| 3 | NotEquals | AssessedGoal | AssessedGoal |

1. 等值一致性依赖约束：等值一致性依赖约束指在数据表上，该数据项的数据和其他某些数据表的某些数据项的数据具有函数式上的等值关系。该约束如表3-6所示：

表3-6 等值一致性依赖约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象属性 | 定义域 | 值域 |
| 1 | ReferenceEquals | AssessedGoal | ReferenceColumn |

1. 逻辑一致性依赖约束：逻辑一致性依赖约束表示在数据表中，该数据项与其他某些数据表的某些数据项存在逻辑上的关系。如大于、小于、不等等。该约束如表3-7所示：

表3-7 逻辑一致性依赖约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象属性 | 定义域 | 值域 |
| 1 | ReferenceGreaterThan | AssessedGoal | ReferenceColumn |
| 2 | ReferenceLessThan | AssessedGoal | ReferenceColumn |

1. 存在一致性依赖约束：存在一致性依赖约束表示在数据表中，数据项的数值与其他某些数据表的某些数据项上的数据是一致的，及类似与外键约束。该约束如表3-8所示：

表3-8 存在一致性依赖约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象属性 | 定义域 | 值域 |
| 1 | HasExist | DataColumn | DataColumn |

1. 连续性约束:表示在给数据项上，数据必须是连续取值的，该约束如表3-9所示：

表3-9 连续性约束

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象属性 | 定义域 | 值域 |
| 1 | HasExist | DataColumn | DataColumn |

第4章 基于Jess的数据质量评价方法

## 4.1 基于SWRL的Jess推理

4.1.1 SWRL与Jess

在我们对数据质量评价的过程中，我们要使用很广泛的规则推理。由于本体本身并不支持用于逻辑推理的规则，所以我们引入SWRL规则。它是类似于编程语句的规则书写，由推理机通过模式匹配获取逻辑，推理出相应的结果的规则[19]。

Jess是一个优秀的推理机，它可以完成下面四个任务:①将SWRL的规则转换为Jess的规则或直接在Jess中基于本体概念创建规则；②将有关OWL个体的知识表示为Jess的事实；③基于规则和事实运行Jess推理引擎进行推理；④用推理后得到的新事实更新OWL知识库。工作过程如下图所示

图4-1 Jess工作原理

类、属性

属

性

值

SWRL规则

个体

(实例)

本

体

知识库

规则

事实

Jess

推

理

引

擎

新事实

(1)

(2)

(3)

(4)

4.2.2 Jess与Protege的集成

在Protege上安装JessTab插件后就完成了Jess与Protege的集成。具体步骤如下：

（1）下载Jess的jar包，解压后将jess.jar文件放在Protege软件的plugins文件夹下的edu.stanford.smi.protegex.owl下；

（2）打开Protege，在界面上点击菜单上的project，选择configure…，在弹出的界面上找到SWRLTab勾选，点ok即可在主界面看到SWRLTab标签；

