

多知识库整合技术在企业供应链中的应用

彭志平¹, 夏战锋², 周超²

(1. 广东石油化工学院计算机与电子信息学院, 广东 茂名 525000;

2. 江苏科技大学计算机科学与工程学院, 江苏 镇江 212003)

摘要: 企业供应链中数据的独立性较差, 智能化程度较低。为此, 提出一种多知识库整合技术, 并将其应用于企业供应链中。通过寻找 TBox 间的重叠区域, 建立概念关联, 消除数据冗余性和不一致性, 以整合知识库。设计多 ABox 优化技术及其实现算法, 给出服务请求子系统结构、服务接收子系统和知识库整合中心框架。实验结果表明, 该技术能减少系统运行时间。

关键词: 供应链; 知识库; 查询算法; 描述逻辑

Application of Multiple Knowledge Base Integration Technology in Enterprise Supply Chain

PENG Zhi-ping¹, XIA Zhan-feng², ZHOU Chao²

(1. College of Computer and Electronic Information, Guangdong University of Petrochemical Technology, Maoming 525000, China;

2. School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang 212003, China)

【Abstract】 The independence and intelligence of data in enterprise supply chain is low. In order to solve the problem, this paper proposes an application of multiple knowledge base integration technology in enterprise supply chain. Integrating knowledge base methods are divided into finding the overlapping region between TBox, establishing the relevance of conception, and eliminating the redundancy and inconsistency of data. The paper designs multiple ABox optimization technique and its implement algorithm, introduces the service request subsystem, service receiving subsystem and knowledge base integration centre. Experimental results show that the application is effective, and can reduce the running time of system.

【Key words】 supply chain; knowledge base; query algorithm; Description Logic(DC)

DOI: 10.3969/j.issn.1000-3428.2012.02.026

1 概述

企业供应链包括原料、生产、销售等环节, 是一个巨大而繁杂的工程。现代企业供应链管理要求供应链上的信息具备可协调性及相互间有很强的依赖制约关系, 且还具备高度智能化查询效果。以二维表结构为核心的关系数据库技术很难满足上述要求。

基于描述逻辑^[1](Description Logic, DL)的知识库^[2]具备推理自主化和数据语义化 2 个特性, 这恰能解决企业供应链现代化管理的瓶颈, 但是单一、分散的知识库并未解决供应链中的“信息孤岛”问题及各部门数据表示方式和数据结构的不一致性问题。

为解决上述问题, 本文在总结他人研究成果的基础上, 针对企业供应链中存在的问题, 利用知识库及其整合技术研究一种多知识库整合技术在企业供应链中的应用。

2 多知识库的整合技术

2.1 多 TBox 的整合及查询算法

知识库整合的目的是消除同领域中知识库间的数据冗余和不一致性, 以使用户可共享其他用户所提供的数据^[3]。此工作主要通过整合领域内各部门知识库中的 TBox 来完成。

整合工作主要分 3 个过程:

- (1)寻找 TBox 间的重叠区域。
- (2)建立概念关联。
- (3)消除数据冗余性和不一致性。

过程(1)、过程(2)在文献[4-5]中有详细阐述且被绝大部分研究者所认可, 本文不再重复。本文重点讨论过程(3)。

若被整合后的某概念 A 与同一 TBox 中的概念 B 和 C 存在 $A \subseteq B$ 、 $A \subseteq C$ 且 $C \subseteq B$ 的关系, 则需消除概念 A 、 B 与 C 其中一个包含关系。该消除冗余过程如图 1 所示。

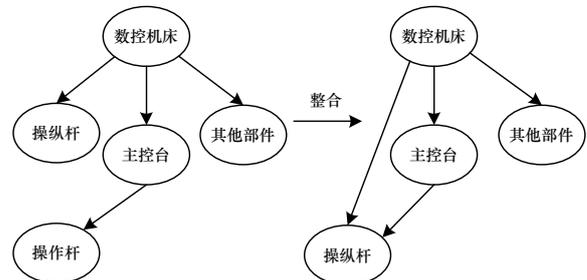


图 1 消除冗余过程

由图 1 可知, 该过程包含数控机床的 TBox 整合, 经过过程(1)和过程(2)后, 得到整合后 TBox 中数控机床的结构如

基金项目: 广东省自然科学基金资助项目(10252500002000001); 广东省教育部产学研结合基金资助项目(2010B090400235); 广东省科技计划基金资助项目(2011B010200011)

