

文章编号:1001-9081(2010)10-2598-04

## 形式概念演化生成算法

杜 鹃<sup>1</sup>, 丁爱萍<sup>1</sup>, 汪传建<sup>2</sup>, 张 卓<sup>2</sup>

(1. 黄河水利职业技术学院 信息工程系, 河南 开封 475003; 2. 武汉大学 计算机学院, 武汉 430079)

(dujuanzzu@qq.com; charles.zz@gmail.com)

**摘 要:** 目前仍然缺乏使用遗传算法构造概念的研究。为此, 首先把形式概念的构造问题转换为以形式背景的对象幂集和属性幂集组合空间为搜索空间, 以伽罗瓦联系为约束条件的约束最优化问题; 然后提出一个新颖的基于遗传演化的概念生成算法——遗传概念生成算法(Geacob)。该算法采用变长结构编码, 不仅满足概念形式的表示和演化过程的需要, 而且使该算法具有更好的扩展性和通用性。实验表明了该遗传算法求解形式概念的可行性和有效性。

**关键词:** 遗传算法; 结构编码; 形式概念分析; 形式概念构造

**中图分类号:** TP311.13; TP18 **文献标志码:** A

## Genetic algorithm to generate formal concept

DU Juan<sup>1</sup>, DING Ai-ping<sup>1</sup>, WANG Chuan-jian<sup>2</sup>, ZHANG Zhuo<sup>2</sup>

(1. Information Technology Engineering, Yellow River Conservancy Technical Institute, Kaifeng Henan 475003, China;

2. School of Computer, Wuhan University, Wuhan Hubei 430079, China)

**Abstract:** At present, there is few research literature about genetically constructing formal concept. After considering formal concept construction as an optimization with constraints of Galois connection, a new concept generating algorithm named Geacob based on genetic evolution was proposed, and its research space consisted of power sets of objects and attributes in formal context. The proposed algorithm adopting variable structure can not only reasonably formalize the concept, but also satisfy the requirements in the procedure of concept's evolution, and has consequential properties of scalability and versatility. The experimental results show that the algorithm is feasible and effective to generate formal concept.

**Key words:** Genetic Algorithm (GA); structural coding; Formal Concept Analysis (FCA); formal concept construction

## 0 引言

遗传算法(Genetic Algorithm, GA)是一种借鉴生物界自然选择和自然遗传机制, 模拟自然进化过程搜索最优解的方法<sup>[1-2]</sup>。由于其解决问题以混沌、随机和非线性为典型特征, 因而近十几年来已在组合优化、机器学习、数据挖掘<sup>[3-5]</sup>等领域得到相当广泛的应用。基于遗传算法的方法是运用遗传算法的自适应寻优及智能搜索技术, 获取与客观事实最相容的问题解。本文把遗传算法应用于形式概念分析理论中的概念生成过程。

形式概念理论<sup>[6]</sup>是20世纪80年代初由德国的R. Wille教授提出。概念格作为形式概念分析(Formal Concept Analysis, FCA)的核心数据结构体现了哲学上概念外延与内涵之间对偶关系, 已经成为有力的数据分析工具, 在诸多领域有着广泛的应用。在知识发现领域, 概念格可以从关系数据中构造出来, 然后在概念格上提取各种类型的知识, 如蕴涵规则、关联规则、分类规则等。在挖掘规则知识的过程中, 规则本身是用内涵集之间的关系来描述的, 而体现于相应外延集之间的包含(或近似包含)关系。由于概念格节点反映了概念内涵和外延的统一, 节点间关系体现了概念之间的泛化和特化关系, 因此非常适合作为规则发现的基础性数据结构<sup>[7-8]</sup>。

从早期的以Bordat与Ganter<sup>[9]</sup>为代表的批处理方式的格构造算法和以Godin<sup>[10]</sup>为代表的渐进式的概念格构造算法, 到21世纪初以Valtchev<sup>[11-12]</sup>为代表的集成概念格构造算法, 概念格的构造一度成为研究的主流。国内也有一些研究成果, 如: 多概念格的横向合并算法<sup>[13]</sup>、基于搜索空间划分的概念生成算法<sup>[14]</sup>。概念构造与概念格构造的主要区别在于是否生成概念间的偏序关系。

