

一种查找形式背景箭头关系的快速算法

杨武松¹ 吕跃进¹ 李金海²

(广西大学数学与信息科学学院 南宁 530004)¹ (西安交通大学理学院 西安 710049)²

摘要 在形式概念分析中,“箭头关系”对简化形式背景、识别兼容子背景等起着至关重要的作用,因此如何快速确定一个给定形式背景的对象与属性间的箭头关系是一个值得探讨的问题。首先针对形式背景给出对象与属性间的箭头关系的等价判定定理,在此基础上提出一个查找形式背景的箭头关系的快速算法,并通过实例与数值实验证实了该算法的可行性与有效性。

关键词 形式概念分析,形式背景,箭头关系,快速算法

中图法分类号 TP18,O153 **文献标识码** A

Quick Algorithm of Determining Arrow Relations of a Formal Context

YANG Wu-song¹ LV Yue-jin¹ LI Jin-hai²

(School of Mathematics and Information Sciences, Guangxi University, Nanning 530004, China)¹

(School of Science, Xi'an Jiaotong University, Xi'an 710049, China)²

Abstract In formal concept analysis, the so-called “arrow relation” plays an important role in reducing the size of a formal context and recognizing a compatible subcontext of a formal context. Thus, how to quick determine the arrow relation between the objects and the attributes of a formal context is worth to be investigated. This study first gave an equivalent theorem of checking whether or not there is an arrow relationship between an object and an attribute of a formal context. Then a quick algorithm of determining the arrow relation between the objects and the attributes of a formal context was developed. Finally, a real example and numerical experiments were used to demonstrate the feasibility and efficiency of the proposed algorithm.

Keywords Formal concept analysis, Formal context, Arrow relation, Quick algorithm

1 引言

形式概念分析^[1]是20世纪80年代初由R. Wille教授提出的,经过将近30年的发展,它已在数字图书馆、软件工程、数据分析与数据挖掘等领域得到成功应用^[2-6]。形式概念分析的两个理论基石是形式背景与形式概念。当形式背景较大时,可能只需要考虑它的某一部分即可,即子背景,尤其是兼容子背景。另外,根据特定需要,也可以通过约简一个给定形式背景的对象或属性,得到它的简化子背景。

为了识别兼容子背景、净化背景等,R. Wille在形式背景中提出了所谓的箭头关系^[7,8]。文献[9]对利用箭头关系实现形式背景的约简做了详细介绍。软件ConExp在version1.3中加入了显示箭头关系的功能^[10]。借助箭头关系约简形式背景时,仅需在带有箭头关系的形式背景中简单地合并相同行或列即可,且该过程可以并行地计算。因此,研究如何在形式背景中快速查找箭头关系显得非常必要。

另外,形式背景的双建立也可以通过箭头关系来描述^[8,11]。文献[12]还利用箭头关系成功给出了一种由 M_3 生

成的变换中双建立概念格的形式背景的刻画。因此,如何在形式背景中提出查找箭头关系的快速算法是一个值得研究的重要课题,因为直接利用箭头关系的原始定义来计算大型形式背景的箭头关系是很复杂且不容易实现的^[9]。

对于形式背景箭头关系快速查找的研究,目前未检索到相关文献,故本文尝试提出一种箭头关系快速查找的算法。首先,基于形式背景给出了对象与属性间的箭头关系的等价判定定理,由此定理可以对给定对象(属性)快速查找到满足箭头关系的所有属性(对象);在此基础上提出了一个形式背景的箭头关系快速查找算法。通过数值实验表明,本文提出的快速算法可以有效地降低形式背景中查找箭头关系所需的计算时间。

2 基础知识

在给出箭头关系的定义与查找算法之前,先简要介绍与本文有关的一些基本概念和性质,其详细的描述可以参见文献[1,7,8]。

为了方便,下文中有时忽略掉“形式”这个修饰词,把“形

到稿日期:2011-10-20 返修日期:2012-02-20 本文受国家自然科学基金(70861001),广西自然科学基金(桂科自0991027)资助。

杨武松(1987—),男,硕士生,CCF会员,主要研究方向为数据挖掘、粗糙集与概念格,E-mail:wusongyang5@163.com;吕跃进(1958—),男,教授,硕士生导师,主要研究方向为数据挖掘、粗糙集与概念格、运筹与控制,E-mail:lvyyjin@126.com(通信作者);李金海(1984—),男,博士生,主要研究方向为粗糙集、概念格与算法分析。

式背景”简称为“背景”，把“形式概念”简称为“概念”。

定义 1(形式背景)^[1] 称 (U, A, I) 为一个形式背景，其中 $U = \{x_1, x_2, \dots, x_n\}$ 为对象集，每个 $x_i (i \leq n)$ 称为一个对象； $A = \{a_1, a_2, \dots, a_m\}$ 为属性集，每个 $a_j (j \leq m)$ 称为一个属性； I 为 U 和 A 之间的二元关系，即 $I \subseteq U \times A$ 。若 $(x, a) \in I$ ，则称 x 具有属性 a ；若 $(x, a) \notin I$ ，则称 x 不具有属性 a 。

