

DOI:10.11992/tis.201507063
网络出版地址: <http://www.cnki.net/kcms/detail/23.1538.tp.20151110.1354.024.html>

一种基于最大满矩阵生成概念格的算法

郭伦众, 宋振明
(西南交通大学 数学学院, 四川 成都 610097)

摘要:形式概念分析的核心是概念格,它在本质上描述了对象和属性之间的联系,表明了概念之间的泛化和例化关系,因此概念格的构造就显得尤为的重要。从形式背景的关系矩阵出发,扫描形式背景的行和列找出属性值为 1 的全部满矩阵,定义了最大满矩阵的概念,证明了最大满矩阵是概念矩阵的充要条件。并在此理论上提出了一种基于最大满矩阵生成概念格的算法,并对所提出的算法进行了理论论证。通过实例的运算,验证了该算法的有效性。

关键词:概念;形式概念分析;矩阵;算法

中图分类号: TP18 **文献标志码:** A **文章编号:** 1673-4785(2015)06-0838-05

中文引用格式:郭伦众,宋振明.一种基于最大满矩阵生成概念格的算法[J].智能系统学报,2015,10(6):838-842.

英文引用格式:GUO Lunzhong, SONG Zhenming. A novel concept-lattice acquisition approach based on the greatest full matrix of formal context[J]. CAAI Transactions on Intelligent Systems, 2015, 10(6): 838-842.

A novel concept-lattice acquisition approach based on the greatest full matrix of formal context

GUO Lunzhong, SONG Zhenming
(School of Mathematics, Southwest Jiaotong University, Chengdu 610097, China)

Abstract:The concept lattice plays a crucial role in formal concept analysis, which describes essential object relations and attributes and illustrates the relation between generalization and specialization. Therefore, the development of effective methods for generating a concept lattice has been an important research topic. Based on the relational matrix of formal context, the rows and columns of the formal context are scanned to identify the full matrix. In this study, we define the concept of the greatest full matrix, and we prove that the greatest full matrix is a necessary and sufficient condition for the concept matrix. We also propose and theoretically demonstrate a concept-lattice generation algorithm based on the greatest full matrix. Finally, we provide an example to illustrate the rationality and effectiveness of the proposed algorithm.

Keywords: concept; formal concept analysis; matrix; algorithm

20 世纪 80 年代初,德国的 Wille 授提出了形式概念分析理论^[1]。到目前,概念格作为形式概念分析的核心数据结构,已经成为一种用于数据组织和数据分析的形式化工具,在数字图书馆及文献检索、软件工程、数据挖掘等许多领域得到了广泛应用^[2]。概念格的构造是应用概念格的前提,因此概

念格的构造效率就成为人们一直关注的焦点,提出了许多的构造算法。例如:Godin^[3]提出了概念格的渐进生成算法;T.B.Ho^[4]提出了基于概念格的概念聚类算法;刘宗田等^[5]对 Godin 算法进行了改进;林春杰等^[6]提出了基于不完备信息的近似概念格改进的概念格增量构造算法;李海霞^[7]提出了基于概念格 Hasse 图的一种对象渐减式构造算法;崔芳婷等^[8]根据用户的需求,把用户感兴趣的背景知识定义为约束,提出一种基于约束的模糊概念格构造算

法;刘宏英等^[9] 基于对多维序列的理解,提出一种新的多维概念格并给出了渐进式构造算法。本文首先将形式背景看成一个 0、1 的关系矩阵,然后将形式背景化为既约的^[10],最后提出了一种基于最大满矩阵来获取概念格的算法。

1 基本概念

下面给出形式背景的相关概念。
定义 1^[10] 设 U 是对象集合, M 是属性的集合, I 是 2 个集合 U 与 M 之间的关系,则称三元组 $K=(U,M,I)$ 为一个形式背景(简称背景)。

