一种基于数据表元组的原始本体抽取方法①

A Method of Semi-Finished Ontology Extraction from Tuples of Relational Table

唐颖峰　周肆清　(中南大学 信息科学与工程学院 湖南 长沙 410083)

摘　要： 在基于关系数据库的本体构建过程中，由数据表抽取的原始本体的质量直接影响到最终生成本体的质量，而传统的抽取方法中将单个数据表映射为单个本体概念的方法忽略了数据表中元组数据所提供的语义信息，造成抽取的原始本体质量不高的问题。本文提出一种了基于数据表的原始本体抽取方法。应用 FCA（形式概念分析）方法对单个数据表的元组数据进行分析，形成概念格，进而产生原始本体。该方法使得数据表中的元组数据得到了有效的利用，提高了原始本体的抽取质量，有利于最终本体生成质量的提高。

关键词： 数据表元组 原始本体 本体抽取 FCA 关系数据库

在传统的基于关系数据库的本体抽取过程中[2,8]，数据表被映射为单个概念节点，即一个数据表对应单个的本体概念。这种做法存在一定的缺陷。 首先忽略了数据表中元组数据的语义信息，造成了数据的浪费。其次从单个表抽取的本体概念缺乏语义信息，影响了最终生成的本体的质量。

本文提出一种基于数据表的本体抽取方法。将单个本体概念作为只含有一个概念的原始本体，从而将由数据表抽取的本体概念范畴扩展为原始本体范畴。对于实际应用数据库中大量存在的多字段数据表应用 FCA（形式概念分析）方法对表中的元组数据进行分析得到概念格，然后再将概念格转化为原始本体。

1　基于关系数据库的本体构建

基于关系数据库的本体构建过程主要分为三大部分：原始本体生成、原始本体的合并以及原始本体的集成。

原始本体来源于数据表。将数据表按照需求进行预处理，生成数据表实体。然后按照一定规则从数据表实体中抽取信息，生成原始本体。最终由整个数据库资源生成原始本体集。

仅生成原始本体集还远远不够，一方面仅仅由零散的、缺乏关系标注的原始本体构成的本体集合基本是无用的；另一方面用户使用由多个关系数据库生成的多个本体也是不方便的。因此我们不仅需要把原始本体集进行合并，而且还需要将从不同数据库生成的本体进行集成，生成一个语义一致的本体返回给用户。基于关系数据库的本体构建过程如图 1.1 所示。

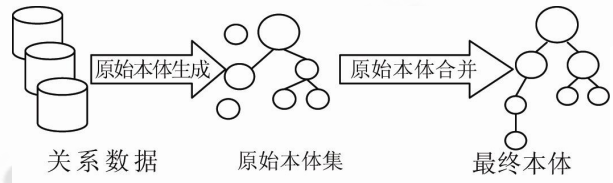

图 1　基于关系数据库的本体构建过程

本文主要就原始本体生成部分进行讨论。

2　形式概念分析

形式概念分析(Formal Concept Analysis，FCA)理论[3,4]，又叫概念格理论，是德国的 R.Wille 教授于 1982 年首先提出的。一种用数学的形式化语言来反映人形成概念的过程的集合理论模型，用来研究特定领域可能存在的概念的几何结构、概念格形式。作为数据分析和知识处理的形式化工具，形式概念分析

① 收稿时间:2008-08-31

已经在信息检索、数字图书馆、软件工程和知识发现等方面获得了广泛而成功的应用。

2.1 形式背景与形式概念

定义 2.1 一个形式背景（Formal Context）是一个三元组 K=(O,A,R)，其中 O 是对象的集合，A 是属性的集合，R 是 O 和 A 之间的一个二元关系，即，$R \subseteq O \times A$。

根据定义 2.1，一个形式背景能够用一个矩形表来表示，表的每一行是一个对象，每一列是一个属性。若 g 行 m 列的交叉处是 X，则表示对象 g 具有属性 m，表 1 所示为某应用系统工单数据表片断对应的形式背景。

表 1　一个形式背景的例子

元组/字段	字段 1	故障	咨询	投诉	已完成	处理中	处理时限	字段 2
1	X	X			X			
2			X			X	X	X
3				X		X		
4	X	X				X	X	