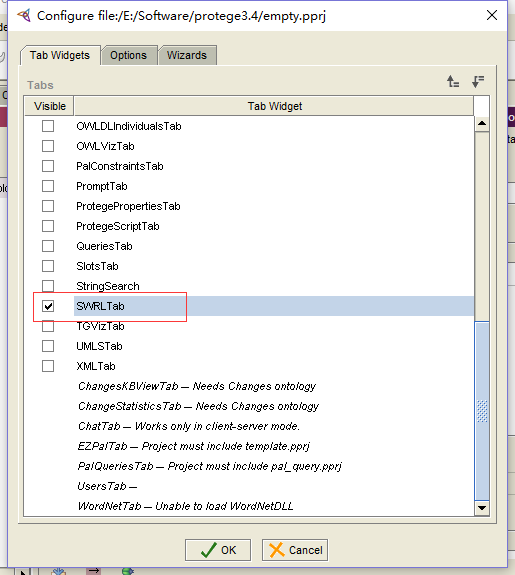


图4-2 显示SWRLTab标签

（3）切换到SWRLTab标签下点击如图4-3的按钮，即可弹出Jess推理工具。

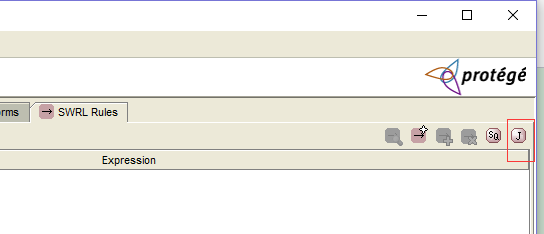


图4-3 启用Jess插件

## 4.2 熵权法在本体中的设计实现

4.2.1熵权法用于数据质量评价的基本原理

熵权法是在已知对象集合各评定指标值的基础上，计算出在竞争意义上的激烈程度[20]。熵是对混乱程度的度量。在数据质量评价中，随着信息量的增加，不确定的因素就会越少，即熵就会越小；反之熵就会越大。 熵值为1即不存在有价值的信息，其熵权则对应为0，这就表示取消该指标。熵权的取值范围是[0,1]，熵权与指标重要性为正关系，表现在结果中即为个指标的加权平均数。我们用表示第个指标的熵权，表示指标总数，则。

4.2.2熵权法的具体设计步骤

1. 首先要确立的，就是熵权法的评价对象，即为指标水平矩阵。矩阵

 （4-1）

其中表示第个评定对象的第个指标的数值即对应位置的重要性，表示横坐标的长度即评价规则的个数，表示纵坐标的长度即指标字段（本体待评价实例）的个数。

(2) 数据的标准化处理。进行同质化是因为存在不同指标之间计量单位不同的情况。在我们进行综合指标计算之前，要先进行解决标准化处理的问题。由于正向指标与负向指标代表的含义不同，所以将正负指标分开处理。公式如下：

正向指标：

 （4-2）

负向指标：

 （4-3）

(3) 计算第项指标的熵值。

处理之后的公式如下：

 （4-4）

式中：，，

(4) 计算评价指标的熵权。公式如下：

 （4-5）

式中：

4.2.3计算综合指标权重

熵权法极大地利用了原始数据中所存在的信息，客观的反映了结果。但是它的缺陷别是不能结合专家的经验与知识，以及决策者的侧重点，造成结构不够切合实际需求。而层次分析法恰好与之相反，是以专家的知识和经验、决策者的侧重点为核心，导致主观随意性过高的问题。综合来看，将两者结合起来使用就能达到理想的效果。公式4-5的结果是通过原始数据计算而得，是比较客观的权重结果；而公式4-7的计算结果是由主观给出后转化而来的。通过公式4-6，将两者融合，使得权重不过于主观，又不会完全不受环境和不同追求对于某指标的偏重。

 （4-6）

 （4-7）

式中：，为层次分析法中主观赋权的权重。

4.2.4确定综合评分结果

通过加权由综合权重计算出综合评分结果

 （4-8）

根据上述熵权法的设计，根据熵权法得出的结果，设计熵权法有关本体的数据属性和推理结果的数据属性如表4-1和4-2所示，该表包含了对所有8种约束的得分统计。

表4-1熵权法本体设计数据属性

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 数据属性 | 定义域 | 值域 |
| 1 | HasNotNullScore | AssessGoal | float |
| 2 | HasRangeScore | AssessGoal | float |
| 3 | HasEqualFunScore | AssessGoal | float |
| 4 | HasLogicScore | AssessGoal | float |
| 5 | HasEqualCScore | AssessGoal | float |
| 6 | HasLogicCScore | AssessGoal | float |
| 7 | HasExistCScore | AssessGoal | float |
| 8 | HasConScore | AssessGoal | float |

表4-2熵权法本体设计对象属性

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 对象属性 | 定义域 | 值域 |
| 1 | HasCompeletenceScore | AssessGoal | HasGrade |
| 2 | HasConsistency | AssessGoal | HasGrade |
| 3 | HasAccuracy | AssessGoal | HasGrade |

## 4.3 数据质量的Jess推理实现

SWRL规则的编写要分别从完整性、一致性、准确性三个方面出发，来判定一个数据是否符合对应的要求。给最终的推理结果一个指定的等级填入本体中。

（1）完整性

完整性包括完整性约束规则、非空约束规则、连续性约束规则三个维度的规则，用来展现数据的完整程度。完整性的推理规则如下：

AssessGoal(?x) ∧HasNotNullScore(?x, ?y) ∧ HasContinentScore(?x,?z) ∧swrlb:greaterThan(?y, 0.098) ∧ swrlb:greaterThan(?z, 0.045) → HasCompeletence(?x, 重要)

（2）一致性

在前文中也提到了有关一致性的两种情况，在这里就不在阐述。在纬度上，一致性对应等值一致性依赖约束规则、存在一致性依赖约束规则、逻辑一致性依赖约束规则三个维度的规则。一致性的推理规则如下：