若用 1 表示 $(g, m) \in I$ ，用 0 表示 $(g, m) \notin I$ ，那么形式背景可以表示成数据取值为 0 或 1 的表格。

定义 2(形式概念)^[8] 设 (U, A, I) 为一个形式背景，在对象集 $X \subseteq U$ 和属性集 $B \subseteq A$ 上分别定义运算：

$$X^* = \{a \in A \mid \forall x \in X, (x, a) \in I\}$$

$$B^* = \{x \in U \mid \forall a \in B, (x, a) \in I\}$$

如果一个二元组 (X, B) 满足 $X^* = B$ ，且 $B^* = X$ ，则称 (X, B) 是一个形式概念，简称概念。其中 X 称为概念的外延， B 称为概念的内涵。

定义 3(概念格)^[8] 若 $(X_1, B_1), (X_2, B_2)$ 是某个背景 (U, A, I) 上的两个概念，记

$$(X_1, B_1) \leq (X_2, B_2) \Leftrightarrow X_1 \subseteq X_2 (\Leftrightarrow B_1 \supseteq B_2)$$

则称 (X_1, B_1) 是 (X_2, B_2) 的子概念， (X_2, B_2) 是 (X_1, B_1) 的超概念。显然，“ $\leq$ ”是一个偏序关系。 (U, A, I) 的所有概念连同偏序关系“ $\leq$ ”记为 $L(U, A, I)$ ，称为 (U, A, I) 的概念格。

定理 1^[8] 设 (U, A, I) 是一个背景， $\forall X_1, X_2, X \subseteq U$ 是对象集， $\forall B_1, B_2, B \subseteq A$ 是属性集，则下列性质成立：

- (1) $X_1 \subseteq X_2 \Rightarrow X_2^* \subseteq X_1^*, B_1 \subseteq B_2 \Rightarrow B_2^* \subseteq B_1^*$ ；
- (2) $X \subseteq X^{**}, B \subseteq B^{**}$ ；
- (3) $X^* = X^{***}, B^* = B^{***}$ ；
- (4) $X \subseteq B^* \Leftrightarrow B \subseteq X^*$ ；
- (5) $(X_1 \cup X_2)^* = X_1^* \cap X_2^*, (B_1 \cup B_2)^* = B_1^* \cap B_2^*$ ；
- (6) (X^{**}, X^*) 和 (B^*, B^{**}) 都是 (U, A, I) 的概念。

对定理 1 的(5)进行推广，有如下定理。

定理 2^[8] 设 (U, A, I) 是一个背景，若 T 是一个索引集，而且对每个 $t \in T, X_t \subseteq U$ 是对象集， $B_t \subseteq A$ 是属性集，则

$$(\bigcup_{t \in T} X_t)^* = \bigcap_{t \in T} X_t^*, (\bigcup_{t \in T} B_t)^* = \bigcap_{t \in T} B_t^*$$

定理 3(概念格基本定理)^[1] 概念格 $L(U, A, I)$ 是一个完全格，其下确界和上确界分别为

$$\bigwedge_{i \in T} (X_i, B_i) = (\bigcap_{i \in T} X_i, (\bigcup_{i \in T} B_i)^{**})$$

$$\bigvee_{i \in T} (X_i, B_i) = ((\bigcup_{i \in T} X_i)^{**}, \bigcap_{i \in T} B_i)$$

一个完全格 V 与 $L(U, A, I)$ 同构，当且仅当存在映射 $\gamma: U \rightarrow V$ 和 $\mu: A \rightarrow V$ ，且 $\gamma(U)$ 在 V 中上确界稠密， $\mu(A)$ 在 V 中下确界稠密；对所有 $x \in U$ 及 $a \in A$ 都有 $(x, a) \in I \Leftrightarrow \gamma x \leq \mu a$ 。特别地， V 与 $L(V, V, \leq)$ 同构。

在下文中， $\forall x \in U$ ，记 $\{x\}^*$ 为 x^* ； $\forall a \in A$ ，记 $\{a\}^*$ 为 a^* 。由定理 1 的(6)可知， (x^{**}, x^*) 与 (a^*, a^{**}) 都是概念，称 (x^{**}, x^*) 为对象概念，记作 γx ；对偶地，称 (a^*, a^{**}) 为属性概念，记作 μa 。

3 形式背景的箭头关系及其判定定理

定义 4(箭头关系)^[8] 如果 (U, A, I) 是一个背景， $x \in U$ 是一个对象， $a \in A$ 是一个属性，那么

- (1) $x \downarrow a$ 当且仅当 $(x, a) \notin I$ ，并且 $x^* \subset y^* \Rightarrow (y, a) \in I$ ；
- (2) $x \uparrow a$ 当且仅当 $(x, a) \notin I$ ，并且 $a^* \subset b^* \Rightarrow (x, b) \in I$ ；