对于 $u \in U, m \in M, (u, m) \in I$ (或写作 uIm) 表示对象 u 具有属性 m 。

定义 2^[10] 设 $K=(U,M,I)$ 是一个背景。对于 $A \subseteq U, B \subseteq M$ 。令

$$f(A) = \{m \in M \mid \forall u \in A, uIm\}$$
$$g(B) = \{u \in U \mid \forall m \in B, uIm\}$$

如果 $A、B$ 满足 $f(A) = B, g(B) = A$, 则称二元组 (A,B) 是一个概念。 A 是概念 (A,B) 的外延, B 是概念 (A,B) 的内涵。

用 $L(U,M,I)$ 表示背景 $K=(U,M,I)$ 上的所有概念集合。

形式背景 K_1 如表 1 所示, 其中对象集 $U = \{u_1, u_2, u_3, u_4\}$, 属性集 $M = \{a, b, c, d, e, f, g\}$, 1 表示所对应的对象和属性具有关系 I , 0 表示所对应的对象和属性不具有关系 I 。

表 1 形式背景 K_1

Table 1 The formal context K_1							
U	a	b	c	d	e	f	g
u_1	1	0	1	0	0	1	1
u_2	1	1	1	0	1	1	1
u_3	0	1	0	0	1	1	0
u_4	0	0	1	1	1	1	1

实际上,该形式背景完全由矩阵

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

刻画,称此矩阵为形式背景矩阵。

可以验证这个背景的全部概念是:
 $(u_1u_2u_3u_4, f)、(u_1u_2, acfg)、(u_2u_3u_4, ef)、$
 $(u_1u_2u_4, cfg)、(u_2u_4, cefg)、(u_2u_3, bef)、(u_2,$
 $abcefg)、(u_4, cdefg)、(\Phi, abcdefg)。$

设 $K=(U,M,I)$ 是一个形式背景, $H_1 =$

$(A_1, B_1), H_2 = (A_2, B_2)$ 是 K 的 2 个概念,在概念集 $L(U,M,I)$ 上定义关系:

$$H_1 \leq H_2 \Leftrightarrow A_1 \subseteq A_2 \text{ (或 } B_2 \subseteq B_1)$$

在概念集 $L(U,M,I)$ 上定义交并运算分别为
 $(A_1, B_1) \wedge (A_2, B_2) = (A_1 \cap A_2, f(g(B_1 \cup B_2)))$
 $(A_1, B_1) \vee (A_2, B_2) = (g(f(A_1 \cup A_2)), B_1 \cap B_2)$

可以证明 $L(U,M,I)$ 对此运算构成完备格,称 $L(U,M,I)$ 为概念格。

为了让形式背景尽可能的简单,下面给出净化背景的定义。

定义 3^[10] 如果任意满足 $f(u) = f(h)$ 的 2 个元素 $u, h \in U$ 都有 $u = h$, 且对任意满足 $g(m) = g(n)$ 的元素 $m, n \in M$, 都有 $m = n$, 则称背景 (U,M,I) 是净化的。

例 1 把表 1 净化后,得到形式背景 K_2 ,如表 2。

表 2 形式背景 K_2

Table 2 The formal context K_2						
U	a	b	c, g	d	e	f
u_1	1	0	1	0	0	1
u_2	1	1	1	0	1	1
u_3	0	1	0	0	1	1
u_4	0	0	1	1	1	1

为了方便,给具有某种特点的对象或属性一个统一的名称。

定义 4^[10] 当 $m \notin X$, 但 $g(m) = g(X)$, 称 m 为可约属性, 称 m 产生的属性概念 $(g(m), f(g(m)))$ 为 \wedge 可约的属性概念。

对应地,当 $u \notin H$, 但 $f(u) = f(H)$ 时,称 u 为可约对象,称 u 产生的对象概念 $(g(f(u)), f(u))$ 为 \vee 可约的对象概念。

显然 \wedge 可约的属性与 \vee 可约的对象是可以删除的。如果所有对象都具有某个属性 m , 则称 m 为“满列属性”;对应地,如果有某个对象 u 具有所有属性,则称 u 为“满行对象”。由定义可得“满列属性”与“满行对象”都是可约的。