然而据作者所知, 目前仍然缺少利用遗传算法构造概念的文献。为此, 本文提出一个新颖的基于遗传算法的形式概念构造算法——遗传概念生成算法 Geacob (Genetic Algorithm for Concept Build)。该算法采用结构编码, 每个个体表示为一个候选形式概念, 包括对象集合和外延集合两个部分, 因而编码具有变长<sup>[3]</sup>的特点, 使得该算法具有较强的扩展性和通用性, 不受形式背景大小的制约。

## 1 相关概念

本章简要介绍所需要的基本概念, 关于概念格的详尽形式化描述可参考文献<sup>[6]</sup>。

**定义1** 形式背景。给定的形式背景为一个三元组  $K = (O, A, I)$ , 其中  $O$  是事务集合,  $A$  是属性集合,  $I$  是  $O$  和  $A$  之间的二元关系, 即  $I \subseteq O \times A$ 。

这里的所有形式背景属性值皆为二元值, 如果数据库中

收稿日期: 2010-04-14; 修回日期: 2010-06-30。

**作者简介:** 杜鹃(1982-), 女, 河南开封人, 助教, 硕士, 主要研究方向: 演化计算; 丁爱萍(1966-), 女, 河南开封人, 副教授, 硕士, 主要研究方向: 智能信息处理; 汪传建(1977-), 男, 安徽怀宁人, 讲师, 博士研究生, CCF 会员, 主要研究方向: 数据库安全、数据挖掘、演化计算; 张卓(1978-), 男, 河南郑州人, 博士研究生, 主要研究方向: 形式概念分析、Web 数据挖掘、数据融合。

的属性值数量大于2就需要进行概念扩展<sup>[6]</sup>。

**定义2** 伽罗瓦(Galois)联系。对于形式背景  $K = (O, A, I)$ , 对象集合  $O$  的幂集  $P(O)$  和属性集合  $A$  的幂集  $P(A)$  之间可以定义两个对偶映射关系  $f$  和  $g$ , 如下:

$$f: P(O) \rightarrow P(A), f(X) = \{a \in A \mid \forall o \in X, oIa\}$$

$$g: P(A) \rightarrow P(O), g(Y) = \{o \in O \mid \forall a \in Y, oIa\}$$

则该映射对偶  $(f, g)$  称为伽罗瓦(Galois)联系。

**定义3** 形式概念。形式背景  $K = (O, A, I)$  上的一个形式概念是一个二元组  $(X, Y)$ , 其中  $X \in P(O), Y \in P(A), Y = f(X), X = g(Y)$ , 称  $X$  是概念  $(X, Y)$  外延,  $Y$  是概念  $(X, Y)$  内涵。用  $C_K$  表示形式背景  $K = (O, A, I)$  上所有概念的集合。

**定义4** 概念格(又称为 Galois lattice)。对于形式背景  $K$  所产生的所有概念集合  $C_K$ , 以及  $C_K$  上的偏序关系所导出的有序集  $L = (C_K, \leq)$ , 称之为形式背景  $K$  的概念格。概念格中的每个节点都是一个形式概念。

举例说明:如表1是一个形式背景,其中“x”表示对象具有该属性,图1是其对应的所有概念组成的概念格。概念  $c = (256, bcd)$  表示对象2,5,6拥有共同属性b,c,d;属性b,c,d被对象2,5,6所共同持有。

表1 形式背景

对象	属性				
	a	b	c	d	e
1	x	x			
2		x	x	x	
3				x	x
4				x	x
5		x	x	x	
6		x	x	x	
7	x	x			

