文章编号: 1000-6788(2011)S2-0197-06 中图分类号: TP301 文献标志码: A

形式背景的属性树表示

张 涛[1,2], 洪文学[2], 路 静[1]

(1. 燕山大学 信息科学与工程学院, 秦皇岛 066004; 2. 燕山大学 电气工程学院, 秦皇岛 066004)

摘 要 给出了一种形式概念分析形式背景的属性树的表示方法. 该方法首先利用属性间的包容与互斥关系等属性特征对属性进行划分, 然后根据划分结果将形式背景从属性角度构造属性包含森林. 从属性包含森林中找到表示属性主要关系的主树, 最后以主树为基础结构构造形式背景的属性树表示. 实验表明, 属性树表示方法在保留概念格方法对形式背景偏序描述关系的同时, 简化了形式背景的表示结构, 通过增加变尺度特性和隐含属性的定义强化了信息挖掘与规则发现能力, 为形式概念分析提供了新的方法.

关键词 形式概念分析; 形式背景; 属性树; 信息挖掘; 规则发现

Attribute tree representation for formal context

ZHANG Tao[1,2], HONG Wen-xue[2], LU Jing[1]

(1. College of Information Engineering, Yanshan University, Qinhuangdao 066004, China;
2. College of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China)

Abstract A novel method for representing the formal context named attribute tree is proposed. In this method, the attributes are grouped by the relationship of compatibility and exclusive between attribute pairs, which is the basis of constructing attrubite forest from formal context. The next step is to obtain the main tree in forest which reprents the fundmental relationship of the context and the last is combining the attribute tree using the main tree structure. The expermint shows, the novel method is represented as a tree structure, which inherits the ordering from concept lattice and simplifies the structure for representation. Except that, the multiscale and hidden attributes is introduced, this strengthens the information mining and rule discovery. All these merits provide a new approach for formal concept analysis.

Keywords formal concept analysis; formal context; attribute tree; information mining; rule discovery

1 引言

形式概念分析是以数学化的概念和概念层次结构为基础的应用数学领域. 自 1982 提出以来 [1], 形式概念分析被深入研究并取得许多重要的成果, 被认为是知识表示和知识处理的一种有效技术, 在数据管理 [2−3]、机器学习 [4] 和软件工程 [5−6] 等领域得到了广泛的应用.

目前对于形式背景的表示主要有两种形式. 一种为属性对象的关系表, 该方法以表格形式对形式背景进行表示; 另一种以 Hasse 图为主的概念格方法, 利用格理论, 结合关系中的偏序关系进行背景表示 [3]. 作为格理论的分支, 概念格的主要优点在于可以将数据中内在逻辑和组织结构完整地图示化, 从而为分析概念数据之间的关联提供系统的可视化工具. 显然, 概念格方法在结构和包含关系上具有更大的优势. 但随着形式背景的复杂化, 背景的概念格网络结构复杂度以指数级数递增, 影响对形式背景的表示和知识获取 [2]. 因此, 需要一种以结构简单却继承了概念格包含关系的形式背景表示方法.

基于此, 本文构造了形式背景的属性树表示新方法. 与采用图的逻辑关系作为基础数据结构的概念格不同, 属性树表示采用了树作为基础数据结构. 该方法首先研究属性的特征, 以属性的包含关系作为形式背景

收稿日期: 2011-07-01
资助项目: 国家自然科学基金 (60904100, 61074130); 河北省自然科学基金 (F2011203073); 秦皇岛市科学技术研究与发展计划 (201001A051)
作者简介: 张涛 (1979–), 男, 讲师, 研究方向: 智能信息处理; 洪文学 (1952–), 男, 教授, 博士生导师, 研究方向: 形式概念分析、可视化模式识别; 路静 (1985–), 女, 硕士研究生, 研究方向: 形式概念分析.

的切入点, 以树的形式进行表示, 有效的避免的属性关系的交叉, 并可以将变尺度特性与属性间的包含、相容、互斥等进行直观表示, 同时可以发现背景中的隐含属性, 对形式背景进行知识发现.

2 基于包容关系的属性分类

对形式背景的表示是形式概念分析研究的基本内容之一. 与基于偏序关系的概念格表示不同, 本文的属性树表示从属性的特征入手, 即利用的是属性间的相容、包含与互斥关系完成对形式背景的刻画. 为了保证属性树方法的系统性, 从属性间的包含角度, 对属性进行如下定义与分类.