如果一个净化背景的每个对象概念都是 \vee 不可约的,则称这个背景为行既约的。如果一个净化背景的每个属性概念都是 \wedge 不可约的,则称这个背景为列既约的。既是行既约的又是列既约的,则称其是既约的。

例 2 把表 2 化成既约的,得形式背景 K_3 , 如表 3 所示。

表 3 形式背景 K_3
Table 3 The formal context K_3

U	a	b	c, g	d	e
u_1	1	0	1	0	0
u_2	1	1	1	0	1
u_3	0	1	0	0	1
u_4	0	0	1	1	1

这里把属性 f 删除,将属性 c 与 g 合并成一个属性也不会影响概念格的结构。

由于对具有相同内涵的对象和具有相同外延的属性的各自合并所得的概念格与原概念格同构。所以在生成概念之前可以先把形式背景化为既约的,这样就会使形式背景变得简单些。

2 满矩阵与概念

定义 5 设 (U, M, I) 为一形式背景,在其对应的形式背景矩阵中,取出对象集 $A \subseteq U$ 对应的任意行和属性集 $B \subseteq M$ 对应的任意列组成新的矩阵 M_A^B , 则称 M_A^B 为形式背景矩阵的子矩阵。

若子矩阵 M_A^B 的每个元素都为 1, 则称子矩阵 M_A^B 为满矩阵。

在满矩阵 M_A^B 中, 规定矩阵 M_A^B 的列数即 $|B|$ 为矩阵的属性秩, 记为 $r(M_A^B) = |B|$; 矩阵 M_A^B 的行数即 $|A|$ 为矩阵的对象秩, 记为 $l(M_A^B) = |A|$, 其中 $|\cdot|$ 表示集合中元素的个数。

定义 6 在形式背景矩阵中, 对于对象集 $A \subseteq U$, 如果属性集 $B \subseteq M$ 有

$$r(M_A^Y) \leq r(M_A^B), \forall Y \subseteq M$$

成立, 则称 M_A^B 为对象集 A 对应的最大属性满矩阵。

对偶地, 可以定义最大对象满矩阵:

在形式背景矩阵中, 对于属性集 $B \subseteq M$, 如果对象集 $A \subseteq U$ 有

$$l(M_X^B) \leq l(M_A^B), \forall X \subseteq U$$

成立, 则称 M_A^B 为属性集 B 对应的最大对象满矩阵。

说明: 给定对象集, 则其对应的最大属性满矩阵是唯一的; 给定属性集, 则其对应的最大对象满矩阵是唯一的。

性质 1 设 (U, M, I) 为一形式背景, 对任意的对象集 $A_1 \subseteq A_2 \subseteq U$, 有

$$r(M_{A_2}^{B_2}) \leq r(M_{A_1}^{B_1})$$

成立, $M_{A_1}^{B_1}$ 、 $M_{A_2}^{B_2}$ 分别为 A_1 、 A_2 的最大属性满矩阵。

说明: 该性质的依据就是对象集越大, 则其具有

的共同属性不会多于其子集所具有的共同属性。所以对对象集 A_2 具有的共同属性应该小于或等于其子集 A_1 具有的共同的属性。

性质 2 设 (U, M, I) 为一形式背景, 对任意的属性集 $B_1 \subseteq B_2 \subseteq M$, 有

$$l(M_{A_2}^{B_2}) \leq l(M_{A_1}^{B_1})$$

成立, $M_{A_1}^{B_1}$ 、 $M_{A_2}^{B_2}$ 分别为 B_1 、 B_2 的最大对象满矩阵。

说明: 该性质的依据就是具有的共同属性越多, 则其对象集的个数不会增多。所以具有属性集 B_2 的对象个数应该小于或等于具有属性集 B_1 的对象个数。

定义 7 若 (A, B) 为形式概念, 则称由对象集 A 所在的行和属性集 B 所在的列组成的矩阵 M_A^B 为概念矩阵。

定理 1 概念矩阵一定是最大属性满矩阵和最大对象满矩阵。

证明: 由定义 5~7 可直接得到结论。反之, 定理 1 不一定成立, 即最大属性满矩阵, 最大对象满矩阵不一定是概念矩阵。如在例 1 中, 对象 u_1 与属性 $acfg$ 组成的子矩阵为最大属性满矩阵, 但由概念的定义知 $(u_1, acfg)$ 不是形式概念, 又对象 u_1u_2 和属性 a 组成的子矩阵为最大对象满矩阵, 但由概念的定义知 (u_1u_2, a) 不是形式概念, 即可得出定理 1 的逆不成立。

下面给出满矩阵是概念矩阵的一个充分条件:

定理 2 若满矩阵 M_A^B , 既是对象集 A 的最大属性满矩阵, 且是属性集 B 的最大对象满矩阵, 则满矩阵 M_A^B 是概念矩阵。

证明 反证, 假设满矩阵 M_A^B 不是概念矩阵, 说明 (A, B) 不是概念即 $f(A) \neq B$ 或 $g(B) \neq A$ 成立。若 $f(A) \neq B$, 则存在对象集 $A' \supseteq A$, 使 $f(A') = B$ 成立, 也就是说属性集 B 有更大的对象满矩阵, 与满矩阵 M_A^B 是属性集 B 的最大对象满矩阵矛盾; 同理可证, 若 $g(B) \neq A$, 则存在属性集 $B' \supseteq B$, 使 $g(B') = A$ 成立, 也就是说对象集 A 有更大的属性满矩阵, 与满矩阵 M_A^B 是对象集 A 的最大属性满矩阵矛盾, 即假设不成立, 得证。

为了找出形式背景中所有的概念, 根据定理 1, 可以先找出它的所有最大属性满矩阵或最大对象满矩阵, 进而根据定理 2 判断那些满矩阵为概念矩阵, 最后得到所有的概念。

3 方法

设 (U, M, I) 是一形式背景, 显然, 它可以看作

是一个由 0、1 表示的矩阵。利用矩阵获得概念的生成算法:

1) 将形式背景化为既约背景。首先净化背景, 及合并具有相同内涵的对象和具有相同外延的属性, 然后删除其内涵等于其他对象内涵交的那些对象, 对应地删除其外延等于其他属性外延的交的那些属性, 得到的就是既约背景。

2) 根据定理 1, 先找出各形式背景矩阵对应的非零矩阵的最大属性满矩阵或最大对象满矩阵, 然后根据定理 2 判断那些满矩阵是概念矩阵, 最后, 在概念矩阵中加上矩阵 M_G^Φ 与 M_Φ^M 即为所有的概念矩阵。

下面用具体例子说明该算法的过程。

例 3 表 3 是表 1 给出的形式背景 (U, M, I) 的既约背景, 它可以表示为一个 0、1 的关系矩阵, 如下所示, 现在用给出的概念格生成算法来找出它的概念格:

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

通过上面的矩阵可以计算表 3 的概念层次如下:

1) 形式背景矩阵对应的非零矩阵的最大属性满矩阵有:

$$M_{u_1}^{ac}, M_{u_2}^{abce}, M_{u_3}^{be}, M_{u_4}^{cde}, M_{u_1u_2}^{ac}, M_{u_2u_3}^{be}, M_{u_2u_4}^{ce}, M_{u_3u_4}^e, M_{u_1u_2u_4}^c, M_{u_2u_3u_4}^e$$

这里, ac 代表集合 $\{a, c\}$, 余同。

2) 由定理 2 得出概念矩阵有:

$$M_{u_2}^{abce}, M_{u_4}^{cde}, M_{u_1u_2}^{ac}, M_{u_2u_3}^{be}, M_{u_2u_4}^{ce}, M_{u_1u_2u_4}^c, M_{u_2u_3u_4}^e, M_{u_1u_2u_3u_4}^\Phi, M_{\Phi}^{abcde}$$