(3) $x \updownarrow b$ 当且仅当 $x \downarrow a$ ，并且 $x \uparrow a$ 。

利用箭头关系的定义在形式背景中查找满足箭头关系的对象-属性对时，需要不断判断一个集合是否真包含于另一个集合。对于给定的 $x \in U$ ，计算 x^* 的时间复杂性为 $O(|A|)$ ；对于给定的 $a \in A$ ，计算 a^* 的时间复杂性为 $O(|U|)$ 。对于给定的 $x \in U$ 和 $a \in A$ ，对任意 $y \in U$ ，判断一个“ $x^* \subset y^*$ 是否满足”的时间复杂性为 $O(|A|)$ ，从而对给定的 $x \in U$ 和 $a \in A$ ，判断“ $x \downarrow a$ 是否满足”的时间复杂性为 $O(|U| |A|)$ ；同理可知，判断“ $x \uparrow a$ 是否满足”的时间复杂性也为 $O(|U| |A|)$ 。因此，基于定义 4 查找背景 (U, A, I) 中箭头关系的时间复杂性为 $O(|U|^2 |A|^2)$ 。

为简化箭头关系的查找过程，减小计算的时间复杂性，下面先给出一个定理，该定理给出了与箭头关系定义等价的条件。

定理 4^[8] 设 (U, A, I) 是一个背景， $x \in U$ 是一个对象， $a \in A$ 是一个属性， γx 表示对象概念， μa 表示属性概念，令

$$\sup(\gamma x) = \bigvee \{(X, B) \in L(U, A, I) \mid (X, B) < \gamma x\}$$

$$\inf(\mu a) = \bigwedge \{(X, B) \in L(U, A, I) \mid (X, B) > \mu a\}$$

则

$$x \downarrow a \Leftrightarrow \gamma x \wedge \mu a = \sup(\gamma x) \neq \gamma x \quad (1)$$

$$x \uparrow a \Leftrightarrow \gamma x \vee \mu a = \inf(\mu a) \neq \mu a \quad (2)$$

文献[8]中对此定理未给出证明，本文给出补充证明。

证明：下面仅证明式(2)，式(1)可以类似证明。

($\Rightarrow$) 易知 $\gamma x \vee \mu a \geq \mu a$ ，由 $\inf(\mu a) = \bigwedge \{(X, B) \in L(U, A, I) \mid (X, B) > \mu a\}$ 可得， $\gamma x \vee \mu a \geq \inf(\mu a)$ 。欲证 $\gamma x \vee \mu a = \inf(\mu a)$ ，下面需证 $\gamma x \vee \mu a \leq \inf(\mu a)$ ，由于 $\mu a \leq \inf(\mu a)$ ，故只需 $\gamma x \leq \inf(\mu a)$ 满足即可。

设 $(X, B) \in L(U, A, I)$ ，且 $(X, B) > \mu a$ 。由 $\mu a = (a^*, a^{**})$ 得 $a^* \subset X$ ，又 $X = B^*$ ，从而 $a^* \subset B^*$ 。由 $\forall b \in B$ ，有 $B^* \subseteq b^*$ ，则 $a^* \subset b^*$ 。根据 $x \uparrow a$ 的定义，有 $(x, b) \in I$ ，即 $b \in x^*$ ，再由 b 的任意性可得 $B \subseteq x^*$ 。故 $(x^{**}, x^*) \leq (X, B)$ ，亦即 $\gamma x \leq (X, B)$ 。根据概念 (X, B) 的取法可知， $\gamma x \leq \inf(\mu a)$ 。

假设 $\gamma x \vee \mu a = \mu a$ ，则 $\gamma x \leq \mu a$ ，根据定理 3(概念格基本定理)，有 $(x, a) \in I$ ，与 $x \uparrow a$ 的定义矛盾，所以 $\gamma x \vee \mu a \neq \mu a$ 。

因此，当 $x \uparrow a$ 成立时，需要满足 $\gamma x \vee \mu a = \inf(\mu a) \neq \mu a$ 。

($\Leftarrow$) 由 $\gamma x \vee \mu a \neq \mu a$ 可得， $(x, a) \notin I$ 。否则若 $(x, a) \in I$ ，由定理 3 有， $\gamma x \leq \mu a$ ，则 $\gamma x \vee \mu a = \mu a$ ，与 $\gamma x \vee \mu a \neq \mu a$ 矛盾。