AssessGoal(?x) ∧HasExistScore(?x, ?y) ∧ HasLogicdependenceScore(?x,?z)∧HasEquiconsistency dependencyScore(x,a) ∧swrlb:greaterThan(?y, 0.01) ∧ swrlb:greaterThan(?z, 0.01)∧swrlb:greaterThan(?a, 0.01)→ HasConsistency(?x, 不重要)  
 （3）准确性

准确性包含值域约束规则、代码约束规则、逻辑依赖约束规则、词法约束规则、非空约束规则、等值函数依赖约束规则六个维度。数据有越高的准确性，推理结果的误差就会越小。影响数据准确性的因素主要就是数据的错误录入、错误拼写、数据空缺、不同来源数据冲突等一直无法有效解决问题的存在导致的。准确性的推理规则如下：

AssessGoal(?x) ∧HasNotNullScore(?x, ?y) ∧ HasLogicScore(?x,?z)∧ HasCodeConstraintScore(?x,?a) ∧HasRangeScore(x,b) ∧HasLexicalScore(x,c)∧HasEquivalentScore(x,d)∧swrlb:greaterThan(?y, 0.01) ∧ swrlb:greaterThan(?z, 0.01)∧swrlb:greaterThan(?a, 0.01) swrlb:greaterThan(?b, 0.01) ∧ swrlb:greaterThan(?c, 0.01)∧swrlb:greaterThan(?d, 0.01)→ HasAccuracy(?x, 不重要)

第5章 系统实现

## 5.1 数据质量本体

在做数据质量评价之前，先要创建基本的数据质量本体。前文中我们已经就构建数据质量本体的方法进行了介绍，并在大体上说明了该数据质量本体应该有的类、属性等项。这里将依此开始数据质量本体的构建，过程如下：

（1）打开protege，这里可以看到如图5-1所示的小窗口，我们这里选择新工程；

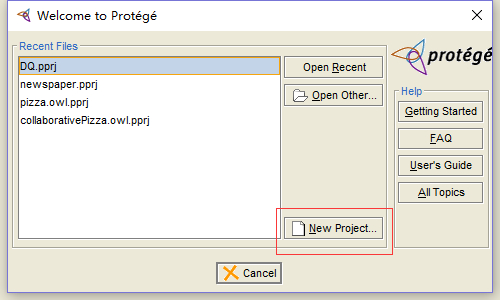


图5-1 创建新工程

（2）选择工程类型，我们选择OWL/RDF Files，本体工程；

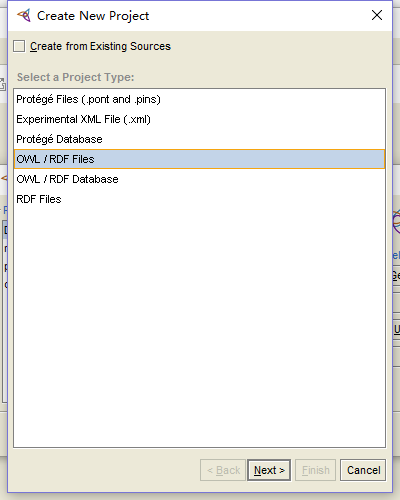


图5-2 选择工程类型

（3）填写本体URI，默认即可；

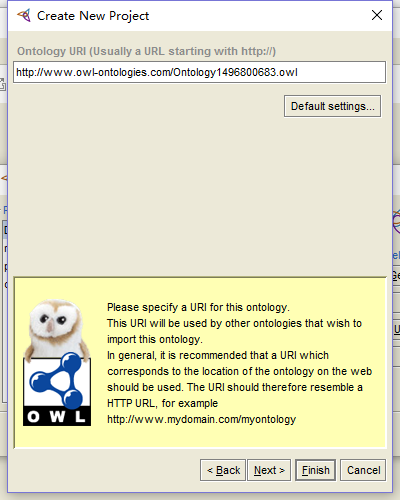


图5-3 填写本体URI

（4）选择文件类型，此处我们选择OWL Full类型或者OWL DL类型，关于OWL Full类型、OWL DL类型和OWL Lite三者的区别已在前文中详细的进行了说明；