最后, 得到概念所有的概念:

$$(u_2abce), (u_4, cde), (u_1u_2, ac), (u_2u_3, be), (u_2u_4, ce), (u_1u_2u_4, c), (u_2u_3u_4, e), (u_1u_2u_3u_4, \Phi), (\Phi, abcde)$$

另外, 可以先找出所有的最大对象满矩阵, 再根据定理 2 得到概念矩阵, 进而得到所有的概念。

为了进一步说明该算法的有效性, 给出实例:

例 4^[11] 软件的结构设计主要是针对软件的数据结构模式, 在需求分析的基础上, 通过合理化组织, 加以分析设计, 从而得出设计的方法。下面假设一个系统, 这个系统包括 6 个过程(分别假设为 $P_1, P_2, P_3, P_4, P_5, P_6$) 以及一个包含 9 个字段的记录。

利用相关的分析器可以分析出每个项目在每一个过程中的使用情况(“1”表示使用), 如表 5 所示。用上面算法构造出对应的概念。

根据上文给出的定义, 删除可约对象 β , 合并相同对象后, 可以写出表 4 对应的既约背景如表 5。

表 4 9 个项目与 6 个特征的对应关系
Table 4 The relation of nine projects and six characteristics

参数	P_1	P_2	P_3	P_4	P_5	P_6
NAME: α	1	0	0	0	0	1
TITLE: β	1	0	0	0	0	0
INTTIAL: γ	1	0	0	0	1	0
PREFIX: δ	1	0	0	0	1	0
NUMBER: ε	0	0	0	1	0	1
NUMBER-EXT: ζ	0	0	0	1	0	1
ZIPCD: θ	0	0	0	1	0	1
SIREFT: ι	0	0	1	1	0	0
CITY: κ	0	1	0	1	0	0

表 5 既约背景						
Table 5 Reducible formal context						
参数	P_1	P_2	P_3	P_4	P_5	P_6
NAME: α	1	0	0	0	0	1
INTTIAL: γ						
PREFIX: δ	1	0	0	0	1	0
NUMBER: ε						
NUMBER-EXT: ζ	0	0	0	1	0	1
ZIPCD: θ						
SIREFT: ι	0	0	1	1	0	0
CITY: κ	0	1	0	1	0	0

下面写出它既约背景对应的关系矩阵:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

通过上面的矩阵可以计算表 5 的概念层次如下:

1) 形式背景矩阵对应的非零矩阵的最大属性满矩阵有:

$$M_{\alpha}^{P_1P_6}, M_{\gamma}^{P_1P_5}, M_{\varepsilon}^{P_4P_6}, M_{\iota}^{P_3P_4}, M_{\kappa}^{P_2P_4}, M_{\alpha\gamma}^{P_1}, M_{\alpha\varepsilon}^{P_6}, M_{\varepsilon\iota}^{P_4}, M_{\varepsilon\kappa}^{P_4}, M_{\iota\kappa}^{P_4}, M_{\varepsilon\iota\kappa}^{P_4}$$

2) 由定理 2 得出概念矩阵有:

$$M_{\alpha}^{P_1P_6}, M_{\gamma}^{P_1P_5}, M_{\varepsilon}^{P_4P_6}, M_{\iota}^{P_3P_4}, M_{\kappa}^{P_2P_4}, M_{\alpha\gamma}^{P_1}, M_{\alpha\varepsilon}^{P_6}, M_{\varepsilon\iota\kappa}^{P_4}, M_{\Phi}^{P_1P_2P_3P_4P_5P_6}, M_{\alpha\gamma\varepsilon\iota\kappa}^{\Phi}$$