对任意 $a \in A$ ，记 $N(a) = \{b \in A \mid a^* \subset b^*\}$ ，记 $(X, B) = ((N(a))^*, (N(a))^{**})$ ，显然 (X, B) 是一个概念。对任意 $b \in N(a)$ ，有 $a^* \subset b^*$ ，从而 $a^* \subset \bigcap_{b \in N(a)} b^*$ 。根据定理 2，有 $\bigcap_{b \in N(a)} b^* = (\bigcup_{b \in N(a)} \{b\})^* = (N(a))^*$ ，则 $a^* \subset (N(a))^*$ 。又 $\mu a = (a^*, a^{**})$ ，从而 $\mu a < (X, B)$ 。根据 $\inf(\mu a)$ 的定义可得， $\inf(\mu a) \leq (X, B)$ 。再由 $\gamma x \vee \mu a = \inf(\mu a)$ 得， $\gamma x \vee \mu a \leq (X, B)$ 。

由于 $\gamma x \vee \mu a = ((x^* \cap a^{**})^*, x^* \cap a^{**})$ ，则 $B \subseteq x^* \cap a^{**}$ 。而 $B = (N(a))^{**}$ ，故 $(N(a))^{**} \subseteq x^* \cap a^{**}$ 。根据定理 1(2)，有 $N(a) \subseteq (N(a))^{**}$ ，则 $N(a) \subseteq x^* \cap a^{**} \subseteq x^*$ 。设 $\forall b \in N(a)$ ，可得 $b \in x^*$ ，亦即当 $a^* \subset b^*$ 时，有 $(x, b) \in I$ 。

因此由 $\gamma x \vee \mu a = \inf(\mu a) \neq \mu a$ 可得， $(x, a) \notin I$ ，并且当 $a^* \subset b^*$ 时，有 $(x, b) \in I$ ，据定义 4 得， $x \uparrow a$ 成立。

定理 4 给出了与箭头关系定义等价的条件，但若利用此定理查找满足箭头关系的对象-属性对，则需要计算 $\sup(\gamma x)$

与 $\inf(\mu a)$, 而这个过程需要对背景中的每一个概念进行判断, 进而计算出 γx 的所有真“子概念”和 μa 的所有真“超概念”。为降低计算复杂性, 基于定理 4, 提出如下箭头关系判定定理, 利用此定理可以方便地计算出对给定的对象(属性)满足箭头关系的所有属性(对象)。

定理 5 设 (U, A, I) 是一个背景, 那么下列命题成立:

(1) 对任意 $x \in U$, 有

$$\{a \in A | x \downarrow a\} = (\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*) \cap p(x)$$

其中 $p(x) = A - x^*$;

(2) 对任意 $a \in A$, 有

$$\{x \in U | x \uparrow a\} = (\bigcap_{x \in q(a)} (a^{**} \cap x^*)^*) \cap q(a)$$

其中 $q(a) = U - a^*$ 。

证明: 这里仅证明式(1), 式(2)可以类似得证。

(i) 对于任意 $b \in \{a \in A | x \downarrow a\}$, 有 $x \downarrow b$ 成立。根据定义 4, 有 $(x, b) \notin I$, 且 $x^* \subset y^* \Rightarrow (y, b) \in I$ 。由 $(x, b) \notin I$ 可得 $b \notin x^*$, 即 $b \in p(x)$ 。下面证明 $b \in \bigcap_{a \in p(x)} (x^{**} \cap a^*)^*$ 成立。

对于任意的 $a \in p(x)$, 有 $x^* \subset x^* \cup \{a\}$, 进而有 $(x^* \cup \{a\})^* \subseteq x^{**}$ 。对任意 $y \in (x^* \cup \{a\})^*$, 有 $y^* \supseteq (x^* \cup \{a\})^{**} \supseteq x^{***}$, 即 $y^* \supseteq x^*$ 。如果 $y^* = x^*$, 则 $a \notin y^*$, 这与 $y \in (x^* \cup \{a\})^* \subseteq a^*$ 是矛盾的, 因此 $y^* \neq x^*$, 故有 $y^* \supset x^*$ 。再由 $x \downarrow b$ 的定义可得 $(y, b) \in I$, 即有 $y \in b^*$ 。因此 $(x^* \cup \{a\})^* \subseteq b^*$, 从而 $b \in b^{**} \subseteq (x^* \cup \{a\})^{**}$ 。根据定理 2, 有 $(x^* \cup \{a\})^{**} = (x^{**} \cap a^*)^*$, 故 $b \in (x^{**} \cap a^*)^*$ 。由 a 的任意性可得, $b \in \bigcap_{a \in p(x)} (x^{**} \cap a^*)^*$ 成立。

因此 $b \in (\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*) \cap p(x)$, 所以

$$\{a \in A | x \downarrow a\} \subseteq (\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*) \cap p(x)$$

(ii) 对于任意 $b \in (\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*) \cap p(x)$, 有 $b \in p(x)$ 且 $b \in \bigcap_{a \in p(x)} (x^{**} \cap a^*)^*$ 。因此, $b \notin x^*$ (即 $(x, b) \notin I$), 且对任意 $a \in p(x)$, 有 $b \in (x^{**} \cap a^*)^*$, 所以 $b^* \supseteq (x^{**} \cap a^*)^{**}$ 。由于 $(x^{**} \cap a^*)^{**} = (x^* \cup \{a\})^{***} = (x^* \cup \{a\})^*$, 故 $(x^* \cup \{a\})^* \subseteq b^*$ 。

