基于元信息的最小决策规则集获取方法①

A meta – information based method for minimal rule set extraction

钟晴江 苏 健 (浙江大学城市学院计算机应用技术研究所 杭州 310015)

摘要:决策规则获取是粗糙集理论的一个重要研究领域,并出现了大量的方法。但是这些方法大都假定数据是集中式存储的。在分布式数据环境中,代价昂贵的数据集成工作是必不可少的。元信息是信息系统及其子系统的紧凑描述,并且各个信息子系统元信息的集成代价远小于原始数据的集成。本文提出基于元信息的最小规则集获取方法。该方法的时间复杂度远小于传统的 LEM2 算法。

关键词:规则获取 粗糙集 元信息

1 引言

决策规则获取是粗糙集理论的一个重要研究领域。在规则获取过程中,我们往往希望获取最小数量的决策规则,即所谓的最小决策规则集。LERS(Learning from Examples based on Rough Sets)是著名的基于粗糙集的规则获取系统之一,它的主要作者是 Grzymala – Busse 教授。LERS 系统有四种不同的规则获取方式,其中最著名的算法是 LEM2。

但是传统的规则获取方法都是基于一种假设,即所有数据是集中式存放的,并不能直接适用于分布式数据环境。然而数据的分布式存储是更加普遍的形式,因此代价昂贵的数据集成工作是必不可少的。为此,我们提出元信息的概念。元信息是信息系统的一种紧凑描述,其存储代价和集成代价都远小于相应的原始数据,因此我们提出以元信息的集成取代代价高昂的原始数据集成,并在元信息基础上改造传统的粗糙集方法。

本文提出的最小决策规则集获取方法以元信息为基础,能够从各个分布式信息子系统集成得到的元信息中获得最小决策规则集,并且代价将远小于直接从原始数据中获取相应规则。

2 信息系统及其元信息

定义 1:信息系统 $S = <U, A>$[1,2],其中:

(1) $U = \{u_1, u_2, \cdots, u_n\}$, $n = |U|$,且 $n > 0$, U 是对象集合;

(2) $A = C \cup D$. A 是属性集合,C 是条件属性集合,D 是决策属性集合。

(3) 对每个属性 $a \in A$,存在从 U 到某一空间的映射 a, $a: U \to V_a$,其中 V_a 被称为属性 a 的值域。

定义 2:信息系统 $S = <U, A>$, $B \subseteq A$ 在 U 上的不分明关系 IND(B)定义为[1,2]:

$$IND(B) = \{(x,y) \in U^2 \mid (\forall a)(a \in B \to a(x) = a(y))\}$$

不分明关系 IND(B)可简写为 B。若 x,y 存在不分明关系 IND(B),即 $(x,y) \in IND(B)$,则可记为 xBy。显然,IND(B)是一个等价关系。根据 IND(B)可以导出一个等价划分 U/B,记 $B^* = U/B$。相应的,条件属性集 C 对应的等价类族记为 $C^* = U/C$,决策属性集 D 对应的决策类族记为 $D^* = U/D$。并称 $X \in C^*$ 为条件类,称 $Y \in D^*$ 为决策类。

定义 3:对等价类 $X \in B^*$,称 $(x, |X|)$ 为等价类 X 的等价类描述,记为 $D_x = (x, |X|)$,其中 $|X|$ 是 X 的基数,$x \in X$ 称为特征元素。

等价类描述简称为类描述。由于经常将 B^* 中的等价类记为 $[x]_B$,所以等价类 $[x]_B$ 的类描述经常记为 $(x, |[x]_B|)$。由于类描述采用一个特征元素和一个数字表示某一等价类,因此类描述所需的存储空间显

① 国家自然科学基金(No. 60473097),973 课题(No. 2003CB317005)资助

然比等价类本身小得多。

属性集合 $B\subseteq A$ 对应类族 $B*$，那么称 $F_B=\{(x,|[x]_B|):[x]_B\in B*\}$ 为(B 属性上的)类描述集合。相应地，$F_C=\{(x,|[x]_C|):[x]_C\in C*\}$ 被称为条件类描述集合，而 $F_D=\{(y,|[y]_D|):[y]_D\in D*\}$ 被称为决策类描述集合。