定义 1 全覆盖属性: 在形式背景 $K=(G,M,I)$ 中, 设 $a\in M$ 且 $a'=G$, 则称 a 为形式背景 K 的全覆盖属性. 形式背景 K 的全覆盖属性集合用 $Cover(K)$ 表示.

定义 2 尺度属性: 在形式背景 $K=(G,M,I)$ 中, $\exists a\in M$, 且 $\exists c\in M$, 若满足 $a'\subset c'$, 则称属性 c 为属性 a 的上尺度属性, 记作 $c=sup(a)$; 属性 a 为属性 c 的下尺度属性, $a=sub(c)$. 具有下尺度属性的属性称为父属性. 若不存在 $b\in M$, 使得 $a'\subseteq b'$ 且 $b'\subseteq c'$, 则称 a 为 c 的直接下尺度属性, 记作 $a=dsub(c)$, c 为 a 的直接上尺度属性, 记作 $c=dsup(a)$.

定义 3 在形式背景 $K=(G,M,I)$ 中, 若属性集合 $A=\{a_i|a_i\in M\}$ 为属性 c 的直接下尺度属性集合, 则 A 中元素 a_i 互为同级属性. 若属性 a、b 互为同级属性且 $a'\cap b'=\Phi$, 则称 a、b 在该级下为互斥属性.

基于以上关于属性的定义, 可得

性质 1 在形式背景 $K=(G,M,I)$ 中, 设 c 为 a、b 的直接上尺度属性, a、b 为互斥属性, $a'\cup b'=c'$, 则:

1) 若 $\exists g\in G$, 使得 $c'\subseteq g$, 则必有 $a'\subseteq g$ 或 $b'\subseteq g$;

2) 若 $\exists g\in G$, 使得 $g\in a'$, 则必有 $g\notin b'$ 且 $g\in c'$.

证明 1) 由条件可知, $a=dsub(c)$, $b=dsub(c)$, 由尺度属性定义可知, $a'\subset c'$, $b'\subset c'$. 又因为 $c'\subseteq g$, 有 $a'\subset c'\subseteq g$, $a'\subset c'\subseteq g$. $\Rightarrow a'\subseteq g$, $\Rightarrow b'\subseteq g$.

又由于 $a'\cap b'=\Phi$, a' 与 b' 互斥, $\Rightarrow a'\subseteq g$ 与 $b'\subseteq g$ 不能同时出现. $\Rightarrow a'\subseteq g$ 或 $b'\subseteq g$.

2) 由于 $a'\cap b'=\Phi$, $\Rightarrow g\in a'\Rightarrow g\notin b'$, 由于 $c=dsup(a)$, $\Rightarrow a'\subset c'\Rightarrow g\in a'\subset c'\Rightarrow g\subset c'$.

性质 2 属性尺度具有传递性.

证明 在形式背景 $K=(G,M,I)$ 中, 设 $\exists a,b,c\in M$. 若 $c=sup(a)$, $d=sup(c)$, $\Rightarrow a'\subset c', c'\subset d'$.

依集合传递性, 有 $a'\subset c'\subset d'$, 有 $a'\subset d'\Rightarrow d=sup(a)$. 同理 $a=sub(d)$. 因此, 属性尺度具有传递性.

定义 4 独立属性: 在形式背景 $K=(G,M,I)$ 中, 设 $a\subset M-Cover(K)$, 若不存在 $c\in M$, 使得 $c'\subseteq a'$, 则称 a 为当前背景下的独立属性. 若 $\exists b\in M$ 且 $b\neq \sup(a)$, 使得 $a'\cap b'\neq \Phi$, 则称 a 为背景 K 的一般独立属性, 否则称为最小独立属性.

定义 5 隐含属性: 在形式背景 $K=(G,M,I)$ 中, $A=\{a_i|a_i=d\sup(c),a_i\in M\}$, 若 $b'=c'-\bigcup\limits_i a_i'$ 且 $b\notin M$, 则称 b 为属性 c 中的隐含属性, 记作 $b=HideIn(c,K)$.