作者简介: 彭志平(1969—), 男, 教授、博士, 主研究方向: 语义 Web, 多主体技术; 夏战锋、周超, 硕士研究生

收稿日期: 2011-07-27 **E-mail:** mmxypzhp@yahoo.com.cn

图 1 左半部分所示, 经系统进一步提出建议, 将“操纵杆”和“操作杆”整合, 结果如图 1 右半部分所示, 然而发现, 对于概念“操纵杆”、“数控机床”、“主控台”三者存在相互包含关系, 此时需删除被整合概念“操纵杆”与其中一个概念间的包含关系, 一般而言, 建议删除“数控机床”与“操纵杆”间的包含关系。

TBox 的查询算法描述如下:

输入 该输入 $f(A: a, x)$ 须满足 RDF 要求, 其中, $A: a$ 是主语; x 是宾语; f 是谓语

输出 x , 并且 x 是与 $f(A: a, x)$ 相匹配的 TBox 中的断言

```
While(graph_queue Not NULL){
Vertex=dequeue();/*顶点出队列*/
servicevertexmatch(f(A: a, x),Vertex);/*服务与顶点进行匹配*/
if(servicevertexmatch(f(A: a, x),Vertex)=true)
{x=servicefindinstance();
return x;
break;}
else if(servicevertexcalcsm(f(A: a, x),Vertex)≥m)/*基于 Wordnet
相似度计算, 大于或等于某阈值 m*/
{x=servicefindinstance();
return x;
break;}}
```

对于 $f(A: a, x)$, 其中, $A: a$ 和 x 的位置可互换, 但查询算法同上。

2.2 多 ABox 的优化技术

定义 1 若任意 $A \in \text{ABox}$, 且 H_A 表示 A 中所有断言下的个体集合, 则 A 的值域 Δ_A^I 可定义为: $\Delta_A^I = \{d \in \Delta^I | a \in H_A \wedge a^I = d\}$, 其中, I 表示一种解释。

由定义 1 可把一个 ABox 分割成多个子 ABox。

定义 2 假设 K 是知识库系统中子 ABox 的个数, 并有: $A = A_1 \cup A_2 \cup \dots \cup A_K$, 那么, $A^I = A_1^I \cup A_2^I \cup \dots \cup A_K^I$ 。其中, A_i 表示知识库系统中的子 ABox, $i \in \{1, 2, \dots, K\}$, 则 A 的值域等价于每个 A_i 值域的并集, 即 $\Delta_A^I = \cup \Delta_{A_i}^I$, 且 $j \in \{1, 2, \dots, K\}$, 从而可得 $A_i \subseteq A$ 。

据定义 2 可扩充知识库的形式化定义。

引理 若 $T \in \text{TBox}$, \tilde{A} 是一个集合, 且 $\tilde{A} = A_1 \cup A_2 \cup \dots \cup A_K$, 则知识库 $\Sigma = \langle T, \tilde{A} \rangle$ 。

由引理得到基于查询的知识库体系结构如图 2 所示。

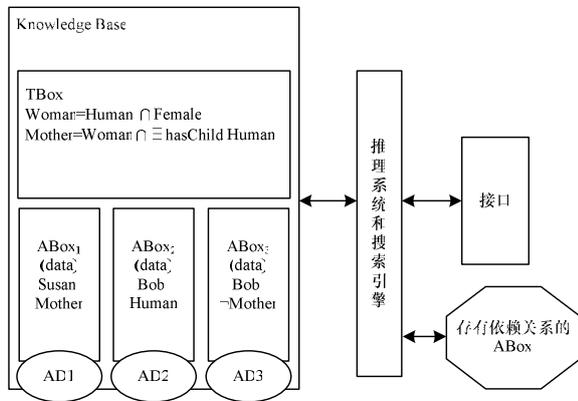


图 2 基于查询的知识库体系结构

由图 2 可知, 一个知识库中的 ABox 被分割成若干个有一定关联规则的子 ABox。本文采用文献[6]提出的“Local as View(LAV)”的分割与关联策略, 其主要思想是按照断言集的相似程度进行分割并通过一定映射机制关联各子 ABox。