定义 4：信息系统 $S=<U,A>$，S 的元信息 I_{CD} 是一个三元组，$I_{CD}=<M_{CD},F_C,F_D>$，其中：

(1) $F_C=\{(x_1,|[x_1]_C|),(x_2,|[x_2]_C|),\cdots,(x_m,|[x_m]_C|)\}$，其中 $m=|C*|$，即 m 是条件类的个数。F_C 称为条件类描述集合。

(2) $F_D=\{(y_1,|[y_1]_D|),(y_2,|[y_2]_D|),\cdots,(y_n,|[y_n]_D|)\}$，其中 $n=|D*|$，即 n 是决策类的个数。F_D 称为决策类描述集合。

(3) $M_{CD}=(r_{ij})$，其中 $r_{ij}=|[x_i]_C\cap[y_j]_D|$，$1\leqslant i\leqslant m$，$1\leqslant j\leqslant n$。$M_{CD}$ 称为等价类矩阵，简称为类矩阵。

显然，类矩阵 M_{CD} 是 $m\times n$ 矩阵。类矩阵用来衡量信息系统中条件类与决策类交集的大小。与条件类与决策类的交集 $[x_i]_C\cap[y_j]_D$ 相比，交集大小 $|[x_i]_C\cap[y_j]_D|$ 只是一个数字，显然所需的存储空间要少的多。元信息 I_{CD} 一般简写为 $I=<M,F_C,F_D>$。

3 粗糙决策规则

定义 5：决策规则具有以下形式：

(1) $R\Rightarrow Q$；或者(2) if R then Q。

其中，R 是规则的条件部分，Q 是规则的决策部分。条件部分 R 是条件值对的合取. 如果决策规则是确定性的，那么决策部分 Q 将是某一单一的基本决策。如果决策规则是非确定性的，那么决策部分 Q 将是多个基本决策的析取。本文讨论决策集属性个数为 1 的情况。令决策属性集合 $D=\{d\}$，那么确定性决策规则的一般形式是

$$(c_1,v_1)\wedge(c_2,v_2)\wedge(c_3,v_3)\wedge\cdots\wedge(c_q,v_q)\Rightarrow(d,v),$$

而非确定性决策规则的一般形式是

$$(c_1,v_1)\wedge(c_2,v_2)\wedge(c_3,v_3)\wedge\cdots\wedge(c_q,v_q)\Rightarrow(d,I_1)\vee(d,I_2)\vee(d,I_3)\vee\cdots\vee(d,I_p),$$

基本条件 $c=(a,v)$ 可以看作是映射 $c:U\rightarrow\{true,false\}$。对 $x\in U$，如果 $c(x)=true$，那么说明对象 x 在属性 a 上的值为 v。记 $T=c_1\wedge c_2\wedge c_3\wedge\cdots\wedge c_q$，T 是 q 个基本条件的合取式。那么 $T(x)=ture$ 表示 $c_1(x)$，$c_2(x)$，$c_3(x)$，$\cdots$，$c_q(x)$ 都为真。

定义 6：令 T 是基本条件的合取式，那么 T 的覆盖 [T] 定义为[3-5]：

$[T]=\{x\in U\mid T(x)=true\}$，

令 K 为决策概念，那么正覆盖 $[T]_K^+$ 定义为：

$[T]_K^+=[T]\cap K$，

负覆盖 $[T]_K^-$ 定义为：

$[T]_K^-=[T]\cap(U-K)$。

T 的覆盖 [T] 表示所有为 T 所描述对象的集合。正覆盖 $[T]_K^+$ 是覆盖 [T] 中属于 K 的正例的集合，而负覆盖 $[T]_K^-$ 是覆盖 [T] 中不属于 K 的负例的集合。显然，覆盖中的对象或者属于正覆盖，或者属于负覆盖。

定义 7：令 R 是基本条件的合取式，K 为决策概念。如果 $[R]_K^-=\varnothing$，那么存在决策规则

$R\Rightarrow Q$

其中，Q 是描述决策概念 K 的基本决策的析取式。

上述定义所给出的决策规则能正确区分正例和负例。$[R]_K^-=\varnothing$ 说明 $[R]\subseteq K$，即覆盖 [T] 中没有不属于 K 的负例。这时就认为存在一条以 R 为条件的描述决策概念 K 的规则。决策规则是否是确定性的取决于描述决策概念的基本决策是否唯一，如果决策概念由单个基本决策描述，那么决策规则是确定性的，如果决策概念由多个基本决策的析取式描述，那么决策规则就是非确定性的。