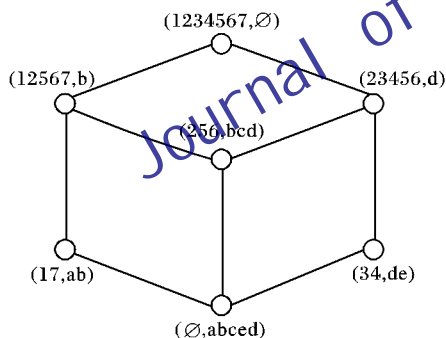


图1 由表1形式背景所生成的概念格

## 2 遗传概念生成算法 Geacob

遗传概念生成算法的主要思想是:因为形式概念  $c = (X, Y)$  的外延  $X$  和内涵  $Y$  分别是其对应形式背景  $K = (O, A, I)$  的对象集  $O$  的子集和属性集  $A$  的子集, 也就是说  $X \in P(O), Y \in P(A)$ ; 并且它们之间满足 Galois 联系。那么就可以认为求解特定形式背景  $K = (O, A, I)$  的所有概念集合  $C_K$  的问题, 等价于以对象集的幂集与属性集的幂集的组合, 即  $P(O) \times P(A)$  为搜索空间, 以 Galois 联系为约束条件的多目标最优解问题。其中搜索空间大小为  $SpaceSize = 2^{|O|} \times 2^{|A|}$ , 并且目标解(形式概念)为满足 Galois 联系的对象集与属性集组合。对于这样的一个具有约束优化问题, 可以使用简单遗传算法来实现。下面就对该遗传算法进行详细说明。

### 2.1 个体编码

在形式概念分析中,形式背景的属性值都是二元值,并且

形式背景的对象和属性都可以用自然数表示,而形式概念的表示分为外延和内涵,所以 Geacob 算法采用结构编码表示单一概念个体,定义如下。

**定义5** 概念个体。由对象集合和属性集合两部分组成,即  $\{(x, y) \mid x \in P(O), y \in P(A)\}$ , 其中对象集合部分称为概念个体的外延,属性集合部分称为概念个体的内涵。概念的外延  $X$  和内涵  $Y$  使用自然数集合编码。

根据定义5,概念个体编码是变长的结构编码,它表示候选形式概念,当其对象集合和属性集合之间满足 Galois 联系,即为形式概念。

### 2.2 适应函数

一个个体的优劣是看其是否满足 Galois 联系的强弱来评价的,因为个体概念  $c = (X, Y)$  有两个部分(定义5),那么就需要对其分别进行评价,然后综合它们的评价结果得到整个概念个体(候选形式概念)的评价值。下面给出它们的适应函数定义。

**定义6** 外延适应函数。  $E_1: C_K \rightarrow \mathbf{R}^+, E_1(c) =$

$$\frac{|X \cap g(Y)|}{|X \cup g(Y)|}$$

**定义7** 内涵适应函数。  $E_2: C_K \rightarrow \mathbf{R}^+, E_2(c) =$

$$\frac{|f(X) \cap Y|}{|f(X) \cup Y|}$$

**定义8** 概念个体适应函数。  $E: C_K \rightarrow \mathbf{R}^+, E(c) = E_1^{\alpha}(c) + E_2^{\beta}(c)$ , 其中  $\alpha$  为正整型常数。

**引理1** 对于一个候选形式概念(概念个体)  $c = (X, Y)$  如果其适应度值(定义8)满足  $E(c) = 2$ , 则该候选形式概念满足 Galois 联系,即  $c$  为一个形式概念。

**证明** 因为  $E(c)$  由两部分组成,先考虑  $E_1(c)$ , 因为  $(X \cap g(Y)) \subseteq (X \cup g(Y))$ , 所以  $E_1(c) \in [0, 1]$ ; 并且只有  $(X \cap g(Y)) = (X \cup g(Y))$  时,  $E_1(c) = 1 \Rightarrow X = g(Y)$ 。同理可得  $E_2(c) \in [0, 1], E_2(c) = 1 \Rightarrow f(X) = Y$ 。综上所述,  $E(c) \in [0, 2]$ , 当  $E(c) = 2$  时, 概念个体的对象集合和属性集合满足 Galois 联系(定义2), 得证。