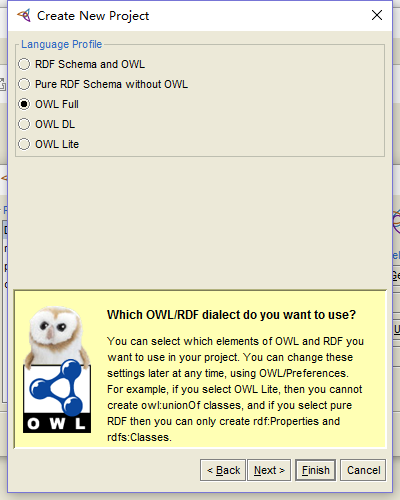


图5-4 选择文件类型

（5）选择视图。视图有Logic View和Properties View两种，选择哪种可不必在意，创建后可在主界面右下角随意切换；

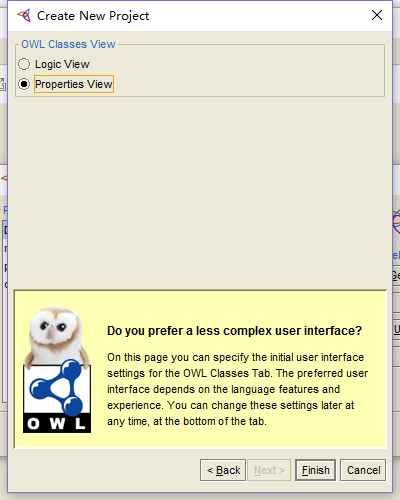


图5-5 选择视图

（6）通过Protege创建class。新建本体后在上方切换到OWL Classes标签下，可以看到如图5-6界面。右键owl:Thing弹出右键菜单，选择Create subclass标签即可创建一个类；