定义 2.2 给定对象集合 O，对于一个对象子集 $E \subseteq O$，定义 $E' = \{m \in A \mid \forall g \in O, gRm\}$，表示"E 中全体对象所共有的属性集"。相应地，对于一个属性集合 $I \subseteq A$，定义 $I' = \{g \in O \mid \forall m \in I, gRm\}$，表示"具有 I 中所有属性的对象的集合"。

定义 2.3 形式背景(O,A,R)的一个形式概念(Formal Concept)是(E,I)，其中 $E \in O$，$I \in A$，且满足 $E' = I$ 和 $I' = E$。我们称 E 是形式概念(E,I)的外延(extent)，I 是形式概念(E,I)的内涵(intent)。(O,A,R)表示形式背景 $\mathfrak{R}$ (O,A,R)的所有形式概念的集合。如果用 C 表示一个给定的形式概念，其内涵和外延也可以分别用 Intent(C)和 Extent(C)表示。

下面的步骤列出了从形式背景得到形式概念的方法：

1）选择对象 A；

2）得到 A 对象具有的所有属性集 A′；

3）得到具有属性集 A′ 的所有对象集 A″；

4）则(A″,A′)就是一个形式概念。

2.2 概念格和 Hasse 图

所有的概念连同定义在其上的层次关系共同构成了概念格，概念格是形式概念分析理论中的核心数据结构，概念格基于二元关系，体现了概念内涵和外延的统一，它从本质上描述了概念之间的泛化与特化关系，非常适合于发现数据中潜在的概念。形式概念分析理论利用其相应的 Hasse 图实现了概念层次的可视化。

定义 2.4 设<H,≤>为偏序集，对于任意的 $B \in H$，如果有 $a \in H$，并且对 B 的任意元素 x，都满足 $x \leq a$，则称 a 为子集 B 的上界。同理，如果对 B 的任意元素 x，都满足 $a \leq x$，则称 a 为子集 B 的下界。

定义 2.5 设<H,≤>是一个偏序集，如果 H 中任意两个元素都有最小上界和最大下界，则称<H,≤>为格。

定义 2.6 对于形式背景 K=(O,A,R)，存在唯一的一个偏序集<H,≤>与之对应，并且该偏序集存在一个唯一的下确界和一个唯一的上确界，这个偏序集产生的格结构称为概念格(concept lattice)，记为 L(O,A,R)。

概念格可以图形化形式表示为有标号的线图，图中的节点表示一个概念，节点间的连线表示节点间存在偏序关系。这种线图也称为 Hasse 图，它是概念格的可视化表示。图 1 所示的是表 1 形式背景对应概念格的 Hasse 图。由图可见，共有 9 个形式概念：工单 1({1,2,3,4},{})，工单 2({2,3,4},{处理中})，工单 3({1,4},{字段 1,故障})，工单 4({2,4},{处理中,处理时限})，工单 5({1},{字段 1,故障,已完成})，工单 6({3},{投诉,处理中})，工单 7({2},{咨询,处理中,处理时限,字段 2})，工单 8({4},{字段 1,故障,处理中,处理时限})，工单 9({},{字段 1,故障,咨询,投诉,已完成,处理中,处理时限,字段 2})

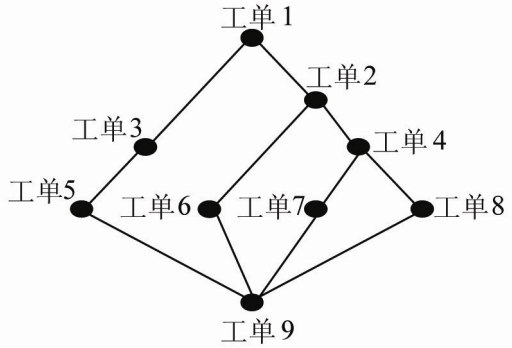

图 2　表 1 所示形式背景对应的 Hasse 图

2.3 概念格构造算法

概念格的构造过程实际上是概念聚类的过程。比较著名的概念格构造的算法有 Ganter's Next-Concept Algorithm[6], Christian Lindig 的 Fast Concept Analysis[6,7]等,这些算法计算出了格的所有概念以及层次关系。这里使用一种自底向上的计算方式计算出概念格。

变量说明:

N:形式背景中单个对象具备属性的最大数目,即是格中的继承层次数目,但不包括上下界所占的层次。

A:形式背景中单个属性或多个属性的集合。

A′：A 中全体对象所共有的属性集

Node:概念节点集。Node(i,j)表示第 i 层上的第 j 个节点。

算法:

(1)计算出 N;

(2)for(int i = N;i≥1;i --)

```
    {
      计算出 A;
      for(int j =1;j<sizeof(A);j++)
        {
          a = A′ (j);
          b=A(j);
          if(a′ == b&&a == b′ ) \
            {
            Node(i,j).attributelist=b;      ]
            Node(i,j).objectlist=a;
```

判断该attributelist(即b)是否包含于下层节点的 attributelist 中,确定节点间父子关系;

```
            }
        }
}
```

3 基于数据表的原始本体抽取

由于关系数据表是二维表结构，和 FCA 中的形式背景相似。根据这一特点，就可以利用 FCA 对数据表进行分析，生成概念格，进而得到原始本体。

但是直接对数据表进行 FCA 处理是不可行的。首先，并不是对数据库中所有的表都要进行 FCA 处理，对于字段单一数据量较少甚至没有数据或者没有概念细分需求的数据表，则无需进行 FCA 处理，直接生成单一概念的原始本体即可。其次，对数据表的字段来说，那些无值字段或者对概念细分没有意义的字段则需要进行裁减。再次，数据表中的数据往往是多值的，而 FCA 中形式背景则是单值的。为了能够运用 FCA 方法处理数据表，还需要将数据表由多值转为单值。最后，为了减少与数据库连接的次数，提高运行效率，需要把表信息存于外部文件中，方便后期处理。

因此，可以把原始本体抽取的过程分为三步，即数据表预处理、生成概念格以及生成原始本体。

3.1 数据表预处理

将数据表预处理后的结果以某种结构保存于文件中，称其为表实体。而表实体所对应的数据可以保存在 XML 文件中，称其为对应表实体的数据实体。表实体的数据结构定义为：实体名，属性描述，数据实体名以及相关实体名。其中属性分为一般属性，无用属性以及标记属性。属性描述包括：属性名、属性类型和集的势。表实体的数据结构如图 3 所示。

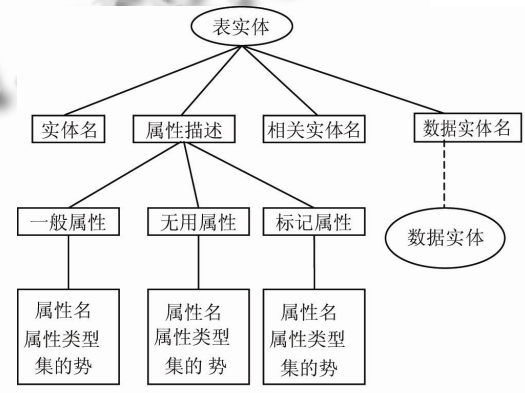

图 3　表实体的数据结构示意图

3.1.1 表实体的生成

一个表实体由一个数据表生成。实体名可以由用户自定义，缺省情况为数据表名。相关实体名通过分析数据表的外键约束得到。由无值字段生成无用属性，标记性字段生成标记属性，其它字段生成一般属性。属性名和属性类型分别来自于字段名和字段类型。集的势可以由字段的约束信息得到，但考虑到实际应用中数据列是否可以为空是由应用程序进行限制，因此可以通过元组数据信息决定：空列为-1，存在空值且不全为空的列为 0，不存在空值的列为 1。

数据实体名的生成分为两种情况：若对当前数据表不进行 FCA 处理，则标记为 NULL。否则生成数据实体，并标记数据实体名。

3.1.2 数据实体的生成

对于要进行 FCA 处理的数据表，需要将其元组数据进行裁减及变换，生成数据实体。

数据列的裁剪主要是将对FCA处理无意义的列进行删除。这个过程可以由程序根据筛选条件自动完成，也可由人工筛选。

数据表的变换是将数据表的多值结构转换为与 FCA 中形式背景对应的单值结构。数据表的转换有两种做法，一种是直接将列的多值数据转换为单值标记。

这种做法适合于只需关心其有值与否而并不关心其具体数据的列。另一种做法是，将一列数据按值分类，然后将分类结果生成新的列并将原有列移除，将元组在对应的新列上进行标记。后一种做法适合于其数据值较少的列，若数据值较多，则可以对数据值按一定标准进行聚类，然后将聚类结果作为新的列。