遗传概念生成算法在每代种群被评价后,把  $E(c) = 2$  并且不重复的概念个体保存到结果中。

### 2.3 选择策略

该算法的父体选择采用锦标赛策略。从种群中随机地选择  $k$  个个体进行比较,适应值最大的个体将被选择作为生成下一代的父体,这个过程反复进行  $N$  次。参数  $k$  称为竞争规模。

### 2.4 交叉算子

以上述选择策略选择出来的父体以平均分布概率参与交叉运算。因为概念个体有两个部分,外延和内涵分别具有各自独立的交叉概率(外延交叉概率  $p_{ex}$  和内涵交叉概率  $p_{ey}$ ), 它们可以是定值也可以变量。考虑到外延和内涵的适应值趋近于1, 概念个体与目标解的距离就越小, 因而外延或内涵应该越稳定, 以达到更快得到目标解的目的。因此本文算法中  $p_{ex}$  和  $p_{ey}$  分别根据外延和内涵适应值动态计算获得, 定义如下。

**定义9** 外延交叉概率  $p_{ex} \circ p_{ex} = a - \text{Max}(E_1(c_1), E_1(c_2))$ , 其中  $c_1, c_2$  为参与交叉运算的两个父体;  $a$  是常量参数,  $a \in [1, 2]$ 。

**定义10** 内涵交叉概率  $p_{ey} \circ p_{ey} = b - \text{Max}(E_2(c_1),$

$E_2(c_2)$ ), 其中  $c_1, c_2$  为参与交叉运算的两个父体;  $b$  是常量参数,  $b \in [1, 2]$ 。

另外, 因为概念个体采用变长编码, 个体外延和内涵的大小不定, 导致参与交叉的两个父体的外延和内涵两部分编码长度各不相同, 因而两个部分的插入点  $I_x, I_y$  也是分别随机产生的。Geacob 算法遵循平均分布产生插入点  $I_x$  和  $I_y$ , 取值范围分别为 0 到父体外延和内涵部分的编码长度的最大值, 即:

$$I_x \in [0, \max(|X_1|, |X_2|)]$$

$$I_y \in [0, \max(|Y_1|, |Y_2|)]$$

这样设计可以使得个体遗传编码在演化过程中实现个体大小的随机演化, 符合目标解(形式概念)的结构特征, 达到求解多个目标概念的目的。交叉算子交叉过程示例如图2所示, 交叉过程(CrossOver 函数)描述如下。

- 1) 以平均概率随机从父体种群  $G_t'$  中选择两个父体  $c_1, c_2$ ;
- 2) 分别根据定义9、10 计算得到  $p_{cx}$  和  $p_{cy}$ ;
- 3) 以平均概率产生两个取值范围为  $[0, 1]$  的随机数  $p_{cross}^1, p_{cross}^2$ ;
- 4) 如果  $p_{cross}^1 < p_{cx}$ , 随机产生一个插入点  $I_x$ , 两个父体外延部分完成交叉运算;
- 5) 如果  $p_{cross}^2 < p_{cy}$ , 随机产生一个插入点  $I_y$ , 两个父体内涵部分完成交叉运算;
- 6) 将交叉运算后产生的新的个体放入到下一代种群  $G_t''$  中;
- 7) 重复步骤1) ~ 6), 直到  $\|G_t''\| = N$  时, 结束交叉运算。

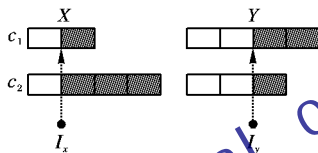


图2 交叉算子交叉过程示例

## 2.5 变异算子

概念个体外延和内涵分别以相互独立的变异概率  $p_{mx}$ 、 $p_{my}$  执行变异操作。因为外延和内涵都是集合, 因此变异运算可以转换为集合的差与并操作, 经过变异过程, 保证了概念个体编码长度随机演化变化。个体外延(内涵)部分变异过程(Mutate 函数)描述如下( $p_m = p_{mx}$  或  $p_{my}$ )。

