

基于属性度的属性排序算法

白冬辉, 张涛, 魏昕宇

BAI Donghui, ZHANG Tao, WEI Xinyu

燕山大学信息科学与工程学院 河北秦皇岛 066004

School of Information and science, Yanshan University, Qinhuangdao 066004, China

BAI Donghui, ZHANG Tao, WEI Xinyu. Attributes-sorting algorithm based on attribute degree. *Computer Engineering and Applications*

Abstract: Formal concept analysis is a powerful tool for data analysis and extracting rules from formal context. The attribute topology, as a novel representation of formal context, describes the association between attributes visually. Using the attribute topology theory to compute formal concepts and concept lattices are more convenient and intuitive. By studying the existing algorithms and processes based on attribute topology, the specificity and level-limitations are analyzed in this paper. By combining the concept of degrees in graph theory, a new way, attribute degree, to measure attributes is proposed. And based on attribute degree, a novel attributes-sorting algorithm is proposed. This attribute degree-based sorting algorithm is more flexible, eliminating level-limitations of attributes sorting. Parent attributes can be easily found, and new algorithms of attribute topology can be explored.

Key words: attribute topology; attributes-sorting; attribute degree; formal concept analysis

摘要: 形式概念分析是一种从形式背景进行数据分析和规则提取的强有力工具。属性拓扑作为一种新型的形式背景表示方法,直观地描述了属性之间的关联。利用属性拓扑可以更方便直观的计算形式概念和概念格。经过对现有属性拓扑的算法与流程的研究,分析了现有属性排序算法的特异性 and 层次局限性,通过结合度的概念,提出了一种属性衡量的新方式——属性度,并提出了基于属性度的属性排序算法。这种排序算法得到的结果更加灵活,消除了属性排序的层次局限性,对父属性的查找有明显的优势,为基于属性拓扑中的属性排序方法提供了指引方向。

关键词: 属性拓扑; 属性排序; 属性度; 形式概念分析

doi:10.3778/j.issn.1002-8331.1506-0065 文献标志码: A 中图分类号: TP18

1 引言

形式概念分析与粗糙集都是从数据集中挖掘隐含知识的数学工具,这两种理论分别由 Wille 和 Pawlak 提出^[1,2],它们作为有力的知识发现工具,已经成功应用于本体分析^[3]、软件工程^[4]、数据挖掘^[5,6]、知识发现^[7,8]等领域,并推动着学科的交叉发展。

随着知识的不断发展,如何从浩瀚的数据中找出有用的知识也变得更加困难。作为形式概念分析领域的有效工具,概念格的每个节点是一个形式概

念,表明了概念间的泛化与例化关系,已经成为从数据中挖掘有效信息的基础方法之一。形式概念是建立在形式背景的基础之上,而属性拓扑作为一种新型的形式背景表示方法,直观地描述了属性之间的关联和关联强度^[9]。近年来,张涛等人提出了基于属性拓扑计算概念和概念格的一系列算法^[10-12],并受到了越来越多的关注。

在应用属性拓扑理论的算法中,通常将属性拓扑分解成多个子属性拓扑,或将算法分解成多个子过程,这些子属性拓扑和子过程可以看成是由属性拓扑中某个顶点引出的,分别对每个子属性拓扑或

基金项目: 国家自然科学基金(61273019),河北省自然科学基金(F2015203013),教育部人文社会科学研究项目(14YJC740038)。

作者简介: 白冬辉(1990—),男,硕士研究生,研究领域为形式概念分析,认知计算;张涛(1979—),男,博士,副教授,研究领域为形式概念分析,认知计算,智能信息处理。E-mail: zhtao@ysu.edu.cn

子过程进行计算,然后经过整体汇总等其他算法达到相应的目标,这种思想与分布式和并行计算的思想^[13,14]相符合,也是目前处理大量数据的一种基本思想。

一个好的属性选取与排序规则,可以有效减小子过程的运算量,无论是对于传统的串行运算或是并行运算,都将使得整体运算量降低。因此,在基于属性拓扑应用的算法中,对属性的排序是普遍存在的。然而目前对属性没有统一的衡量方法,文献[10][11]中仅对顶层属性按照出度降序顺序进行排列。文献[12]中首先将顶层属性排列在伴生属性之前,分别对顶层属性和伴生属性按照出度降序顺序排列。这些属性排序算法具有层次局限性和特异性,不具备普遍适用性。良好的属性排序方法也将构造出新型的子算法,可以以更高的效率完成目标任务的计算。