对于任意 $y \in (x^* \cup \{a\})^*$, 显然有 $y \in b^*$ 。根据 $a \in p(x)$, 可得 $x^* \subset x^* \cup \{a\}$, 从而 $(x^* \cup \{a\})^* \subseteq x^{**}$, 故 $y \in x^{**}$ 。因此, $y^* \supseteq x^{***}$, 即 $y^* \supseteq x^*$ 。由 $y \in b^*$ 可知 $b \in y^*$, 但 $b \notin x^*$, 所以 $x^* \subset y^*$ 。从而 $x^* \subset y^*$ 时, 有 $(y, b) \in I$ 成立。根据定义 4 可知, $x \downarrow b$, 也即 $b \in \{a \in A | x \downarrow a\}$ 。这就证得:

$$(\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*) \cap p(x) \subseteq \{a \in A | x \downarrow a\}。$$

综上所述, 可得:

$$\{a \in A | x \downarrow a\} = (\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*) \cap p(x)$$

在形式背景中, 计算出背景的所有对象概念和属性概念后, 根据定理 5, 对一个给定的对象, 可以很方便地找出满足箭头关系的属性集合; 对一个指定的属性, 亦可方便地找出满足箭头关系的对象集合, 故相对于定义 4, 定理 5 的效率较高。对比定理 5 与定理 4 可以看出, 定理 5 不需要计算背景的所有概念, 而只需要计算所有对象概念和属性概念。对于一个背景来说, 计算对象概念和属性概念是容易的, 定理 5 可以用作设计快速实现查找箭头关系算法的理论依据。

在定理 5 中, 对 $\forall x \in U, \forall a \in A, x^{**}$ 和 a^* 已知时, 计算

$\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*$ 需要进行 $|A - x^*|$ 次“ $*$ ”运算。在设计箭头关系查找算法时, 为了进一步减少计算 $\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*$ 的复杂性, 根据定理 2, 可以得到推论 1。利用此推论, 计算 $\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*$ 仅需一次“ $*$ ”运算。

推论 1 设 (U, A, I) 是一个背景, 则下列命题成立:

(1) 对任意 $x \in U$, 有

$$\{a \in A | x \downarrow a\} = (x^{**} \cap (\bigcup_{a \in p(x)} a^*))^* \cap p(x)$$

其中 $p(x) = A - x^*$;

(2) 对任意 $a \in A$, 有

$$\{x \in U | x \uparrow a\} = (a^{**} \cap (\bigcup_{x \in q(a)} x^*))^* \cap q(a)$$

其中 $q(a) = U - a^*$ 。

证明: 这里仅证明式(1), 式(2)可以类似证明。

根据定理 2, 有

$$\bigcap_{a \in p(x)} (x^{**} \cap a^*)^* = (\bigcup_{a \in p(x)} (x^{**} \cap a^*))^* = (x^{**} \cap (\bigcup_{a \in p(x)} a^*))^*$$

再结合定理 5, 可得

$$\begin{aligned} \{a \in A | x \downarrow a\} &= (\bigcap_{a \in p(x)} (x^{**} \cap a^*)^*) \cap p(x) \\ &= (x^{**} \cap (\bigcup_{a \in p(x)} a^*))^* \cap p(x) \end{aligned}$$

4 快速查找形式背景的箭头关系的算法

在推论 1 的基础上, 对一个给定的形式背景, 提出如下箭头关系快速查找算法。

算法 1 快速查找形式背景的箭头关系

输入: 一个背景 (U, A, I)

输出: 带有箭头关系标号的背景

- (1) 对于 U 中的每个对象 x_i , 令 $UI(i) \leftarrow x_i^*, UE(i) \leftarrow (UI(i))^*$; //
- $UI(i), UE(i)$ 分别存储对象概念 γx_i 的内涵和外延//
- (2) 对于 A 中的每个属性 a_j , 令 $AE(j) \leftarrow a_j^*, AI(j) \leftarrow (AE(j))^*$; //
- $AE(j), AI(j)$ 分别存储属性概念 μa_j 的外延和内涵//
- (3) 令 $Temp1 \leftarrow U$;
- (4) 若 $Temp1 = \emptyset$, 则转(9), 否则任取 $Temp1$ 中的一个对象 x_i , 执行 $p(i) \leftarrow A - UI(i)$;
- (5) 若 $p(i) = \emptyset$, 则转(8); 否则执行 $E_temp \leftarrow UE(i) \cap (\bigcup_{a_j \in p(i)} AE(j))$;
- (6) 令 $Itemp \leftarrow (E_temp)^*, P(i) \leftarrow Intemp \cap p(i)$;
- (7) 在背景 (U, A, I) 中, 把 x_i 所在行与 $P(i)$ 中的属性所在列的交叉处标记为“ $\downarrow$ ”;
- (8) 令 $Temp1 \leftarrow Temp1 - \{x_i\}$, 并返回(4);
- (9) 令 $Temp2 \leftarrow A$;
- (10) 若 $Temp2 = \emptyset$, 则转(15), 否则任取 $Temp2$ 中的一个属性 a_j , 执行 $q(j) \leftarrow U - AE(j)$;
- (11) 若 $q(j) = \emptyset$, 则转(14); 否则执行 $I_temp \leftarrow AI(j) \cap (\bigcup_{x_i \in q(j)} UI(i))$;
- (12) 令 $Etemp \leftarrow (I_temp)^*, Q(j) \leftarrow Etemp \cap q(j)$;
- (13) 检查属性 a_j 所在列与 $Q(j)$ 中的对象所在行的交叉处是否已标记有“ $\downarrow$ ”, 如果是则将其修改为“ $\uparrow$ ”; 否则标记为“ $\uparrow$ ”;
- (14) 令 $Temp2 \leftarrow Temp2 - \{a_j\}$, 并返回(10);
- (15) 结束算法, 输出带有箭头关系的背景。