性质 3 形式背景 $K=(G,M,I)$ 中, 对于属性 $c\in M$ 具有隐含属性 b, 则属性 c 必具有下尺度属性.

证明 由隐含属性定义可知, 对隐含属性 $b=HideIn(c,K)$, $b'=c'-\bigcup\limits_i a_i'$.

设 c 不具有下尺度属性, 则 $A=\Phi$ 有 $\bigcup\limits_i a_i'=\Phi\Rightarrow b'=c'-\bigcup\limits_i a_i'=c'-\Phi=c'\Leftrightarrow b=c$. 又由于 $c\in M\Rightarrow b\in M$ 与隐含属性定义中 $b\notin M$ 矛盾, 假设条件不成立. 因此, c 必然具有下尺度属性.

定义 6 在形式背景 $K=(G,M,I)$ 中, $A=\{a_i|a_i\in M\}$ 为属性 c 的直接下尺度属性集合, 若 $c'=\bigcup\limits_{i=1}^{n} a_i'$, 则称 c 为完全属性. 若 $c'\subset\bigcup\limits_{i=1}^{n} a_i'$, 则称 c 为准完全属性.

3 属性树的构造

第 2 节从属性间的包容关系定义了形式背景中的属性关系及其性质. 本节利用这些定义进行属性树的构造, 从而对形式背景中的各个属性进行基于包容关系的可视化表示, 为后续应用奠定基础.

3.1 形式背景的包含分解

定义 7 形式背景的包含分解: 对形式背景 $K=(G,M,I)$, $\#\{M\}=n$, 可以将其分解为 m 个子背景 $\{K_i|i=1,2,\cdots,m,m\leq n\}$, K_i 满足以下关系:

1) $K_i=(G_i,M_i,I_i)$, 其中 $G_i=\{A_k|k=0,1,2,\cdots\}'$, $M_i=\{A_k|k=0,1,2,\cdots\}$, $I_i=\{A_k|k=0,1,2,\cdots\}'\times\{A_k|k=0,1,2,\cdots\}$;

2) $A_{k+1}=\{a_i|a_i=dsub(b_i),b_i\in A_k,a_i\in M-\bigcup_{l=0}^{k}A_l,a_i\cap a_j=\Phi\}$;

3) $A_0=\{a_i|a_i=Cover(K),a_i\in M\}$.

由于属性包含分解的基础为属性间的包容关系, 因此对于无包容关系的形式背景, 无法进行属性包含分解. 显然, 对形式背景 $K=(G,M,I)$, 分解后的子背景 $\{K_i|i=1,2,\cdots m,m\leq n\}$, 有 $\#\{K_i\}\leq\#\{M\}$ 且背景分解结果不唯一.

3.2 属性包含树与属性包含森林

根据 3.1 小节中的背景分解结果可知, 对一个形式背景可以进行属性包含分解, 分解后的每个子背景均可用树的结构进行描述. 以下定义属性包含树与属性包含森林, 以完成最终的形式背景表示.

定义 8 属性包含树: 属性包含树 T 是包括 n 个节点的有限非空集合 D,R 是 D 中元素的序偶的集合, D 和 R 满足以下特性:

1) 对形式背景 $K=(G,M,I)$, $D\subseteq M$;

2) 有且仅有一个节点 $r\in D$, 不存在任何节点 $v\in D$, $v\neq r$, 使得 $\langle v,r\rangle\in R$. r 为树的根;

3) 除根 r 以外的所有节点 $u\in D$, 都有且仅有一个节点 $v\in D$, $v\neq u$, 使得 $<v,u>\in R$.

属性包含树可以描述包含分解后的子背景. 但由于一个形式背景分解后可能形成多个子背景, 因此对于整个形式背景的描述可以利用属性包含森林描述.

定义 9 属性包含森林: 对于一个形式背景 $K=(G,M,I)$, 将其包含分解后子背景集合为 $\{K_i|i=1,2,\cdots,m\}$, 其对应的属性包含树集合 $F=\{T_i|i=1,2,\cdots,m\}$ 构成属性包含森林. 在属性包含森林中, 高度最大、节点最多的树称为森林的主树, 其他树称为辅树. 辅树中包含的属性称为辅助属性.

性质 4 在属性包含树中, 根节点必为全覆盖属性. 除根节点外, 每个节点只有一个父节点, 该父节点为当前节点的直接上尺度属性.