本文第二章首先介绍了形式概念和属性拓扑的基本概念理论。第三章结合图论中度的相关概念,提出了一种属性的衡量标准——属性度。然后提出了一种基于属性度的属性排序算法,这种属性排序方法得到的属性排序结果不受限于属性层次的关系,更加灵活。

2 基本概念

2.1 形式概念的基本理论

形式概念分析的研究对象是形式背景,形式背景是形式概念分析的数据来源。以下介绍了形式概念相关的几个相关概念和理论^[1]。

设 G 是对象的集合, M 是属性的集合, I 是两个集合 G 和 M 之间的关系,则称三元组 $K=(G, M, I)$ 是一个形式背景,简称背景。若 $g \in G, m \in M$, 满足 $(g, m) \in I$, 则称对象 g 具有属性 m , 在背景中的 g 行 m 列的交叉处用 \times 表示。

若 $A \subseteq G, B \subseteq M$, 如果 A, B 满足条件

$$f(A) = B, g(B) = A$$

其中

$$f(A) = \{m \in M \mid \forall g \in A, (g, m) \in I\}$$

$$g(B) = \{g \in G \mid \forall m \in B, (g, m) \in I\}$$

则称 (A, B) 为一个形式概念,简称概念。对象集 A 称为概念 (A, B) 的外延,属性集 B 称为概念 (A, B) 的内涵。

如果任意满足 $f(g_i) = f(g_j)$ 的两个对象 $g_i, g_j \in G$, 都有 $g_i = g_j$, 而且对于任意满足 $g(m_i) = g(m_j)$ 的两个对象 $m_i, m_j \in M$, 都有 $m_i = m_j$, 则称形式背景 $K=(G, M, I)$ 是净化的。

形式背景的预处理的目的是去除掉形式背景中包含的冗余信息,即对全局对象、全局属性、空对象、空属性、等价对象和等价属性做必要的约简^[11], 所得形式背景为一个净化背景。如果没有特殊说明,下文所述形式背景均为预处理之后的净化背景。

2.2 属性拓扑的基本理论

属性拓扑作为形式背景的一种表示方法,从图论的角度看,是各个属性间关系的加权表示。本章简要的介绍了形式概念和属性拓扑的基本理论^[9,11], 为属性的度和属性排序算法的奠定了理论基础。

设 $T=(V, E)$ 为形式背景 $K=(G, M, I)$ 的属性拓扑表示,其中 $V=M$ 为属性拓扑的顶点集合, E 为各个顶点之间的关联,顶点之间的关联可以由邻接矩阵方式表示:

$$E(m_i, m_j) = \begin{cases} g(m_i) & i = j \\ \emptyset & g(m_i) \cap g(m_j) = \emptyset (i \neq j) \\ \emptyset & g(m_i) \cap g(m_j) = g(m_i) (i \neq j) \\ g(m_j) & g(m_i) \cap g(m_j) = g(m_j) (i \neq j) \\ g(m_i) \cap g(m_j) & \text{others} \end{cases}$$

顶点之间的关联矩阵可以用如下矩阵表示:

$$E'(m_i, m_j) = \begin{cases} -1 & g(m_i) - g(m_j) = \emptyset \\ 0 & g(m_i) - g(m_j) = g(m_i) \\ 1 & \text{others} \end{cases}$$

属性拓扑中的所有属性可以分为顶层属性和伴

生属性。其中, 如果 $\forall E(m_j, m_i) \neq \emptyset$, 都有 $E(m_i, m_j) \neq \emptyset$, 则属性 m_i 称为顶层属性。如果 $\forall E'(m_i, m_j) = -1$, 则属性 m_j 为属性 m_i 的伴生属性。

3 基于属性度的属性排序算法

3.1 属性的度

度是图论中的一个基本概念^[15], 对于有向图 $G = (V, E)$ 而言, 假设存在一个顶点 $v \in V$, 指向这个顶点 v 的边数称为顶点 v 的入度, 记做 $ID(v)$; 从这个顶点 v 指出的边数称为顶点 v 的出度, 记做 $OD(v)$ 。