- 1) 以平均概率产生两个取值范围为  $[0, 1]$  的随机数  $p_{mutate}^1, p_{mutate}^2$ ;
- 2) 如果  $p_{mutate}^1 < p_m$ , 参与变异运算的个体外延(内涵)集合中的每个元素按照 0, 1 分布随机去除(0 去除, 1 保留);
- 3) 如果  $p_{mutate}^2 < p_m$ , 按照平均概率从全体对象(全体内涵)集合中随机选取元素与参与变异运算的个体外延(内涵)集合执行并操作。

## 2.6 Geacob 算法描述

输入: 形式背景  $FormalContext$ ; 种群的规模  $N$ ; 适应度函数  $E(c)$  的参数  $\delta$  (定义8); 最大演化代数  $G_{Max}$ , 也是整个算法的终结条件; 选择策略的竞争规模  $k$ ; 计算外延和内涵交叉概率  $p_{cx}, p_{cy}$  时所需要的参数  $a, b$ ; 外延和内涵变异概率  $p_{mx}, p_{my}$ 。

输出: 形式概念集合  $C_K$ 。

Algorithm Geacob

Input:  $FormalContext, N, \delta, G_{Max}, k, a, b, p_{mx}, p_{my}$ ;

```

output:  $C_K$ ;
 $t \leftarrow 0$ ;
 $G_t \leftarrow \text{Initialize} (FormalContext, N)$ ;
Evaluate ( $FormalContext, G_t, \delta$ );
while  $t \leq G_{Max}$ 
     $G_t' \leftarrow \text{ParentSelect} (G_t, N, k)$ ;
     $G_t'' \leftarrow \text{CrossOver} (G_t', a, b)$ ;
     $G_t''' \leftarrow \text{Mutate} (G_t'', p_{mx}, p_{my})$ ;
    Evaluate ( $FormalContext, G_t''', \delta$ );
     $C_K' \leftarrow \text{GetConcept} (G_t''')$ ;
     $C_K \leftarrow C_K \cup C_K'$ ;
     $G_t \leftarrow G_t \cup G_t'''$ ;
     $t \leftarrow t + 1$ ;
end while
return  $C_K$ ;

```

该算法中 Initialize 函数执行种群初始化任务, 随机产生规模为  $N$  的概念个体, 每个概念个体的外延与内涵部分随机从形式背景的对象集合和属性集中选取; Evaluate 过程使用适应度函数  $E(c)$  对种群  $G_t$  中的每个个体进行评估, 评估值保持在个体的数据结构中, 避免重复计算; ParentSelect 函数负责选择参与交叉运算的个体, 选择策略2.3节已经说明; CrossOver 函数负责进行交叉运算, 2.4节已经说明具体过程; Mutate 函数负责进行变异运算, 2.5节也已经说明; Getconcept 负责从变异后的种群中获得  $E(c) = 2$  并且不重复的概念个体, 并加入到结果集合  $C_K$  中。

## 3 实验结果及分析

本文实验使用 Matlab 分别实现 Geacob 算法和 Petko 算法<sup>[11]</sup>。Petko 算法是一个基于并置集成的概念格构造算法, 在本文中将其作为基准算法, 由该算法得到的所有形式概念集合  $C_K'$ 。与 Geacob 算法得到的形式概念集合  $C_K$  进行比较, 使用精确度 (precision) 和召回率 (recall) 对 Geacob 算法的结果质量进行评价, 评估函数如下:

$$precision = \frac{C_K' \cap C_K}{C_K}$$

$$recall = \frac{C_K' \cap C_K}{C_K'}$$

实验主要验证 Geacob 算法的可行性, 设置如下。

数据集1 使用本文表1中的形式背景作为测试数据集, 该形式背景一共产生7个形式概念。

数据集2 T510.01D0.03K, 测试数据由 IBM 数据生成器产生, 数据集含有30个事务, 10个属性项, 每个事务平均拥有5个属性。该数据集即为形式背景, 一共产生31个形式概念。