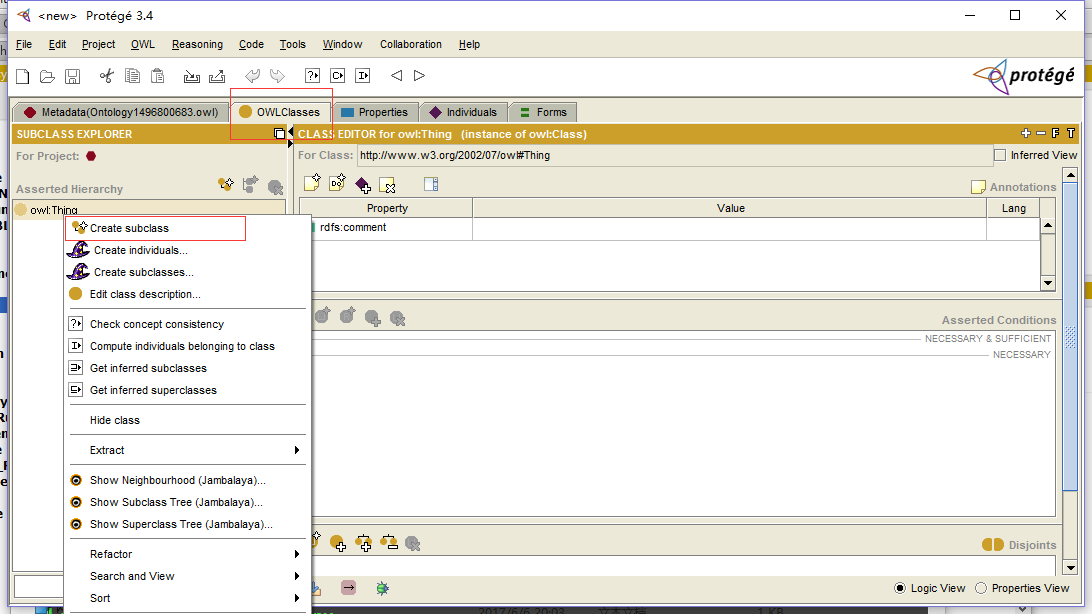


图5-6 通过Protege创建class

（7）通过Protege编辑class。创建class后双击创建的class即可进入编辑界面，以及数据质量本体的部分类如图5-7所示。

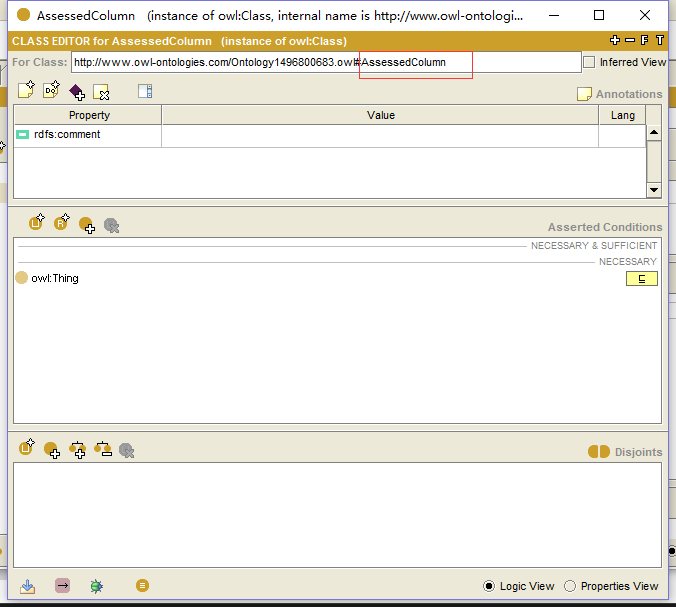


图5-7 通过Protege编辑class

图5-8展示了创建完成后的全部数据质量本体类。

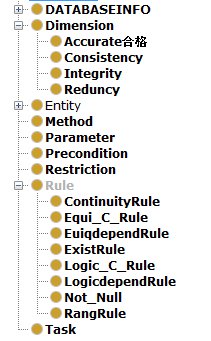


图5-8 数据质量本体类

（8）通过Protege够建Object Properties；

选择主界面的Properties标签页，选择Object，点击添加按钮，如图5-9所示；

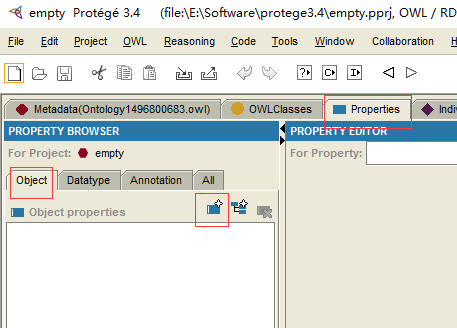


图5-9 通过Protege创建Object Properties

创建之后双击创建的对象属性，即可进如编辑页，可以进行重命名，设置定义域（Domain）和值域（Range），以及数据质量部分对象属性如图5-10和所示。

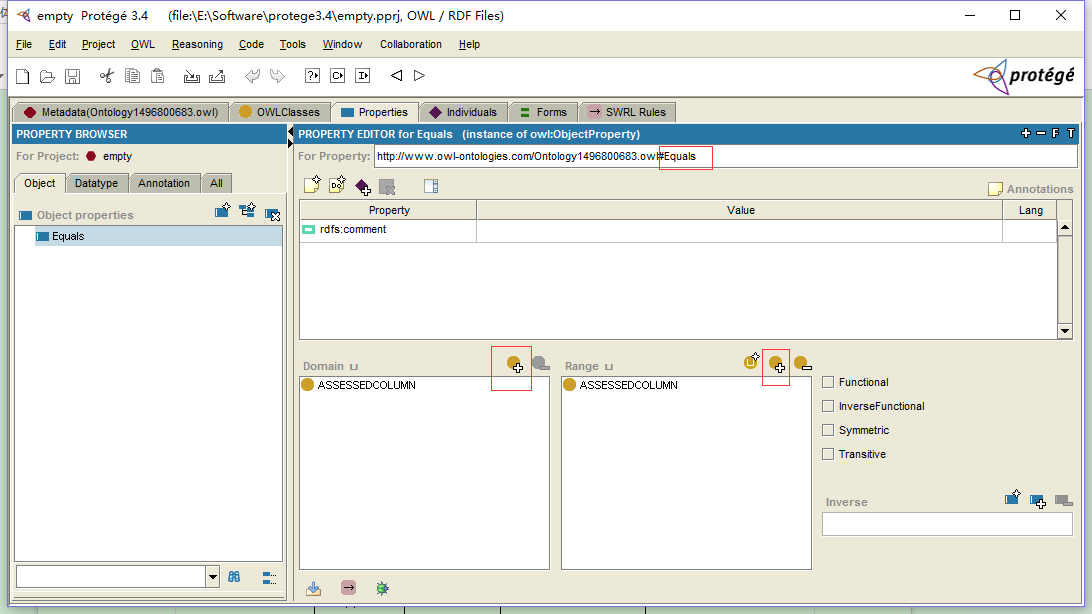


图5-10 通过Protege编辑Object Properties

创建完成后的对象属性如图5-11所示：

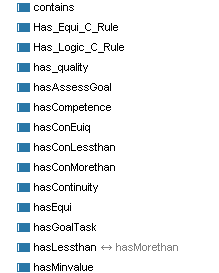


图5-11 数据质量部分对象属性

（9）通过Protege创建Data Properties；

依然是这个标签页，点击Datatype，点击创建按钮创建数据属性，如图5-12所示；

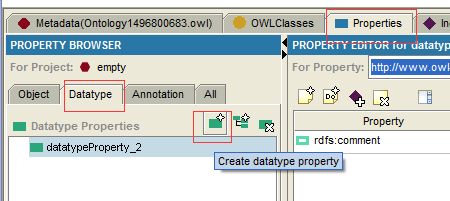


图5-12通过Protege创建Data Properties

双击创建的数据属性，即可进入进如编辑页，可以进行重命名，设置定义域（Domain）和值域（Range），以及数据质量部分数据属性如图5-13所示。

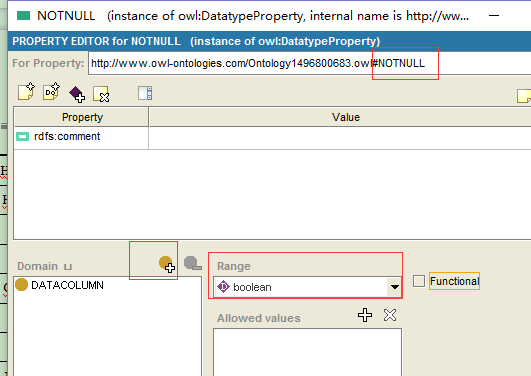


图5-13 通过Protege编辑Data Properties

创建完成后的部分数据属性如下图5-15所示：



