

文章编号: 1002-0411(2000)01-082-04

基于 RDBMS 的知识表示及其实现

刘栋梁 高国安

(哈尔滨工业大学现代生产技术中心 哈尔滨 150001)

摘 要: 本文描述了目前几种常用的知识表示形式, 介绍了利用 RDBMS 表达存储知识的方法, 并在 ORACLE 环境下给出其相应的实现形式。

关键词: RDBMS, 知识表示, 知识库

中图分类号: TP182

文献标识码: B

1 引言

随着计算机科学的发展, 知识工程技术已经成为人工智能(AI) 一个重要的研究领域. 通过知识工程技术建立的各种智能应用系统在各个行业都获得广泛应用. 知识表示是知识工程中一个非常重要的问题, 其成功与否直接关系到智能系统的水平^[1], 因此, 必须实现对知识有效的表示机制以充分利用现有的宝贵资源, 更好地为人类探索未知(或未知)世界提供支持.

关系数据库是目前非常流行的数据管理工具, 有一套完整的关系理论基础, 并在实际应用中取得很大发展^[2], 其强大的数据存储、检索、组织及管理的能力正是当前知识库管理系统(KBMS)所不具备的. 如果能够充分利用关系数据库强大的管理能力构造管理 KBMS, 那么无疑会有利于知识的共享和检索效率, 进而提高推理能力^[3].

本文正是以此为出发点研究在 RDBMS 下实现知识表示的机制方法.

2 知识表示形式

知识表示是对人类已有知识的形式化描述, 其主要表示形式有以下 6 种^[4]:

(1) 产生式规则

产生式规则是一种应用最广泛的知识表示形式. 一般表示形式为: if A then B, 记为 $A \rightarrow B$. 它主要由条件和结论构成, 其特点为: ①相同条件可以得出不同的结论; ②不同条件可以得出相同结论; ③可以有多个条件, 但只有一个结论; ④规则中的条件可以作为另一规则的结论.

(2) 语义网络

这种表示形式类似于人的联想记忆模型, 由网络节点和连接节点间的弧构成, 其一般形式为节点及节点间的弧线构成的网络图. 其中网络节点代表概念, 节点间的弧线代表概念间的联系. 语义网络概念间的关系主要有四类:

- ① 分类: 表示个体的值与型的关系, 用 instance_of 表示;
- ② 聚集: 用于高层概念的分解, 是型之间或值之间的关系, 用 part_of 表示;
- ③ 泛化: 表示事物的属概念和种概念之间的关系, 用 is_a 表示;
- ④ 联合: 表示个体概念与整体概念之间的关系, 用 member_of 表示.

(3) 框架

框架是由一组描述物体的各个方面的槽(属性)组成, 每个槽又可包含多个侧面, 各侧面均有自己的名字和填入的值, 其一般定义结构为:

框架(FRAME)::= <槽名> <槽值> | <侧面名 侧面值> {, <侧面名 侧面值> }{,
<槽名> <槽值> | <侧面名 侧面值> {, <侧面名 侧面值> } }

(4) 谓词逻辑

谓词逻辑是唯一的具有公式形式的知识表示方法, 其基本组成部分是谓词符号, 变量, 常量和函数, 简单原子公式的形式为: $P(A_1, A_2, \dots)$, 其中 P 为谓词符号, 表示 A_1, A_2, \dots 之间的关系.

(5) 剧本(SCRIPT)

SCRIPT 是以一组框架描述一个事件序列, 每个框架描述一个事件, 一组框架表达一个知识. 其特点是能够表达现实世界中发生的起因、因果关系及事件间的联系, 从而构成一个大的因果链.

(6) 过程性知识

用于表示比较确定的知识. 其一般形式为描述性文件或计算程序.

上面介绍的都是符号化的知识表示形式, 此外也有非符号化的表示方法(如神经网络法), 这些知识表示方法都是目前运用较成功的, 我们主要针对上述 6 种符号化表示形式讨论其基于 RDBMS 的表示机制.

3 基于 RDBMS 的知识表示

RDBMS 是目前发展最为成熟的数据库管理系统, 其特点是信息共享程度高、表达完备、一致性完整性约束强. 而知识作为信息加工提炼的结果, 以数据作为知识载体, 因此, 可以利用数据管理技术实现知识的管理, 这不仅满足智能系统对知识的特性要求, 而且知识与数据的一致性管理有助于知识的获取及共享. 下面就以 ORACLE 数据库为例针对不同的知识表示形式给出其对应的关系模式表示.

3.1 产生式规则的关系表示

在 ORACLE RDBMS 中, 我们分别建立了与产生式规则中条件和结论相对应的实体表 TBL_CONDITION 和 TBL_CONCLUSION, 它们分别记录了产生式的条件集和结论集编码; 而产生式规则通过建立在上述两表之上的关系实体表实现, 其结构为:

TBL_RULE(RULE_ID, CONDITION_ID, CONCLUSION_ID)

其中 RULE_ID 为产生式规则的编号, CONDITION_ID 和 CONCLUSION_ID 是该产生式规则的条件编码和结论编码, 每个产生式规则对应于表 TBL_RULE 中的 n 个记录($n > = 1$), 这些记录间的成组关系对应于产生式规则条件之间的“与”(AND)连接, 而产生式规则的“或”(OR)连接都可以转换为 AND 连接形式表达.