下面结合具体实例来说明数据表的变换过程。

表 3　某应用系统工单处理表片断

元组/字段	字段 1	工单类型	处理状态	处理时限	字段 2
1	X	故障	已完成		
2		咨询	处理中	24	X
3		投诉	处理中		
4	X	故障	处理中	48	

表 3 所示是一张某应用系统中的工单处理表，共有 5 个字段，其中字段 1 和字段 2 是单值的，而工单类型、处理状态及处理时限三个字段则是多值的。按照上述方法进行转换，其中处理时限按照第一种方式转换，工单类型和处理状态按照第二种方式转换。转换后的数据表如表 2 所示。

最后将经过裁减和变换的元组数据存于 XML 文件中，生成一个数据实体。

3.2 生成概念格

利用上述概念格生成算法将数据实体生成概念格。

3.3 生成原始本体

原始本体的生成要考虑两种情况：对于没有对应数据实体的表实体，可直接生成只含有单个概念的原始本体。对于存在数据实体的表实体，则需要结合对应的概念格生成原始本体。

由概念格生成原始本体只需通过直接删除概念格最底层元素，可以将其转换成偏序关系，将生成的形式概念作为本体的概念。本体概念的命名则可以由设计人员根据概念的内涵及设计需求进行命名。由图 2 所示概念格生成的原始本体如图 4 所示。

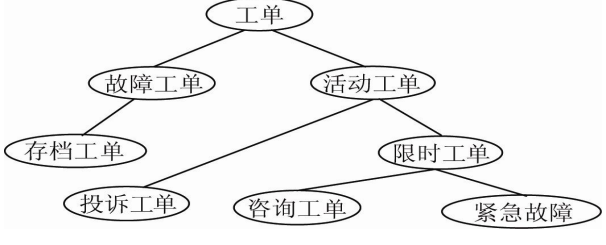

图 4　由概念格生成的原始本体

由于实际应用中元组数据不准确或不完整等局限性，生成的原始本体可能与实际理解有偏差，这时可以由设计人员根据需求或者实际理解对原始本体进行修正。修正后的原始本体如图 5 所示。

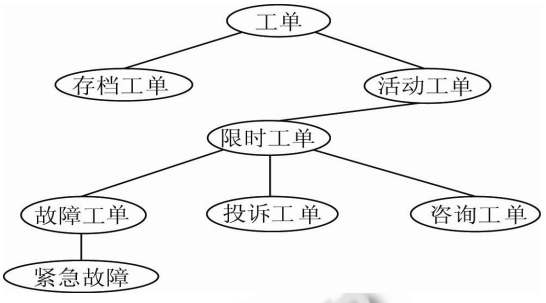

图 5　经过修正后的原始本体

4　结论

本文就基于关系数据表的本体抽取过程中的问题，将抽取本体概念范畴拓展为原始本体范畴，提出了一种原始本体的抽取方法。该方法使得数据表中的元组数据得到了有效的利用，提高了原始本体的抽取质量，有利于最终本体生成质量的提高。

原始本体集和传统的抽取方法生成的单个概念构成的集合有着较大的差异，因而传统的概念集成方法将不适用于原始本体的合并。因此，找到一个适合于原始本体合并的方法，将原始本体有效的合并集成，将是下一步工作的主要内容。

参考文献

1 Yuri A, David W. Ontology generation from tables. Proceedings of the Fourth International Conference on Web Information Systems Engineering (WISE'03). 2003:520－529.

2 刘勇军,聂规划.多信息源下本体自动抽取的实现.计算机应用研究,2007,24(11):183－187.

3 Wille R. Restructuring lattice theory: an approach based on hierarchies of concepts.In:I.Rival,eds. Ordered Sets. Dordrecht, Reidel, 1982, 445－470

4 Ganter B, Wille R. Applied lattice theory: Formal ConceptAnalysis.http://www.math.tu-dresden.de/~ganter/ps-files/concept.ps,1996.

5 谢志鹏,刘宗田.概念格与关联规则发现.计算机研究与发展,2000,37(12):1416－1421.

6 Lindig C. Introduction to Concept Analysis. http://www.st.cs.uni-sb.de/~lindig/talks/fca-intro/slides.pdf, 2002.

7 Lindig C. Fast Concept Analysis. http://www.st.cs.uni-sb.de/~lindig/papers/fast-ca/iccs-lindig.pdf, 2002.

8 韩石.基于关系数据库的本体构建方法的研究 [硕士学位论文]. 哈尔滨:哈尔滨工业大学,2006.