证明 属性包含树为树状结构, 因此其符合树的定义, 即每个节点只包含一个父节点. 由背景的包含分解可知, 任一节点的子节点一定是该节点的直接下尺度属性, 任一节点的父节点一定是该节点的直接上尺度属性. 由于每个子背景均包含 $Cover(K)$, 而 $Cover(K)$ 为任意属性的上尺度属性. 因此其必为根节点.

性质 5 在属性包含树中, 兄弟节点互斥, 父子节点具有偏序关系.

证明 对形式背景 $K=(G,M,I)$, 兄弟节点互斥可由定义直接推得, 证明略.

设节点 b 为节点 a 的子节点. $\Rightarrow b=dsub(a)\Rightarrow b'\subset a'$ 若取 $g\in G$ 且 $g\in b'$, 则 $\forall m\in b$, 有 gIm.

$b'\subset a'\Rightarrow\forall m\in a$, 都有 $gIm\Rightarrow g\in a'$, 有 $a\subset b\Rightarrow(a',a)$ 形成对 (b',b) 的超概念 $\Rightarrow(b',b)\leq(a',a),<b,a>$ 具有偏序关系 $\Rightarrow$ 父子节点间具有偏序关系.

性质 6 对同一形式背景, 其属性包含森林不唯一.

证明 由于属性包含森林与属性包含分解一一对应, 又由于属性的包含分解不唯一, 因此其属性包含森林不唯一.

3.3 属性树的建立

由 3.2 小节可知, 通过对形式背景的包含分解与分析, 可以将一个形式背景以属性包含森林的形式进行表示. 该过程实质是对形式背景的分解. 属性树的建立, 是要将属性森林归纳为树的形式, 从而利用一个图表示整个形式背景. 属性树的建立过程, 可以看作是对子背景的合成过程.

由性质 6 可知, 对于同一个形式背景, 其属性包含森林不唯一. 因此需先对森林进行分析, 选择信息表示相对充分的森林进行背景合成, 寻找不同森林表示中的主森林.

定义 10 主森林: 对于一个形式背景 $K=(G,M,I)$, 设其共可以形成的 n 个属性包含森林, 其集合为 $\{F_i|i=1,2,\cdots n\}$. 其中森林 F_j 称为该形式背景的主森林, 满足 $F_j=\arg\min_i\#\{F_i\}$ 且 $F_j=\arg\max_i\left\{\arg\max_k\{\sum\text{degree}(MT(F_k))\}\right\}$.

该条件要求主森林中树的数目最少且主背景中主树包含节点最多, 实质上是要保证主树结构的最大化, 从而保证合成出的属性树具有最简表述.

从属性的包含角度看, 任两个属性 a、b 其关系只可能为:

1) 互斥: 即 $a' \cap b' = \Phi$. 在本文中, 用 $a \circ b = a \vee b \vee \Phi$, 表示互斥关系;

2) 相容: 即 $a' \cap b' \neq \Phi$. 在本文中, 用 $a \bullet b = (a \circ b) \vee ab = a \vee b \vee ab \vee \Phi$ 可简记为 ab, 表示相容关系.

定义 11 属性树: 对一个形式背景 $K = (G, M, I)$, 将其进行包含分解找到其主森林 F, 其主树 $T = MT(F)$. 则其属性树为以主树 T 为基本结构, 各辅助属性以 $\bullet$ 运算参与到主树结构中的数据结构表示.

显然, 在一个属性森林中, 主树与属性树同构. 由于包含关系的存在, 属性树中由根节点至叶子节点为包含关系, 即由抽象到具体的过程. 每个节点均可再展开为节点子树, 该节点子树为当前节点属性集合的小尺度表示.

属性树可以对形式背景中的显式数据进行表示. 为了信息挖掘的需要, 将形式背景中的隐含属性进行显式表示的属性树即成为扩展属性树. 显而易见, 属性树是扩展属性树的子结构.

3.4 属性树的生成过程

从一个形式背景, 到生成属性树, 可以分为以下几步:

步骤 1 对形式背景进行净化;

步骤 2 对净化后的背景进行包含分解, 并形成属性包含森林集合;

步骤 3 在属性包含森林集合中找到主森林;

步骤 4 对主森林中的主树为基本结构, 辅树与主树通过 $\bullet$ 运算连接, 形成属性树;

步骤 5 对主森林中的主树进行隐藏属性补齐, 并以其为基本结构, 辅树与主树通过 $\bullet$ 运算连接, 形成扩展属性树.