图5-14 数据质量部分数据属性

（10）通过Protege创建Individuals

选择主界面的Individuals标签页，选择一个类，点击创建按钮即可创建给累的一个实例，如图5-15所示；

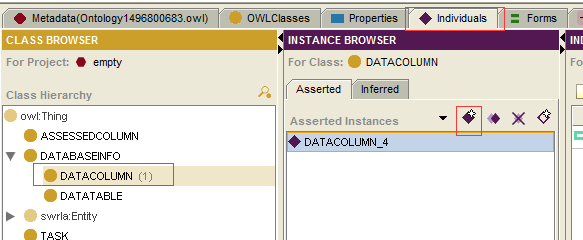


图5-15 通过Protege创建Individuals

双击创建的实例，即可进入编辑页，可以进行重命名，设置值等操作，如图5-16所示；

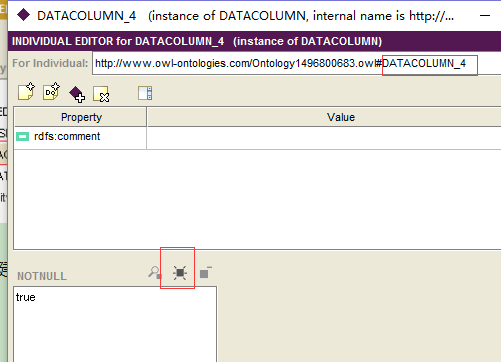


图5-16 通过Protege编辑Individuals

## 5.2 熵权法与数据质量评价的实现

本文是通过Java语言实现来实现熵权法，使用的集成开发工具是IDEA 2017，Java采用的版本为较为新的8.0版本。熵权法实现的最大的难度是熵权法中复杂的数学计算，稍有不慎，就有可能导致错误的计算结果。下面我们以一个实际的例子来说明。

（1）在我们的原始数据中，纵向坐标表示约束规则，共有8种类型，分别为：非空约束、值域约束、等值函数依赖约束、逻辑依赖约束、等值一致性依赖约束、逻辑一致性依赖约束、存在一致性依赖约束和连续性约束；横坐标为钻井地质信息数据项，比如井号（JH）、井型（JX）、井径（JJ）、油补距（YBJ）等多项数据项；数值表示该数据项对该规则的重要性评分，该分数是由领域专家给出的。我们选取其中的部分项来完成这个例子，如表5-1所示；

表5-1 原始数据

|  |  |  |  |
| --- | --- | --- | --- |
|  | JH | JJ | YBJ |
| NOTNULL | 100 | 70 | 30 |
| Range | 20 | 50 | 80 |
| Logic | 80 | 100 | 80 |

（2）对上表数据进行标准化处理；

按照公式4-2和4-3的标准化公式，对表5-1的数据进行标准化处理，得到表5-2的指标权重如下图所示：

表5-2 各指标权重

|  |  |  |  |
| --- | --- | --- | --- |
|  | JH | JJ | YBJ |
| NOTNULL | 1 | 0.4 | 0 |
| Range | 0 | 0 | 1 |
| Logic | 0.75 | 1 | 1 |