数据集3 Balloons Data Set (UCI 数据集<sup>[15]</sup>), 该数据集拥有20个实例, 概念扩展具有10个单值属性; 一共产生91个形式概念。

种群规模  $N$  分别为 20、40、80、100; 演化代数  $G_{Max}$  分别选择 100、200、400、500、800、1 000。竞争规模  $k = 2$ ; 另外, 令  $a = 1.2, b = 1.2$ , 这样既能保证外延或内涵随着其适应值趋近于1, 其编码部分越稳定; 又能保证外延或内涵部分适应值为1时, 也有交叉可能性, 避免算法陷入局部最优解中。外延和内涵变异概率取值分别为  $p_{mx} = 0.25, p_{my} = 0.25$ 。

在不同种群  $N$  和演化代数  $G_{Max}$  情况下, Geacob 算法由数



据集1、数据集2和数据集3所生成的形式概念的精确度均为1,故本文不再一一列出各种情况下的精确度值。这说明概念个体适应度函数(定义6~8)的定义是正确,使得最终演化生成的概念个体符合 Galois 联系,即为形式概念。但是由于遗传算法的随机性,搜索到全部的形式概念需要更大的种群和演化代数,这对算法的性能会有一些的影响。下面具体分析在不同的数据集下,种群和演化代数对召回率的影响。

在数据集1的实验中,种群规模远远大于目标概念数量(7个形式概念),Geacob 算法的召回率如图3所示:随着种群规模增加,召回率逐渐提高;随着演化代数增加,获得新的形式概念的数量就越多,召回率也随之提高,最好的情况可以获得所有的形式概念。

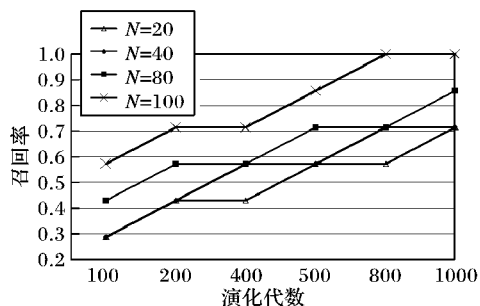


图3 种群和演化代数不同情况下 Geacob 算法的召回率(数据集1)

在数据集2的实验中,Geacob 算法的召回率如图4所示:召回率与种群规模、演化代数的大小依然成正比的关系。该实验中最好的情况是0.73,这是因为种群规模和演化代数本身就限制着最终符合 Galois 联系约束条件的概念个体的数量。

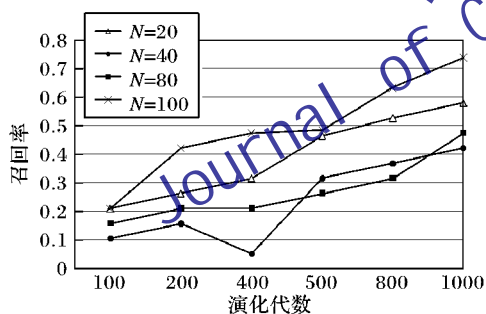


图4 种群和演化代数不同情况下 Geacob 算法的召回率(数据集2)

数据集3的实验进一步验证了结果概念数量与种群规模对 Geacob 算法的召回率的影响,如图5所示:虽然随着种群规模和演化代数的增加,召回率也随之提高的趋势不变,但是整体的召回率数值明显下降。

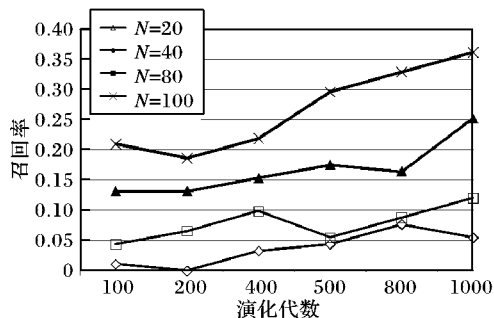


图5 种群和演化代数不同情况下 Geacob 算法的召回率(数据集3)