4 实例与讨论

为了说明属性树对形式背景的刻画, 本节以特定的形式背景为例进行讨论. 在讨论过程中, 利用概念格进行对比表示, 以此发现二者在对形式背景刻画中的不同.

生物和水的形式背景来自于匈牙利的科教电影 "生物和水", 是形式概念分析中最为经典的形式背景. 其如表 1 所示. 其概念格表示、属性树表示与扩展属性树表示分别如图 1-3 所示. 各图中字母符号与属性的对应表如表 2 所示.

由对比可以看到, 属性树和概念格的表示方法均利用了偏序关系. 所不同的是, 概念格从概念角度定义偏序, 并以此作为结构中的层序基础; 而属性树则是从属性的角度定义偏序, 并结合包含关系进行层级的划分. 由偏序理论可知, 相对于概念格, 属性树的偏序含义更为广泛, 因此具备更好的扩展性和适应性.

表 1 生物和水形式背景

对象	需要水	水里生活	陆地生活	有叶绿素	双子叶	单子叶	能运动	有四肢	哺乳
蚂蝗	x	x					x		
娃娃鱼	x	x					x	x	
蛙	x	x	x				x	x	
狗	x		x				x	x	x
水草	x	x		x		x			
芦苇	x	x	x	x		x			
豆	x		x	x	x				
玉米	x		x	x		x			