下面分析算法 1 的时间复杂度, 这里仅考虑最坏情况下的时间复杂度。

根据上一节的讨论可知, 执行步骤(1)和执行步骤(2)的

时间复杂度分别为 $O(|U|^2|A|)$ 和 $O(|U||A|^2)$ 。在步骤(4)中,差运算需 $O(|A|)$ 时间;步骤(5)中,交、并运算的时间复杂度为 $O(|U||A|)$;步骤(6)中,“*”运算的时间复杂度为 $O(|U||A|)$,交运算需 $O(|A|)$ 时间;步骤(7)中,执行标记任务的时间复杂度为 $O(|A|)$;步骤(8)中,差运算需要 $O(|U|)$ 时间。因此,步骤(3)~(8)的时间复杂度为 $O(|U|^2|A|)$ 。类似地,执行步骤(9)~(14)的时间复杂度为 $O(|U||A|^2)$ 。

综上可知,算法1的时间复杂度为:

$$O(|U|^2|A|) + O(|U||A|^2) + O(|U|^2|A|) + O(|U||A|^2) = O((|U| + |A|)|U||A|)$$

由于基于定义4查找箭头关系的时间复杂度为 $O(|U|^2|A|^2)$,因此从理论上讲,算法1能够有效地降低查找形式背景的箭头关系的计算复杂性。

下面给出一个实例来说明算法1的有效性与可行性。

例1 表1^[8]给出了一个形式背景 (U, A, I) ,其中对象集为 $U = \{x_1, x_2, x_3, x_4, x_5\}$,属性集为 $A = \{a_1, a_2, a_3, a_4\}$, i 行 j 列的交叉处,1表示对象 x_i 具有属性 a_j ,0表示对象 x_i 不具有属性 a_j 。求该背景的箭头关系。

表1 一个形式背景

	a_1	a_2	a_3	a_4
x_1	1	1	1	1
x_2	1	1	0	0
x_3	0	1	1	1
x_4	0	1	0	0
x_5	0	1	1	0

为便于计算,下面叙述中,用 $x_1x_2x_3x_4x_5$ 表示 $\{x_1, x_2, x_3, x_4, x_5\}$,用 $a_1a_2a_3a_4$ 表示 $\{a_1, a_2, a_3, a_4\}$,其他对象集、属性集可以类似表示。

对于表1所列的背景,文献[9]中已经给出利用箭头关系的定义查找箭头关系的过程。下面利用本文提出的箭头关系快速查找算法(算法1)查找表1所列背景中满足箭头关系的对象-属性对,并在表中用 $\downarrow$ 、 $\uparrow$ 或 $\updownarrow$ 标出。

(1)对于 U 中的每个对象 $x_i (i \leq 5)$,计算 $UI(i)$ 和 $UE(i)$,可得:

$$UI(1) = x_1^* = a_1a_2a_3a_4, UE(1) = x_1;$$

$$UI(2) = x_2^* = a_1a_2, UE(2) = x_1x_2;$$

$$UI(3) = x_3^* = a_2a_3a_4, UE(3) = x_1x_3;$$

$$UI(4) = x_4^* = a_2, UE(4) = x_1x_2x_3x_4x_5;$$

$$UI(5) = x_5^* = a_2a_3, UE(5) = x_1x_3x_5。$$

(2)对于 A 中的每个属性 $a_j (j \leq 4)$,计算 $AE(j)$ 和 $AI(j)$,可得:

$$AE(1) = a_1^* = x_1x_2, AI(1) = a_1a_2;$$

$$AE(2) = a_2^* = x_1x_2x_3x_4x_5, AI(2) = a_2;$$

$$AE(3) = a_3^* = x_1x_3x_5, AI(3) = a_2a_3;$$

$$AE(4) = a_4^* = x_1x_3, AI(4) = a_2a_3a_4。$$

(3)令 $Temp1 \leftarrow U$;