实际上,虽然形式背景的增大导致了结果形式概念的增多,然而形式背景的增大所导致的搜索空间的增加远远大于

结果形式概念的增多。根据本文概念个体编码的结构(定义5),搜索空间为  $2^{|O|+|A|}$ ,所以搜索空间随形式背景的增大成指数倍增加。这样就导致了种群规模和演化代数相同的情况下,虽然目标概念增多,但是出现召回率下降的现象。

通过以上实验表明,Geacob 算法可以有效地获得形式概念,但是由于遗传算法的固有特性和形式概念构造的特点(搜索空间与形式背景大小呈指数关系),使得该算法在大规模数据集上的应用具有局限性。

## 4 结语

本文探讨了应用遗传算法来生成形式概念的问题的可行性,并且给出了相应的遗传概念生成算法(Geacob)。实验结果表明该算法是可行的。另一方面,由于遗传算法具有优秀的并行计算可扩展性的特点,因此该算法的性能具有进一步提升的空间,这也是作者进一步工作的方向。

### 参考文献:

- [1] EIBEN A E, SMITH J E. Introduction to evolutionary computing [M]. Berlin: Springer-Verlag, 2003.
- [2] 潘正军,康立山,陈毓屏.演化计算[M].北京:清华大学出版社,1998.
- [3] 张卿,谢志鹏,刘宗田.基于变长编码遗传算法的最小缩减计算[J].小型微型计算机系统,2001,22(9):1057-1059.
- [4] PROBLEWSKI A. Find minimal reducts using genetic algorithm, ICS Research Report 16/95 [R]. Warsaw: Warsaw University of Technology, Institute of Computer Science, 1995.
- [5] 贾兆红,倪志伟,赵鹏.改进型遗传算法及其在数据挖掘中的应用[J].计算机应用,2002,22(9):31-33.
- [6] GANTER B, WILLE R. Formal concept analysis [M]. Berlin: Springer-Verlag, 1999.
- [7] 王黎明,张卓.基于 Iceberg 概念格并置集成的闭频繁项集挖掘算法[J].计算机研究与发展,2007,44(7):1184-1190.
- [8] 张卓,李石君,余伟,等.基于 Iceberg 概念格叠置半集成的全局闭频繁项集挖掘算法[J].小型微型计算机系统,2010,31(3):391-397.
- [9] GANTER B. Formal concept analysis: Algorithmic aspects [EB/OL]. [2010-03-06]. <http://www.math.tu-dmsden.de/~ganter/cl02/>.
- [10] GODIN R, MISSAOUI R, ALAOUI H. Incremental concept formation algorithms based on Galois (concept) lattices [J]. Computational Intelligence, 1995, 11(2): 246-267.
- [11] VALTCHEV P, MISSAOUI R, LEBRUN P. A partition-based approach towards constructing Galois (concept) lattices [J]. Discrete Mathematics, 2002, 256(3): 801-829.
- [12] VALTCHEV P, MISSAOUI R. Building concept (Galois) lattices from parts: generalizing the incremental methods [C]// ICCS'01: Proceedings of the 9th International Conference on Conceptual Structures: Broadening the Base, LNCS 2120. Berlin: Springer-Verlag, 2001: 290-303.
- [13] 李云,刘宗田,陈峻,等.多概念格的横向合并算法[J].电子学报,2004,32(11):1849-1854.
- [14] 齐红,刘大有,胡成全,等.基于搜索空间划分的概念生成算法[J].软件学报,2005,16(12):2029-2035.
- [15] Balloons Data Set, Machine Learning Repository [DB/OL]. [2010-03-01]. <http://archive.ics.uci.edu/ml/datasets/Balloons>.