产生式规则的关系语法结构如下:

```
CREATE TABLE TBL_CONDITION
  (CONDITION_ID      CHAR(10) NOT NULL,
   CONDITION_DESC    VARCHAR2(20),
   PRIMARY KEY(CONDITION_ID));
```

```

CREATE TABLE TBL_CONCLUSION
  (CONCLUSION_ID      CHAR(10) NOT NULL,
   CONCLUSION_DESC   VARCHAR2(20),
   PRIMARY KEY(CONCLUSION_ID));
CREATE TABLE TBL_RULE
  (RULE_ID           NUMBER(9),
   CONDITION_ID      CHAR(10)      NOT NULL,
   CONDITION_DESC    VARCHAR2(20) NOT NULL,
   PRIMARY KEY(CONDITION_ID, CONCLUSION_ID),
   FOREIGN KEY(CONDITION_ID) REFERENCES TBL_CONDITION(CONDITION_ID),
   FOREIGN KEY(CONCLUSION_ID) REFERENCES TBL_CONCLUSION(CONCLUSION_ID));

```

利用 ORACLE RDBMS 提供的层次查询可以很容易地实现产生式规则的正、反向推理。

3.2 语义网络的关系表示

语义网络的知识表示可以设计为对应于 RDBMS 中两个实体表 TBL_NODE、TBL_SEMANTICNET, 其中 TBL_NODE 为辅助表, 用于存放语义网络中的概念, 如果概念比较多, 可以考虑分片或分库存储策略; TBL_SEMANTICNET 对应于语义网络表示形式, 结构为 TBL_SEMANTICNET(SEMANTICNET_ID, RELATION_ID, NODE_ID1, NODE_ID2), 其属性分别为语义网络标识、以弧表示的关系标识及用弧连接的两个概念节点标识。这样, 利用 SQL 语法可以比较方便地完成对语义网络及网络节点间关系的检索, 实现语义网络的联想推理过程。

3.3 框架的关系表示

同上述表示方法一样, 框架的关系表示也应有三个辅助表: TBL_FRAME、TBL_SLOT 和 TBL_FACET, 它们分别对应框架、槽和侧面属性。这三个表中分别定义了各自的标识和名称, 以便于在表示框架结构的实体表 TBL_FRAMESTRU 中使用。此外, 表 TBL_SLOT 和表 TBL_FACET 中还含有槽及槽侧面属性的类型值标识, 以便能够正确理解框架结构表中槽或槽侧面的取值性质(具体值、过程或框架名等)。

框架结构为 TBL_FRAMESTRU (FRAME_ID, SLOT_ID, FACET_ID, SLOTORFACET_VALUE), 其主键为 FRAME_ID+ SLOT_ID+ FACET_ID, 当 FACET_ID 为无效值时, SLOTORFACET_VALUE 为槽值, 否则为侧面属性值。SLOTORFACET_VALUE 类型为字符型, 根据 SLOT 或 FACET 的类型标识决定其转换的目标值。这种表示方法中, 框架的匹配与继承可通过 SQL 语句的条件组合和过滤来实现。

3.4 谓词逻辑的关系表示

谓词逻辑的关系描述比较直观, 我们可以将一个谓词对应于一个实体表, 而表的元组对应于相应谓词的一个原子公式, 表的属性为原子公式变量集合。

3.5 剧本的关系表示

我们通过构造实体表的逻辑层次关系描述剧本知识。在上层表 TBL_SEQSET(ROLE, SEQUENCE, CONDITION, CONCLUSION) 中描述了剧本事件的主体、事件发生的顺序、发生的条件及结局; 而在下层表 TBL_EVENTSET(SEQUENCE, EVENT, EVENT_OBJ) 中通过上层表中事件发生的顺序 SEQUENCE, 检索出发生的事件 EVENT 及其对象 EVENT_

OBJECT. 通过两个层次表间的检索完成剧本推理.

3.6 过程性知识的关系表示

过程性知识的表示可以通过在数据库中建立过程性知识索引完成, 对应描述性的知识可以利用 ORACLE RDBMS 中的 LONG RAW 数据类型完成, 而对于计算性知识可利用 ORACLE 提供的存储过程实现.

以上叙述了几种主要知识表示形式在 ORACLE RDBMS 中的实现方法. 可以看到, 利用关系数据库 RDBMS 表示知识后, 可以充分利用数据库强大的检索优化能力, 结合 SQL 语句实现知识的推理过程.

4 结论

本文简要介绍知识表示的几种形式并结合 RDBMS 的功能特性讨论了利用 RDBMS 实现知识表示的方法. 可以认为, 基于 RDBMS 表示的知识库能够充分利用 RDBMS 的优点, 加强知识库的管理, 同时也有利于数据与知识的统一管理. 本文的讨论只是针对知识表示的, 在 RDBMS 中如何保证知识的有效性、减少知识的冗余及加强知识的推理能力还需要进一步的研究.

参 考 文 献

- 1 史忠植. 知识工程. 清华大学出版社, 1988
- 2 董逸生等. CIMS 中的数据库技术. 机械工业出版社, 1997
- 3 T P Liang. Integrating Model Management with Data Management in Decision Support Systems, Decision Support Systems, 1985, (1)
- 4 陈屹等. 实用专家系统指南. 科海总公司培训中心, 1991

KNOWLEDGE REPRESENTATION AND REALIZATION BASED ON RELATIONAL DATABASE MANAGEMENT SYSTEM (RDBMS)

LIU Dong-liang GAO Guo-an

(Modern Production Technology Centren, Harbin Institute of Techonology 150001)

Abstract: This paper describes some popular know ledge representation form s and introduces a method of know ledge representation based on relational database m anagement system (RDBM S). The implem entation of know ledge representation in Oracle is also given in this paper.

Keywords: RDBMS, know ledge base, know ledge representation

作者简介

刘栋梁(1969-), 博士研究生. 研究领域为网络数据库, 决策支持, 知识工程.

高国安(1943-), 博士生导师, CIMS 专家组成员. 研究领域为计算机 CAD 与计算机图形学, 决策支持, CIMS 工程, 专家系统.