在属性拓扑 $T = (V, E)$ 中, $\forall m_i \in V$, 属性 m_i 的入度、出度可以表示为:

$$D_{in}(m_i) = \#\{m \mid g(m) \cap g(m_i) \neq \emptyset, \\ g(m) \cap g(m_i) \neq g(m), \forall m \in V\}$$

$$D_{out}(m_i) = \#\{m \mid g(m) \cap g(m_i) \neq \emptyset, \\ g(m) \cap g(m_i) \neq g(m_i), \forall m \in V\}$$

其中, $\#\{\cdot\}$ 表示集合中元素的个数。

定义 1 $\forall m_i \in V$, 属性 m_i 的度比定义为属性 m_i 的出度与入度的比值, 记为

$$D_{ratio}(m_i) = \frac{D_{out}(m_i)}{D_{in}(m_i)}$$

定理 1 如果属性 m_i 为顶层属性, 则 $D_{ratio}(m_i) \geq 1$ 。

证明 根据顶层属性的定义, 如果属性 m_i 为顶层属性

$$\forall E(m_j, m_i) \neq \emptyset, \forall E(m_i, m_j) \neq \emptyset$$

则根据邻接矩阵的定义可知

$$\forall g(m_j) \cap g(m_i) \neq \emptyset \text{ 且 } g(m_j) \cap g(m_i) \neq g(m_j)$$

都满足

$$g(m_i) \cap g(m_j) \neq \emptyset \text{ 且 } g(m_i) \cap g(m_j) \neq g(m_i)$$

即

$$\{m \mid g(m) \cap g(m_i) \neq \emptyset, g(m) \cap g(m_i) \neq g(m_i), \\ \forall m \in V\} \supseteq \{m \mid g(m) \cap g(m_i) \neq \emptyset, \\ g(m) \cap g(m_i) \neq g(m), \forall m \in V\}$$

$$D_{out}(m_i) \geq D_{in}(m_i)$$

$$D_{ratio}(m_i) = \frac{D_{out}(m_i)}{D_{in}(m_i)} \geq 1$$

证毕。

定义 2 已知一个形式背景 $K = (G, M, I)$, 构造有向图 $T_G = (V_G, E_G)$, 其中

$$V_G \subseteq V$$

对于 $\forall m_i, m_j \in V_G$

$$E_G(m_i, m_j) = \begin{cases} 1 & g(m_i) \supseteq g(m_j) \\ 0 & \text{others} \end{cases}$$

则属性 m_i 在属性集 V_G 中的动度表示为

$$D_{dy}(m_i, V_G) = ID(m_i)$$

定理 2 如果属性 m_i 为顶层属性, 则 $D_{dy}(m_i, V_G) = 0$ 。

图 1 为一属性拓扑, 将各个属性的入度、出度、度比和动度列于表 1 中, 其中计算动度时取 $V_G = V$ 。

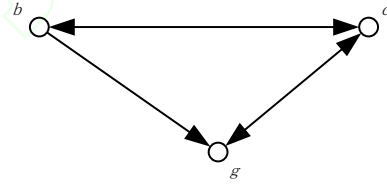


图 1 属性拓扑示例

表 1 图 1 所示属性拓扑中属性的度

m	$D_{in}(m)$	$D_{out}(m)$	$D_{ratio}(m)$	$D_{dy}(m)$
b	1(c)	2(c, g)	2	0
c	2(b, g)	2(b, g)	1	0
g	2(b, c)	1(c)	1/2	1

在图 1 所示的属性拓扑中, 属性 b 和属性 c 是属性, 它们的度比值都不小于 1, 动度均为 0。

在一个属性拓扑中, 属性的动度并不是固定的, 而是随着选取的属性集的改变而动态变化的。通过合理的选取属性集合 V_G , 伴生属性的动度可以为 0。

属性的动度打破了顶层属性和伴生属性这种生硬的划分层次, 动态的描述了属性的层次关系。在每一层内, 可以使用属性的出度、入度、度比等概念衡量同层属性, 本文提出的属性排序算法中选取属性的度比作为层内的属性衡量度。

3.2 属性排序算法描述

本节以属性拓扑为基础, 以属性度的相关概念

为衡量方式, 提出一种新型的属性排序方法。

已知一个形式背景 $K = (G, M, I)$, $T = (V, E)$ 为这个形式背景对应的属性拓扑表示, 需要排序的属性集 $V_G \subseteq V$, 基于属性度的属性排序算法描述如下:

输入: 需要排序的属性集合 V_G

输出: 有序属性组 N

步骤 1 令有序属性组 $N = \emptyset$ 。

步骤 2 选取层内衡量度: $\forall m \in V$, 计算 $D_{ratio}(m)$ 。

步骤 3 计算属性的动度: $\forall m \in V_G$, 计算 $D_{dy}(m)$ 。

步骤 4 属性分层: 令 $V_G = V_0 + V_+$, 其中

$$V_0 = \{m \mid D_{dy}(m) = 0, \forall m \in V_G\}$$

$$V_+ = \{m \mid D_{dy}(m) > 0, \forall m \in V_G\}$$

步骤 5 选取一个属性 m_i , 满足 $m_i \in V_0$, 且 $\forall m \in V_0 - m_i$, 有 $D_{ratio}(m_i) > D_{ratio}(m)$

步骤 6 清理有向图的边: $\forall m_j \in V_G \subseteq V$, 令

$$\begin{cases} E_G(m_i, m_j) = \emptyset \\ E_G(m_j, m_i) = \emptyset \end{cases}$$

步骤 7 清理有向图的顶点, 令

$$V_G = V_G - \{m_i\}$$

步骤 8 如果 $V_G \neq \emptyset$, 跳转到步骤 3。否则算法结束, 得到的有序属性组为 N 。

假设 V 中包含 n 个属性, V_G 中包含 n_G 个属性, 根据属性拓扑的存储方式, 计算度比的复杂度为 $O(n^2)$, 计算动度为 0 的属性集合 V_0 所需的时间复杂度为 $O(n_G^2)$, 在集合中选取最大属性的复杂度为 $O(n_G)$, 因此算法的时间复杂度为 $O(n^2) + n_G(O(n_G^2) + O(n_G))$, 若对所有属性进行排序, 则时间复杂度为 $O(n^3)$ 。

该算法在排序过程中, 每次选定一个属性后, 会重新对属性集合中剩余的属性分层, 这打破了现有排序算法的层次局限性, 当 $\#V_0 > 1$ 时, 将属性度作为属性优先选取的标准, 增强了排序算法的普适性。

以上描述的算法流程图如图 2 所示。

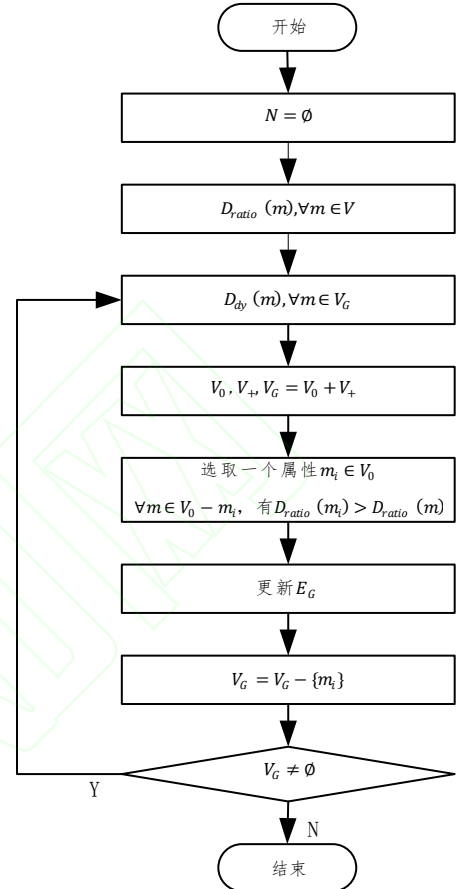


图 2 基于属性度的属性排序算法流程图

性质 1 $\forall m_j \in M$, $\exists m_i \in M$, 满足 $g(m_j) \supsetneq g(m_i)$, 则在有序属性组 N 中, m_i 一定排在 m_j 之前。

证明 假设 m_j 排在 m_i 之前, 即 $m_i, m_j \in V_G$

由于 $g(m_j) \subseteq g(m_i)$

$$g(m_j) \cap g(m_i) = g(m_j)$$

即:

$$\begin{aligned} m_i &\in \{m \mid g(m) \cap g(m_j) \neq \emptyset, \\ &g(m) \cap g(m_j) \neq g(m), \forall m \in V_G\} \end{aligned}$$