3)最后,得到概念所有的概念:

$$(\alpha, P_1 P_6), (\gamma, P_1 P_5), (\varepsilon, P_4 P_6), (\iota, P_3 P_4), \\ (\kappa, P_2 P_4), (\alpha\gamma, P_1), (\alpha\varepsilon, P_6), (\varepsilon\iota\kappa, P_4), \\ (\Phi, P_1 P_2 P_3 P_4 P_5 P_6), (\alpha\gamma\varepsilon\iota\kappa, \Phi)$$

5 结束语

目前,已有不少概念格的建格算法,本文从矩阵的角度出发,利用矩阵与概念之间的联系,定义了一种新的基于概念的概念格——最大满矩阵,找出了最大满矩阵与概念之间的联系,进而得出了一种基于矩阵的概念格生成算法,具体例子说明该算法是有效的。

参考文献:

- [1] WILLE R. Restructuring lattice theory: an approach based on hierarchies of concepts[M]//RIVAL I. Ordered Sets. Berlin Heidelberg: Springer, 1982: 445-470.
- [2] OOSTHULZEN G D. The application of concept lattice to machine learning[R]. South Africa: University of Pretoria, 1996.
- [3] GODIN R, MISSAOUI R, ALAOUI H. Incremental concept formation algorithms based on Galois (concept) lattices[J]. Computational Intelligence, 1995, 11(2): 246-267.
- [4] HO T B. Incremental conceptual clustering in the framework of Galois lattice[C]//LU H, LIU H, MOTODA H. KDD: Techniques and Applications. Singapore: World Scientific, 1997: 49-64.
- [5] 谢志鹏, 刘宗田. 概念格的快速渐进式构造算法[J]. 计算机学报, 2002, 25(5): 490-496.
XIE Zhipeng, LIU Zongtian. A fast incremental algorithm for building concept lattice[J]. Chinese Journal of Computers, 2002, 25(5): 490-496.
- [6] 林春杰, 普杰信, 张瑞玲. 近似概念格及其增量构造算法研究[J]. 计算机应用研究, 2012, 29(1): 25-27.
LIN Chunjie, PU Jinxin, ZHANG Ruilin. Approximation concept lattice and incremental constructing algorithm[J]. Application Research of Computers, 2012, 29(1): 25-27.
- [7] 李海霞. 基于 Hasse 图的概念格的一种渐减式构造算法[J]. 河南科技学院学报, 2015, 43(3): 57-60, 66.
LI Haixia. A decreasing algorithm of concept lattice based on Hasse diagram[J]. Journal of Henan Institute of Science and Technology, 2015, 43(3): 57-60, 66.
- [8] 崔芳婷, 王黎明, 张卓. 基于约束的模糊概念格构造算法[J]. 计算机科学, 2015, 42(8): 288-293, 318.
CUI Fangting, WANG Liming, ZHANG Zhuo. Construction algorithm of fuzzy concept lattice based on constraints[J]. Computer Science, 2015, 42(8): 288-293, 318.
- [9] 刘宏英, 郭显娥, 胡小珍. 多维概念格及其构造算法[J]. 计算机工程与应用, 2012, 48(12): 96-99, 111.
LIU Hongying, GUO Xian'e, HU Xiaozhen. Multidimensional concept lattice and constructing algorithm[J]. Computer Engineering and Applications, 2012, 48(12): 96-99, 111.
- [10] 马垣, 曾子维, 迟呈英, 等. 形式概念及其新进展[M]. 北京: 科学出版社, 2010: 11-24.
- [11] 蒋平, 任胜兵, 林鹃. 形式概念分析在软件工程中的应用[J]. 计算机技术与发展, 2008, 18(4): 127-129, 213.
JIANG Ping, REN Shengbing, LIN Juan. Using formal concept analysis for software engineering[J]. Computer Technology and development, 2008, 18(4): 127-129, 213.

作者简介:



宋振明,男,教授,硕士生导师,主要研究方向为智能信息处理、运筹与控制、不确定性推理。



郭伦众,女,1992年生,硕士研究生,主要研究方向为智能信息处理。