(4)由于 $Temp1 = \emptyset$ 不成立,取 $x_1 \in Temp1$,计算 $p(1) = A - UI(1) = \emptyset$ 。

(5)由于 $p(1) = \emptyset$,则 $Temp1 = Temp1 - \{x_1\} = x_2x_3x_4x_5$ 。

(6)由于 $Temp1 = \emptyset$ 不成立,取 $x_2 \in Temp1$,计算 $p(2)$

$= a_3a_4$ 。

(7)由于 $p(2) = \emptyset$ 不成立,执行

$$E_temp = UE(2) \cap (AE(3) \cup AE(4)) = x_1。$$

$$(8) Itemp = x_1^* = a_1a_2a_3a_4,$$

$$P(2) = Itemp \cap p(2) = a_3a_4。$$

(9)在背景 (U, A, I) 中,在第2行与第3列和第4列的交叉处分别标记 $\downarrow$,如表2所列。

重复算法1中的步骤(4)~(8),直至 $Temp1 = \emptyset$ 成立,此时可以得到如表2所列的形式背景。

表2 背景中的“ $\downarrow$ ”关系

	a_1	a_2	a_3	a_4
x_1	1	1	1	1
x_2	1	1	$\downarrow$	$\downarrow$
x_3	$\downarrow$	1	1	1
x_4	0	1	0	0
x_5	0	1	1	$\downarrow$

(10)令 $Temp2 \leftarrow A$;

(11)由于 $Temp2 = \emptyset$ 不成立,取 $a_1 \in Temp2$,计算 $q(1) = U - AE(1) = x_3x_4x_5$ 。

(12)由于 $q(1) = \emptyset$ 不成立,执行 $I_temp = AI(1) \cap (UI(3) \cup UI(4) \cup UI(5)) = a_2$ 。

(13) $Etemp = a_2^* = x_1x_2x_3x_4x_5, Q(1) = Etemp \cap q(1) = x_3x_4x_5$ 。

(14)在表2所列的形式背景中,由于第1列第3行交叉处已存在 $\downarrow$,因此修改标号为 $\updownarrow$;第1列第4行和第5行的交叉处都没有 $\downarrow$,则分别标记 $\uparrow$,如表3所列。

表3 背景中的箭头关系

	a_1	a_2	a_3	a_4
x_1	1	1	1	1
x_2	1	1	$\updownarrow$	$\downarrow$
x_3	$\updownarrow$	1	1	1
x_4	$\uparrow$	1	$\uparrow$	0
x_5	$\uparrow$	1	1	$\updownarrow$

重复算法1中的步骤(10)~(14),直至 $Temp2 = \emptyset$ 时算法终止。此时可以得到最终的具有箭头关系标号的背景(见表3)。

5 数值实验

上一节已经从理论上证明了“算法1较基于传统定义查找箭头关系的方法更能够有效地降低计算复杂性”。这一节通过一些数值实验对有效降低查找箭头关系的时间复杂性的程度作进一步评估,为此需要选取各种规模的形式背景进行查找箭头关系的仿真。对于较小的形式背景,该文选取文献[8]中的“Live in water”和文献[13]中的“Lattice properties”。为了获取较大的形式背景,该文选择 UCI 知识库中的 Wine、Tic-Tac-Toe Endgame 和 Car Evaluation 数据集。由于这3个数据集中的属性取值都为多值或连续值,即原始背景都为多值背景,因此需要事先将这3个多值背景转换成单值背景。文献[8]中给出了利用标尺把多值背景转换成单值背景的方法(即各种转换标尺),FCA Data Repository^[14]公布了对以上3个 UCI 数据集进行转换的最终结果。实验所需的形式背景的详细情况如表4所列。

表4 数值实验所需的形式背景

数据库编号	形式对象个数	形式属性个数	形式概念个数	数据来源
Database 1	8	9	19	"Live in water" ^[8]
Database 2	14	16	24	"Lattice properties" ^[13]
Database 3	178	68	25409	Wine Data Set ^[14]
Database 4	958	29	59505	Tic-Tac-Toe Endgame Data Set ^[14]
Database 5	1728	25	12640	Car Evaluation Data Set ^[14]

使用 MATLAB 进行编程,对于每一个形式背景都分别采用基于箭头关系定义的箭头关系查找方法和算法 1(箭头关系快速查找算法)查找箭头关系,并记录运行时间,具体运行时间如表 5 所列。

表5 两种方法运行时间对比

数据库编号	形式对象个数	形式属性个数	基于定义查找运行时间(s)	算法 1 运行时间(s)
Database 1	8	9	0.0478	0.0064
Database 2	14	16	0.2031	0.0156
Database 3	178	68	178.1255	0.8449
Database 4	958	29	1508.1537	3.0780
Database 5	1728	25	4538.0611	6.4454