因此

$$\begin{aligned} D_{dy}(m_j) &= \#\{m \mid g(m) \cap g(m_j) \neq \emptyset, \\ &g(m) \cap g(m_j) \neq g(m), \forall m \in V_G\} \\ &= \#\{m_i\} + \#\{m \mid g(m) \cap g(m_j) \neq \emptyset, \\ &g(m) \cap g(m_j) \neq g(m), \forall m \in V_G - m_i\} \end{aligned}$$

$$= 1 + \#\{m \mid g(m) \cap g(m_j) \neq \emptyset\},$$

$$g(m) \cap g(m_j) \neq g(m), \forall m \in V_G - m_i\}$$

$$> 0$$

因此 $\forall m_j \in M$, $\exists m_i \in M$, 满足 $g(m_j) \subseteq g(m_i)$, 根据算法中的选取规则, 不满足选取 m_j 的条件。即 m_i 一定排在 m_j 之前。

证毕。

性质 2 若 $m_i, m_j \in V_G$, 且 m_i 为顶层属性, m_j 为伴生属性, 则伴生属性 m_j 可以出现在顶层属性 m_i 之前。

证明 m_j 为伴生属性, 记集合

$$P = \{m \mid g(m) \supseteq g(m_j), \forall m \in V_G - m_j\}$$

若顶层属性

$$m_i \notin P$$

且

$$V_G \cap P = \emptyset$$

则

$$D_{dy}(m_j) = \#\{m \mid g(m) \cap g(m_j) \neq \emptyset,$$

$$g(m) \cap g(m_j) \neq g(m), \forall m \in V_G\} = 0$$

根据算法中的选取规则, 满足选取 m_j 的条件。

即伴生属性 m_j 可以出现在顶层属性 m_i 之前。

证毕。

3.3 实验

生物与水的形式背景是五个经典形式背景之一^[16], 本文中以生物与水的形式背景为例, 对基于属性度的属性排序算法描述进行实验, 首先将其形式背景列于表 2 中。

表 2 生物与水的形式背景

	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>	<i>h</i>	<i>i</i>
1	×					×		
2	×					×	×	
3	×	×				×	×	
4		×				×	×	×
5	×		×		×			
6	×	×	×		×			
7		×	×	×				
8		×	×		×			

按照属性拓扑的定义可以画出生物和水的属性拓扑图 $T=(V, E)$, 如图 3 所示。为了便于描述, 绘于图 3 中的属性拓扑没有标明边上的权值。

计算属性拓扑中所有属性 $V=\{b, c, d, e, f, g, h, i\}$ 的度比 $D_{ratio}(m)$, 作为同层

属性衡量的标准, 列于表 3 中。

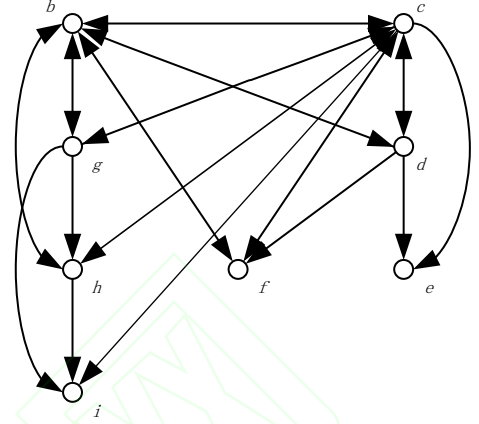


图 3 生物与水的属性拓扑 $T=(V, E)$

表 3 属性集 V 中各个属性的度比

<i>m</i>	$D_{ratio}(m)$	<i>m</i>	$D_{ratio}(m)$
<i>b</i>	1	<i>e</i>	0
<i>c</i>	7/5	<i>f</i>	2/3
<i>d</i>	2	<i>h</i>	1
<i>g</i>	2	<i>i</i>	0

当选取排序属性集为 $V_G=V$ 时, 有向图 $T_G=(V_G, E_G)$ 如图 4 所示。计算 T_G 中各个属性的动度列于表 4 中:

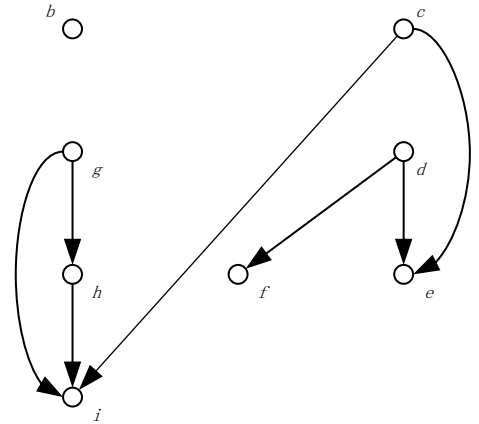


图 4 属性集 $V_G=\{b, c, d, e, f, g, h, i\}$ 时有向图 T_G

表 4 属性集 $V_G=\{b, c, d, e, f, g, h, i\}$ 时各个属性的动度

<i>m</i>	$D_{dy}(m)$	<i>m</i>	$D_{dy}(m)$
<i>b</i>	0	<i>e</i>	2
<i>c</i>	0	<i>f</i>	1
<i>d</i>	0	<i>h</i>	1
<i>g</i>	0	<i>i</i>	3

此时属性 b 、 c 、 d 、 g 的动度都为 0, 即 $V_0=\{b, c, d, g\}$ 。 $\forall m \in V_0$, 经过比较表 3 中计算的度比值, 选取度比最大的属性即属性 g , 加入有序属性组 N 中。此时有序属性组 $N=\{g\}$ 。

从顶点集 V_G 中删除顶点属性 g , 同时更新 E_G 中与属性 g 的关联。生成的有向图 $T_G=(V_G, E_G)$ 如

图 5 所示,将此时有向图 T_G 中各个属性的动度列于表 5 中。

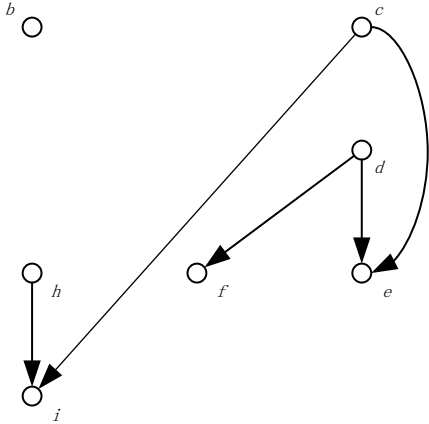


图 5 属性集 $V_G = \{b, c, d, e, f, h, i\}$ 时有向图 T_G

表 5 属性集 $V_G = \{b, c, d, e, f, h, i\}$ 时各个属性的动度

m	$D_{dy}(m)$	m	$D_{dy}(m)$
b	0	e	2
c	0	f	1
d	0	h	0
		i	2

此时属性 b 、 c 、 d 、 h 的动度都为 0, 即 $V_0 = \{b, c, d, h\}$ 。 $\forall m \in V_0$, 经过比较表 3 中计算的度比值, 度比最大的属性为属性 d , 根据算法中属性的选取规则, 此处将属性 d 加入到有序属性组 N 中。此时有序属性组 $N = \{g, d\}$ 。再次更新有向图 T_G 中的顶点集以及与顶点 d 之间的关联。

依次计算步骤 3 至步骤 8, 最后得到有序属性组为 $N = \{g, d, c, h, b, f, e, i\}$ 。

然而, 按照此算法构造的有序属性组并不是唯一的, 产生有序属性组多样性的原因是, 对于算法中满足步骤 5 的属性, 同时满足 $\#\{m_i \mid D_{ratio}(m_i) \geq D_{ratio}(m), \forall m \in V, m_i \in V_0\} > 1$ 时, 算法中没有指定集合中选取属性的优先顺序。例如在选取第一个属性时, 属性 $g, d \in V_0$, 且满足 $D_{ratio}(g) = D_{ratio}(d) = 2$, 实验中选择的是属性 g , 得到有序属性组为 N , 如果选取的属性为 d , 则最终得到的排序结果为 $N_0 = \{d, g, c, h, b, f, e, i\}$ 。

但无论是哪组有序属性组, 由于 $g(i) \subseteq g(h) \subseteq g(g)$, 属性 g 必然出现在属性 h 与属性 i 之前。同时从有序属性组 N 中可以看出, 虽然属性 b 是顶层属性、属性 h 是伴生属性, 但伴生属性 h 位于顶层属性 b 之前。