定义 8：令 r 为决策概念 K 中提取的决策规则，R 为决策规则 r 的条件部分。令 $R=c_1\wedge c_2\wedge c_3\wedge\cdots\wedge c_q$。如果满足下面两个条件，那么称 R 为最小的。

(1) $[R]_K^-=\varnothing$；

(2) $[R']_K^-\neq\varnothing$，其中 R' 为 R 删除任意一个基本条件 $c_i(i=1,2,3,\cdots,q)$ 后的基本条件合取式。

决策规则的条件部分最小的含义是，它的条件部分已经删除了所有冗余的基本条件，而剩余的基本条件对于区分它所对应的决策概念 K 与其它决策概念是必要的。

正覆盖与负覆盖也可以表示为条件类描述集合。T 的正覆盖 $[T]_K^+=[T]\cap K$。[T] 的条件类描述集合是 F(T)，而 K 的条件类描述集合是 D(K)，因此正覆盖 $[T]_K^+$ 的条件类描述集合表示形式为 $F(T)\cap D(K)$。

令 U 为信息系统中全体对象集合。U 的对象子集

U－K 也必然对应若干个条件类，这些条件类对应的条件类描述集合是 $F_C - D(K)$，即 F_C 删除属于 D(K) 的条件类描述后剩余的条件类描述集合。T 的负覆盖 $[T]_K^- = [T] \cap (U-K)$。同样，负覆盖 $[T]_K^-$ 可以通过 F(T) 和 D(K) 表示。负覆盖 $[T]_K^-$ 的条件类描述集合表示形式为 $F(T) \cap (F_C - D(K))$。

命题 1：令 T 为基本条件的合取式，K 为决策概念。

(1) $F(T) \subseteq D(K) \Leftrightarrow [T] \subseteq K$。

(2) $F(T) \cap D(K) \neq \varnothing \Leftrightarrow [T]_K^+ \neq \varnothing$。

(3) $F(T) \cap (F_C - D(K)) = \varnothing \Leftrightarrow [T]_K^- = \varnothing$。

(4) $F(T) \subseteq D(K) \Leftrightarrow [T]_K^- = \varnothing$。

证略。

4　粗糙决策规则获取方法

定义 9：令 Φ 为描述决策概念 K 的一组决策规则的集合，如果 Φ 满足下面三个条件，那么称之为最小决策规则集[3-5]。

(1) 任意决策规则 $r \in \Phi$ 的条件部分 R 都是最小的。

(2) $\cup_{r \in \Phi}[R] = K$。

(3) 对任意的 $r' \in \Phi$，令 $\Phi' = \Phi - r'$，那么 Φ′ 都不满足上述的(1)和(2)。

显然，判断决策规则集合 Φ 是否为最小决策规则集，首先需要判断其中每个规则的条件部分是否都是最小的。命题 1 已经给出了通过元信息判断决策规则的条件是否最小的途径。将 K，[R] 分别表示为条件类描述集 D(K)、F(R) 的形式，判断 R 是否最小只需判断是否 $F(R) \subseteq D(K)$ 成立，且对任意删除某一基本条件的 R′，$F(R') \subseteq D(K)$ 不成立。判断 $\cup_{r \in \Phi}[R] = K$ 也可以用相似的方法。下面先给出命题 2。

命题 2：$\cup_{r \in \Phi}[R] = K \Leftrightarrow \cup_{r \in \Phi}F(R) = D(K)$

证略。

命题 2：说明，$\cup_{r \in \Phi}[R] = K$ 和 $\cup_{r \in \Phi}F(R) = D(K)$ 是等价的。因此，我们只需通过判断 $\cup_{r \in \Phi}F(R) = D(K)$ 是否为真，即可判断 $\cup_{r \in \Phi}[R] = K$ 是否成立。