表 2 属性符号对照表

属性	需要水	水里生活	陆地生活	有叶绿素	双子叶	单子叶	能运动	有四肢	哺乳
符号	a	b	c	d	e	f	g	h	i

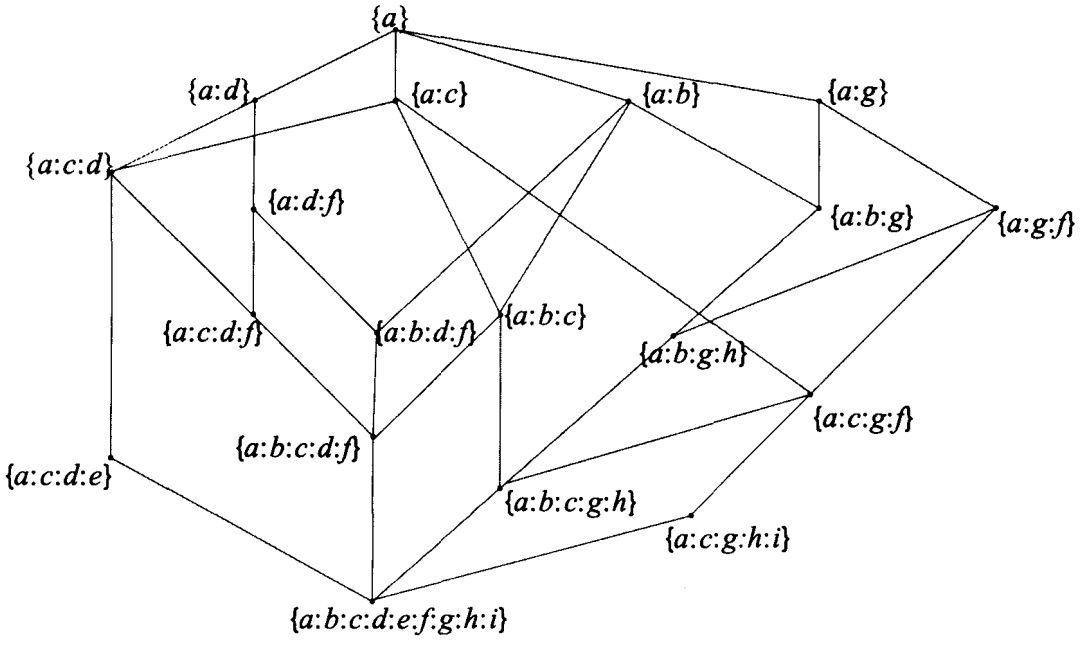

图 1　表 1 的概念格表示

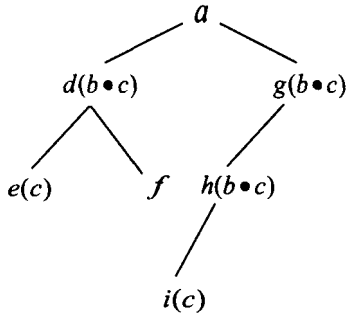

图 2　表 1 的属性树表示

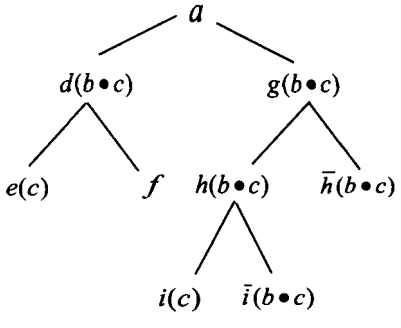

图 3　表 1 的扩展属性树表示

从表现形式上, 概念格从概念的网状结构进行背景内容的解析, 而属性树从整体看是对形式背景的树状分析. 两种不同的数据结构必然带来对同一形式背景的两种认知. 概念格强调的是偏序关系, 而属性树在广义偏序基础上, 强调的是属性间包含与互斥形成的划分. 另外, 由于属性树方法将形式背景的表示由概念格的图方法改为了树方法, 在结构上不存在横向交叉, 在对大规模背景的分析上具有更好的可视化效果的同时, 具备了属性的自然分级特性, 可以利用树的理论进行继承与同级的分析.

从节点的构成看, 概念格的每个节点表示一个概念, 其节点总数与形式背景中形成的形式概念数目一致. 而属性树的节点则表示一种运算, 其表示在当前尺度下可构成关系的属性集合与集合间关系. 因此, 从整体节点数上看, 属性树节点数目较少, 适合于大规模数据集的表示. 且属性树强调的是属性间关系, 而不强调概念关系, 因此为信息的进一步挖掘提供了基础.

从信息挖掘深度看, 概念格方式可以对现有背景进行规则发现. 但由于其局限于对形式概念的表述, 挖掘深度不足. 而属性树通过对隐藏属性的引入, 探讨了不同尺度下反属性的问题. 比如, 在该例子中, 挖掘出了生命形式中还具有 "能运动但无四肢 ($\bar{h}(b\bullet c)$)" 的生物, 以及 "有四肢的非哺乳类 ($\bar{i}(b\bullet c)$)" 生物. 这些都是原有的概念格表示方法所无法进行表示的.

5　结束语

针对传统概念格对形式背景表示复杂度高的问题, 本文提出了属性树的形式背景描述方法. 该方法通过对象分析属性间的包容关系, 对于可以进行属性森林描述的形式背景, 可以将其表示从格结构改变为树结构, 在保留概念格方法对形式背景偏序描述关系的同时增加了变尺度特性并通过隐含属性的定义强化了信息挖掘与规则发现, 为形式概念分析提供了新的思路.

但在对形式背景的表示过程中, 该方法目前仅能对主树清晰的形式背景进行简洁表示. 如何对该方法进行泛化使其可以应用于更多的形式背景以及挖掘形式背景属性树表示方法的其他特性是下一步的研究方向.

参考文献

[1] Wille R. Restructuring Lattice Theory: All Approach Based on Hierarchies of Concepts[M]. Dordrecht: Reidel, 1982: 445–470.

[2] Gabriela A, Stéphane D, Silvia G. Generating a catalog of unanticipated schemas in class hierarchies using formal concept analysis[J]. Information and Software Technology, 2010, 52(11): 1167–1187.

[3] Chen R C, Bau C T, Yeh C J. Merging domain ontologies based on the WordNet system and fuzzy formal concept analysis techniques[J]. Applied Soft Computing, 2011, 11(2): 1908–1923.

[4] Liu X L, Hong W X. Using formal concept analysis to visualize relationships of syndromes in traditional Chinese medicine[J]. LNCS, 2010, 6165: 315–324.

[5] Arevalo G, Mens T. Analyzing object-oriented application frameworks using concept analysis[J]. LNCS, 2002, 2426: 53–63.

[6] Wang L D, Liu X D, Cao J N. A new algebraic structure for formal concept analysis[J]. Information Sciences, 2010, 180(24): 4865–4876.