（3）计算第项指标的信息熵值。

根据4-4给出的公式进行计算，即可由表5-2计算出信息熵如表5-3所示：

表5-3 信息熵

|  |  |  |  |
| --- | --- | --- | --- |
|  | JH | JJ | YBJ |
| 信息熵 | 0.62 | 0.51 | 0.63 |

（4）计算评价指标的熵权

根据公式4-5，计算出熵权如表5-3所示：

表5-3 熵权

|  |  |  |  |
| --- | --- | --- | --- |
|  | JH | JJ | YBJ |
| 熵权 | 0.31 | 0.40 | 0.29 |

（5）计算综合指标权重

由层次分析法得到的主观权重如表5-4所示：

表5-4 主观权重

|  |  |  |  |
| --- | --- | --- | --- |
|  | JH | JJ | YBJ |
| 权重 | 0.48 | 0.25 | 0.27 |

根据公式4-6、4-7，结合表5-4的数据，即可计算出综合指标权重如表5-5所示：

表5-5 综合权重

|  |  |  |  |
| --- | --- | --- | --- |
|  | JH | JJ | YBJ |
| 权重 | 0.45 | 0.31 | 0.24 |

（6）由综合权重计算出综合评分结果

根据公式4-8，计算出的综合评分结果如表5-6所示：

表5-5 综合评分

|  |  |  |  |
| --- | --- | --- | --- |
|  | NOTNULL | Range | Logic |
| 综合评分 | 74.9 | 43.7 | 86.2 |

通过上一步熵权法的计算，得到了对应项的综合评分。此时，我没就可以通过前面书写的SWRL规则使用Jess插件进行逻辑推理。这里以完整性为例，推理可得到如图5-17所示的结果。

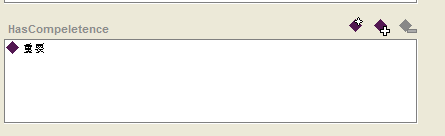


图5-17 完整性重要性

结论

信息时代到来已久，大量数据的产生使得数据筛选、数据分析、数据挖掘等成为信息时代大跨步发展前进的巨大障碍。数据质量本体是对数据评估领域知识的建模，是对数据评估领域知识的规范化模式。主要是抽取了数据质量的评估规则，并且对这些规则进行抽象，提取来构建数据质量本体。通过对数据评估领域知识的规范化，形式数据评估领域的评估标准，使得在数据评估领域有一套可以通用，共享的数据评估标准。利用数据质量本体，对数据进行质量评估，选出其中的不合格数据。所以数据质量的评估对于数据的分析以及研究具有很大的现实意义，能够发现质量不高的数据，将其剔除，提高大数据分析结果的参考价值。

本文结合当前发展实况，身先士卒尝试前辈呕心沥血的研究成果，并经过这几个月的学习与尝试，成功通过自己的双手实现了这个目标。本文通过IDEF-5方法对数据质量本体的建模，成功构建出可用于推理的数据质量本体。之后利用熵权法和层次分析法对于评价数据的计算，得出一个相对客观，又符合专家建议的结果，最后再通过Jess推理机经过逻辑推理获取到数据的质量等级，以此剔除数据系统中会降低我们数据分析结果的脏数据，提高数据分析的可靠性。另外根据本文的探究结果，可见这种思路是切实可行的。希望我的这篇文章能给后来者起到敲门砖的作用。

自然了，本文也有一些探究不足的地方。首先，本文只尝试了一种数据质量评价方法，也没有对其他数据质量评价方案做研究对比，后续还需尝试探究其他的技术方案，与当前方案进行对比，总结出在不同环境下的最佳选择；再次，创建的数据质量本体对于此次的探究到是不存在什么缺陷，但是却缺乏灵活性，扩展性也比较差，因此这也是我值得反思与改进的地方。希望我在数据质量的研究道路上越走越远，越做越好。

参考文献

[1] 官思发,孟玺,李宗洁,刘扬. 大数据分析研究现状、问题与对策[J]. 情报杂志,2015,(05):98-104.

[2] 邓志鸿,唐世渭,张铭,杨冬青,陈捷. Ontology研究综述[J]. 北京大学学报(自然科学版),2002,(05):730-738.

[3] 袁国铭,李洪奇,樊波. 关于知识工程的发展综述[J]. 计算技术与自动化,2011,(01):138-143.

[4] 黄甫山. 大数据时代要警惕“脏数据”[N]. 人民邮电,2013-11-25(008).