从表 5 可以看出,当形式背景较大时,本文提出的算法 1 在计算效率上显著高于基于定义查找箭头关系的方法,说明算法 1 可以有效地降低背景中查找箭头关系所需的计算量。

结束语 在形式概念分析中,箭头关系在约简形式背景、构造兼容子背景等方面有着重要的作用,因此如何快速确定出一个形式背景的箭头关系是一个有意义的课题。为此,本文提出了一个快速确定形式背景的箭头关系的算法,并通过实例与数值实验表明了它的可行性与高效性。由于箭头关系的基础性作用,本文提出的快速算法实际上也为形式背景的约简、子背景的构建以及双建立背景的研究等提供了计算上的支持。

参考文献

[1] Wille R. Restructuring lattices theory: an approach based on hi-

erarchies of concepts [C]//Rival I, ed. Ordered Sets. Dordrecht-Boston:Reidel,1982:445-470

- [2] Young P. Software retrieval by samples using concept analysis [J]. The Journal of Systems and Software, 2000, 54: 179-183
- [3] 张文修,魏玲,祁建军. 概念格的属性约简理论与方法[J]. 中国科学 E 辑,信息科学, 2005, 35(6): 628-639
- [4] 仇国芳,朱朝晖. 基于经典-模糊变精度概念格的决策规则获取及其推理算法[J]. 计算机科学, 2009, 36(12): 216-218
- [5] 吕跃进,李金海. 基于一一映射的概念格属性约简算法[J]. 计算机应用研究, 2009, 26(3): 849-851
- [6] 吕跃进,李金海. 概念格属性约简的启发式算法[J]. 计算机工程与应用, 2009, 45(2): 154-157
- [7] Wille R. Subdirect decomposition of concept lattices [J]. Algebra Universalis, 1983, 17: 275-287
- [8] Ganter B, Wille R. Formal Concept Analysis [M]. Mathematical Foundations, Berlin: Springer, 1999
- [9] Belohlavek R. Introduction to formal concept analysis [EB/OL]. <http://belohlavek.inf.upol.cz/vyuka/IntroFCA.pdf>, 2011-10-16
- [10] Serhiy Y. System of data analysis "Concept Explorer" [C]// Proceedings of the 7th national conference on Artificial Intelligence KII-2000. 2000: 127-134
- [11] Geyer W. The generalized doubling construction and formal concept analysis [J]. Algebra Universalis, 1994, 32: 341-367
- [12] Stephan D. A Context-Based Description of the Doubly Founded Concept Lattices in the Variety Generated by M3 [C]// Proceedings of the 9th international conference on Formal concept analysis (ICFCA'11). Berlin, Heidelberg: Springer-Verlag, 2011: 93-106
- [13] Wille R. Concept Lattices and Conceptual Knowledge Systems [J]. Computers Mathematics and Applications, 1992, 23(6-9): 493-515
- [14] FCA Data Repository [EB/OL]. <http://fcarepository.com>, 2011-10-20

(上接第 161 页)

- [9] The hadoop fair scheduler [EB/OL]. <https://issues.apache.org/jira/browse/HADOOP-3746>
- [10] Capacity Scheduler for Hadoop [EB/OL]. http://Hadoop.apache.org/common/docs/current/Capacity_scheduler.html, 2010-03-22
- [11] Kamal K, Kemafor A. Scheduling hadoop jobs to meet deadlines [C]// Proceedings of 2nd IEEE International Conference on Cloud Computing Technology and Science. US, 2010: 388-392
- [12] Tian Feng-guang, Chen Ke-ke. Towards optimal resource provisioning for Running MapReduce programs in public clouds [C]// Proceedings of 2011 IEEE International Conference on Cloud Computing. US, 2011: 155-162
- [13] Thomas S, Kevin L. MapReduce optimization using regulated dynamic prioritization [C]// Proceedings of the Eleventh International Joint Conference on Measurement and Modeling of Computer Systems. US, 2009: 299-310
- [14] Joel W, Deepak R, Kirsten H, et al. FLEX: A slot allocation

scheduling optimizer for MapReduce workloads [C]// Proceedings of International Middleware Conference. Germany, 2010: 1-20

- [15] Michael I, Vijayan P, Jon c, et al. Quincy: Fair scheduling for distributed computing clusters [C]// Proceedings of the 22nd ACM SIGOPS Symposium on Operating Systems Principles. US, 2009: 261-276
- [16] Jia Yu, Buyya R, Ramamohanarao K. Workflow Scheduling Algorithms for Grid Computing [R]. The University of Melbourne, Australia, 2007
- [17] Herald K, Eva S, Manolis MT. Schedule optimization for data processing flows on the cloud [C]// Proceedings of the ACM SIGMOD International Conference on Management of Data. US, 2011: 289-300
- [18] Yu Zhi-feng, Shi Wei-song. A planner-guided scheduling strategy for multiple workflow applications [C]// Proceedings of the International Conference on Parallel Processing Workshops. US, 2008: 1-8