通过算法描述和实验举例, 相比于文献[10-12]中采用的属性排序算法, 虽然本文算法的复杂度略高于其它算法, 但这种基于属性度的动态排序算法, 更加准确的描述了当前时刻拓扑的属性耦合程度, 即这种排序更加符合最初排序的目的, 同时这种算法与其它算法的不同之处在于这种算法不仅用于属性排序, 而具有更加实际的意义, 如根据性质 1, 可以加快父属性的搜索。

4 结束语

本文提出的基于属性度的属性排序算法, 以属性动度为分层标准, 以属性的其他度在同一层内辅助排序, 每确定一个属性后, 重新对剩余的属性分层, 以此得到最终的排序结果。该算法突破了传统的顶层属性与伴生属性的层次划分, 并消除了现有属性排序算法的特异性, 增强了属性排序的普适性。

参考文献

- [1] Ganter B, Wille R. Formal concept analysis: mathematical foundations[M]. Springer Science & Business Media, 2012.
- [2] Pawlak Z. Rough sets[J]. International Journal of Computer & Information Sciences, 1982, 11(5): 341-356.
- [3] Cigarrán-Recuero J, Gayoso-Cabada J, Rodríguez-Artacho M, et al. Assessing semantic annotation activities with formal concept analysis[J]. Expert Systems with Applications, 2014, 41(11): 5495-5508.
- [4] Kumar G N, Kumar C. Generation of high level views in reverse engineering using formal concept analysis[C]//Networks & Soft Computing (ICNSC), 2014 First International Conference on. IEEE, 2014: 334-338.
- [5] Yan Hehua, Zou Caifeng, Liu Jianqi, et al. Formal concept analysis and concept lattice: perspectives and challenges[J]. International Journal of Autonomous and Adaptive Communications Systems, 2015, 8(1): 81-96.
- [6] Jamalpur B, Prakash R V, Sarma S. Mining Association Rules Using Formal Concept Analysis[J]. International Journal of Innovative Research and Development, 2014, 3(12).
- [7] Jing Jing, Bian Hong, Pan Lei, et al. Visualized study on formulation regularities of decoctions in traditional Chinese medicine[J]. Journal of Chemical and Pharmaceutical Research, 2014, 6(7): 2169-2174.
- [8] Sumangali K, Kumar C. Determination of interesting rules in FCA using information gain[C]//Networks & Soft Computing (ICNSC), 2014 First International Conference on. IEEE, 2014: 304-308.
- [9] Zhang Tao, Ren Honglei. Attribute Topology of Formal Context [J]. Journal of Chinese Computer Systems, 2014, 3: 033.
- [10] 张涛, 任宏雷. 形式背景的属性拓扑表示[J]. 小型微型计算机系统, 2014, 3: 033.
- [11] Zhang Tao, Ren Honglei, Wang Xiaomin. A calculation of

- formal concept by attribute topology[J]. ICIC express letters. Part B, Applications: an international journal of research and surveys, 2013, 4(3): 793-800.
- [12] Zhang Tao, Ren Honglei, Hong Wenxue, et al. The Visualizing Calculation of Formal Concept that Based On the Attribute Topologies [J], Acta Electronica Sinica, 2014, 42(5): 925-932.
- [13] 张涛, 任宏雷, 洪文学, 等. 基于属性拓扑的可视化形式概念计算[J]. 电子学报, 2014, 42(5): 925-932.
- [14] Zhang Tao, Li Hui, Hong Wenxue, et al. Deep First Formal Concept Search[J]. The Scientific World Journal, 2014, 2014.
- [15] Adve S, Adve V S, Agha G, et al. Parallel computing research at illinois: The upcrc agenda[J]. Urbana, IL: Univ. Illinois Urbana-Champaign, 2008.
- [16] Moore G E. Cramming more components onto integrated circuits[J]. Proceedings of the IEEE, 1998, 86(1): 82-85.
- [17] Diestel R. Graph theory (Graduate texts in mathematics)[J]. Springer-Verlag, 2005.
- [18] Wille, R. Liniendiagramme hierarchischer Begriffssysteme. In: Bock, HH (ed.), Studien zur Klassifikation, Bd.15, Indeks-Verlag, Frankfurt, 1984, 32-51