[5] 崔金栋. 基于本体的网格信息检索模型研究[D].吉林大学,2011.

[6] 张晓冉,舒昝. 基于关系数据库的油田领域数据质量本体构建[J]. 微型电脑应用,2016,(07):71-73.

[7] Liang Chen,Yang Gong. Knowledge Representation in Patient Safety Reporting: An Ontological Approach[J]. Journal of Data and Information Science,2016,(02):75-91.

[8] 仇宝艳. 面向领域本体的知识建模问题研究[D].山东师范大学,2009.

[9] 汤青. 本体概念及概念间关系抽取方法研究[D].北京信息科技大学,2013.

[10] 李景. 领域本体的构建方法与应用研究[D].中国农业科学院,2009.

[11] Houda Mnasser,Kathia Oliveira,Maha Khemaja,Mourad Abed. Towards an Ontology-based Transportation System for User Travel Planning[J]. IFAC Proceedings Volumes,2010,43(8):.

[12] . On environment-driven software model for Internetware[J]. Science in China(Series F:Information Sciences),2008,(06):683-721.

[13] 章穗,张梅,迟国泰. 基于熵权法的科学技术评价模型及其实证研究[J]. 管理学报,2010,(01):34-42.

[14] B.Chandrasekaran, J.R.Josephson, and V.Richard Benjamins. Ontology of Tasks and Methods[C]. In Workshop on Knowledge Acquisition, Modeling and Management, Canada, 2015.

[15] 林汝坤,刘芳,戴长华,姚莉. OWL本体建模中约束公理的应用[J]. 计算机工程,2006,(16):193-194+223.

[16] 廖莉莉. 本体质量评估方法研究与实现[D].南京航空航天大学,2015.

[17]张文秀,朱庆华. 领域本体的构建方法研究[J]. 图书与情报,2011,(01):16-19+40.

[18] Uschold M,Gruninger M.Ontologies:Principles,Methods and Applications[J]. Knowledge Engineering Review,1996,11(2):93-155.

[19] 纪兆辉,李存华. 基于SWRL和Jess构造语义Web规则及其对策分析[J]. 淮海工学院学报(自然科学版),2009,(04):26-29.

[20] 李汶静. 基于熵权AHP法的企业价值组合评估应用研究[D].西南石油大学,2010.

致谢

时光荏苒，大学四年的生活即将接近尾声，在这大学四年之间，我遇到了很多优秀的老师，结交了很多的朋友，我碰到过很多的难题，我也学会了妥善的解决这些问题。在毕业设计的这几个月中，我学习到了很多的知识，很多解决问题的方法。面对编不出的代码，调试不出来的程序，不合理的本体构建，我都会去耐心的分析问题，从不同的角度思考来解决这些问题。我要衷心的感谢在这个毕业设计中帮助我，关心我的老师同学们。

首先，我要向指导我、帮助我的袁满老师表示衷心的感谢。袁老师是一个热爱科研，钻于科研，治学严谨，工作认真，爱护学生的好老师。本论文从选题到设计到最后论文的撰写，袁老师都一直在热心的指导我，帮助我，给我答疑解惑。不仅让我明白了专业的学科知识，还以自身专业严谨的科研态度深深的感染了我。在此，我衷心的感谢袁老师，感谢袁老师在教会我知识的同时还让我成长了。

其次，我要向帮助过我的师兄师姐们表达我的感谢。谢兰师姐一直在耐心的指导我，我遇到的每一个问题，师姐都会由浅入深、耐心十足的引导我，帮我发现问题、解决问题，所以我要再次感谢谢兰师姐。还有实验室的其他师兄师姐们，他们都给了我很大的帮助，遇到问题时，都会指导我的思路，给我提供学习的资料，帮助我解决问题，让我的动手能力也得到了很大的提升。在此，让我向他们表示我最真诚的感谢。

在这里，感谢与我并肩作战的舍友与同学们，感谢关心我支持我的朋友们，感谢学校领导、老师们给我做出的谆谆教导。感谢你们在我迷茫无助的时候向我伸出上手，感谢你们在我疲惫不堪的时候给予我的鼓励与陪伴，感谢你们在我大学四年里的引导、支持、陪伴，感谢我们在一起的欢声笑语、喜怒哀乐、爱恨情仇、打闹嬉戏。最后，向所有帮助过我、由此受累的老师、同学们表达我诚挚的感谢和祝福，祝愿他们身体健康，工作顺利。