定理 若 LAV 中有一映射 M , 则它由断言集构成, 即:

$$M: s \sim \mathcal{O}_S$$

其中, s 是 AD_i 的结构表示; S 是 $\text{ABox}_i(A_i)$ 的结构表示; \mathcal{O}_S 是在 S 上的一个查询; \sim 为描述逻辑中一种运算符。假如 AD_i 与 A_i 满足 M , 则 $s^{AD_i} \subseteq \mathcal{O}_S^{A_i}$, 其中, $i \in \{1, 2, \dots, K\}$ 。由定理知可用 $AD_i^{[7]}$ 指代 ABox_i 。

可把相互间存有依赖关系的 ABox_i 连接起来, 使它们在逻辑上成单个 ABox。每个 ABox_i 按照个体成员间关系来构建相应 ABox 图。

定义 3 任意 $A \in \text{ABox}$ 对应的 ABox 图 $(AG(A))$ 由 (V, E, R, f) 构成, 其中, V 是 ABox 图顶点集合, 每个顶点表示个体名; E 是图中边的集合; f 表示从 E 到 $\{(a, b) | a, b \in V\}$ 的一个函数; R 是图中边上的标识; R_i 表示成员断言关系的关系名即 $R_i(a, b) \in A$, 其中, $a, b \in V, i = \{1, 2, \dots, n\}$ 。

每个 ABox_i 按照定义 3 构建相应的 $AG(A_i)$, 且它是一个无向连通图, 因此, 利用连通图的判定算法判定 2 个 $AG(A_i)$ 间的连通性, 以确定 2 个 ABox_i 是否存有依赖关系。对存在依赖关系的 A_i 和 A_{i+1} , 把它们归为一类, 根据定理知可用 $AD_j = \{A_i, A_{i+1}\}$ 表示, 其中, $i, j = \{1, 2, \dots, n\}$ 。对不存在依赖关系的 ABox_i 需分别考虑。

供应链上的查询在 ABox 上的优化查询算法描述如下:

输入 查询条件 $query$

输出 查询答案 $answer$

```
Partitioned_Q(query, A)
{depset={};/*存放有依赖关系的 A_i*/
Answer=False;
While (all A_i ∈ A be judged)
{AG(A_i)=create_ABox_graph(A_i);/*创建相应的 ABox 图*/
While (all combination(A_i, A_j) be judged)/*每对 A_i 判断完毕*/
{if (find_ABox_dependency(AG(A_i), AG(A_j))=true)/*判断 A_i 和
A_j 存有依赖关系*/
add combination(A_i, A_j) to depset;}
A^c=combine_dependent_ABox(A, depset);/*把多个存在依赖关系的
ABox 归为一类*/
create_abstract_description(A^c);/*创建同一类 ABox 的指代 AD*/
while(all A^h ∈ A^c be judged)/*每个 A^c 是不存在依赖关系的*/
{if(query_relevancy(AD(A^h), query)= true)
answer=answer ∪ content_checking(A^h, query);/*利用查询条件在
相应的 A_i 中寻找答案, 并进行答案合并*/}
return answer;}
```

3 企业供应链模式

本文提出的企业供应链模式包括 3 个部分, 即服务请求子系统、服务接收子系统、知识库整合中心。

3.1 供应链中服务请求子系统

服务请求子系统如图 3 所示。

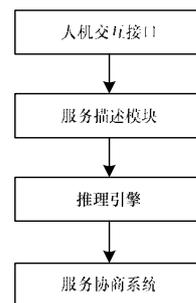


图 3 服务请求子系统结构

由图3可知,服务请求子系统主要实现以下功能:

(1)服务构建,服务请求者通过“人机交互接口”传递服务,由“服务描述模块”来构建系统所能理解并进行相关推理的服务。

(2)服务协商系统的功能为,实现与供应链中服务接收子系统的协调功能。

3.2 供应链中服务接收子系统

供应链中服务接收子系统包括查询预处理、推理引擎和服务管理中心,主要功能有:

(1)服务的预处理,为后续的推理和查询工作提供更加精确的条件。

(2)服务管理中心,包括3个主要功能:

- 1)选择知识库。
- 2)下达整合指令。
- 3)传递功能。

3.3 知识库整合中心

知识库整合中心框架如图4所示。

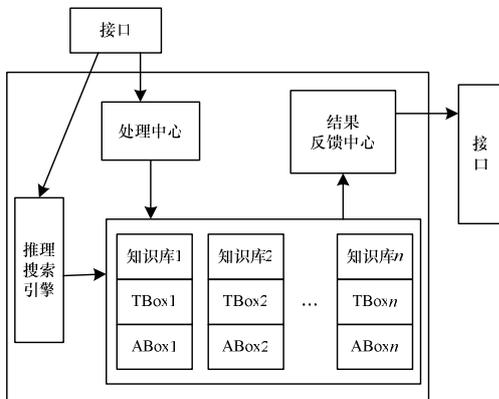


图4 知识库整合中心框架

由图4可知,该中心是整个供应链模式中最核心的一部分,主要功能是:

- (1)整合知识库。
- (2)查询结果。
- (3)优化结果。

3.4 企业供应链中服务查询算法

企业供应链上服务查询的实现算法描述如下:

```
sendservice_SS();/*发送服务请求*/
receiveservice_RS();/*接收服务请求*/
while (R-feedback-center is NULL)/*结果反馈中心值为空*/
    {setintegrate-TBox();/*启动 TBox 整合程序*/
    Setpartitioned_Q-ABox();/*启动 ABox 分割查询程序*/
    sendtoclient();/*结果返回到客户端*/
```

4 实验结果与分析

鉴于目前基于知识库的语义 Web 信息查询没有统一标准的测试平台,但为验证和评估本文提出的企业供应链模式的可行性和有效性,本文采用 LUBM^[8]数据集进行实验。本文利用 LUBM 数据集中部分数据,分成6组。测试环境:软件为 Windows XP SP2+Eclipse 7.0+JDK 1.6.0+PELLET 推理机+Jena 推理机;硬件为 Intel Pentium-Dual 2.0 GHz 的 CPU,以

及 2.0 GB 的 DDR 内存。

本文针对上述6组数据在整合前和整合后进行仿真,查询操作在整合知识库前后的系统运行时间比较如图5所示。

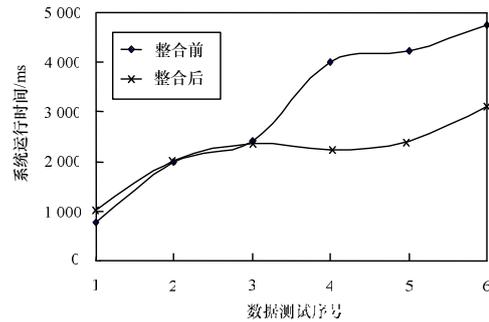


图5 查询操作在整合前后的系统运行时间比较

由图5可知,从测试序号为3及其之后的数据实验中,查询操作在知识库整合后的系统运行时间明显比整合前的运行时间少,减少幅度随 ABox 中事例断言数增加而增大。

5 结束语

本文提出一种多知识库整合技术在企业供应链中的应用。设计多 TBox 的整合技术及其查询算法,给出多 ABox 优化技术及其实现算法。实验结果表明,该应用能优化供应链服务,使查询更佳智能化和提高准确性。今后的研究重点是利用最小代价,将应用系统换成知识库系统。

参考文献

- [1] Baader F, Calvanese D, McGuinness D L, et al. The Description Logic Handbook: Theory, Implementation, and Applications[M]. 2nd ed. [S. l.]: Cambridge University Press, 2007.
- [2] 许文艳,刘三阳.知识库系统的逻辑基础[J].计算机学报,2009,32(11):2123-2129.
- [3] 陈亚兵,孙济庆.基于知识库的专家咨询系统设计与实现[J].计算机工程,2007,33(16):196-198.
- [4] Guo Yuanbo, Pan Zhengxiang, Heflin J. An Evaluation of Knowledge Base Systems for Large OWL Datasets[C]//Proc. of Semantic Web Conference. [S. l.]: Springer-Verlag, 2004.
- [5] Rahaal S, Olivier C. Integrating Data into an OWL Knowledge Base via the DBOM Protégé Plug-in[EB/OL]. (2010-11-21). <http://www.mendeley.com/research/integrating-data-owl-knowledge-base-via-dbom-prot-eg-e-plugin/>.
- [6] Pakornpong P. Query Answering for Multiple Complex Resources: Description Logic in the Semantic Web Context[D]. [S. l.]: University of Queensland, 2007.
- [7] Alferes J J, Matthias K, Terrance S. Queries to Hybrid MKNF Knowledge Bases Through Oracular Tabling[C]//Proc. of the 8th International Semantic Web Conference. [S. l.]: Springer-Verlag, 2009.
- [8] Guo Yuanbo. Lehigh University Benchmark[EB/OL]. (2010-11-21). <http://swat.cse.lehigh.edu/project/lubm>.

编辑 刘冰