总之，通过元信息的条件类描述集合，能够判断一个决策规则集合是否为最小规则集。这为基于元信息的最小决策规则集的获取提供了理论依据。

下面将给出基于元信息的最小决策规则集获取算法。本算法的基本思想与 Grzymala 的 LEM2 一致。

算法 1 基于元信息的最小决策规则集获取算法

输入：D(K)。输出：Φ。

(1) $G = D(K)$，$\Phi = \varnothing$。

(2) $R = \varnothing$。$R(G) = \{c : \exists d(d \in G \wedge d = (x, |[x]_C|) \wedge c(x) = \text{true})\}$。

(3) 通过以下各步选择 R(G) 中某一基本条件 c。

① 对所有的 $c \in R(G)$，求 $Z(c) = \{(x, |[x]_C|) \in G : c(x) = \text{true}\}$；令 $Z(c) = \{(x_1, |[x_1]_C|), (x_2, |[x_2]_C|), \cdots, (x_t, |[x_t]_C|)\}$，$t = |Z(c)|$。并对 Z(c) 求 $z(c) = \sum_{i=1}^{t}|[x_i]_C|$。

② 选择 z(c) 值最大的 c。如果 z(c) 值最大的 c 多于一个，那么对它们求 $T(c) = \{(x, |[x]_C|) \in F_C : c(x) = \text{true}\}$；令 $T(c) = \{(x_1, |[x_1]_C|), (x_2, |[x_2]_C|), \cdots, (x_t, |[x_t]_C|)\}$，$t = |T(c)|$ 并对 T(c) 求 $t(c) = \sum_{i=1}^{t}|[x_i]_C|$，并选择 t(c) 最小的 c。

③ 如果 t(c) 最小的 c 也多于一个，那么任意取其中一个。

(4) $R = R \cup \{c\}$，$F(R) = \{(x, |[x]_C|) \in F_C : R(x) = \text{true}\}$，如果 $F(R) \subseteq D(K)$，那么转 6。

(5) $G = Z(c)$，$R(G) = \{c : \exists d(d \in G \wedge d = (x, |[x]_C|) \wedge c(x) = \text{true})\}$，$R(G) = R(G) - R$。转 3。

(6) 保证 R 是最小的，进行以下循环：

① 任取 $c \in R$，$R' = R - \{c\}$，$F(R') = \{(x, |[x]_C|) \in F_C : R'(x) = \text{true}\}$。如果 $F(R') \subseteq D(K)$，那么 $R = R'$。

② 重复(6a)，直到对所有的 $c \in R$，$F(R - \{c\}) \subseteq D(K)$ 都不成立。

(7) $\Phi = \Phi \cup \{r\}$，其中 r 是以 R 中基本条件合取式为条件部分来描述决策概念 K 的决策规则。$G = D(K) - \cup_{s \in \Phi}F(S)$，其中 S 是决策规则 $s \in \Phi$ 的条件部分。如果 $G \neq \varnothing$，那么转步 3。

(8) 保证 Φ 是最小决策规则集，进行以下循环：

① 任取 $s \in \Phi$，$\Phi' = \Phi' - \{s\}$，如果 $\cup_{s \in \Phi}F(S) = D(K)$，那么 $\Phi = \Phi'$。

② 重复(8a)，直到对所有的 $s \in \Phi$，$\cup_{s \in \Phi}F(S) = D(K)$ 都不成立。

(9) 算法结束。

上述算法的时间复杂度取决于步 3 到步 7 的时间复杂度，即 $O(|D(K)|^2 \times |F_C| \times |C|^2)$。

算法 1 的基本思路与 LEM2 相同，算法结构都是类似的。与 LEM2 相比，算法 1 在处理方式上没有较大优

势,但是在计算量上有较大优势。因为算法 1 处理的是条件类描述,而不是对象集合。由于每个条件类描述代表了大量的对象,因而大大降低了处理量,从而减少了计算量。算法 1 在步(3a)的时间复杂度是 O(|G|);而 LEM2 在对应步(3a)的步骤求各个基本条件 c 的|[c]∩G'|,其中 G'是决策概念的某个对象子集,即对应 G 中的所有条件类描述所对应的对象集合,显然需要 G'中搜索满足 c(x)为真的对象 x,因此其计算时间复杂度是 O(|G'|)。一般情况下,|G|比|G'|小得多。所以算法 1 在步(3a)的计算量小得多。对于步 3 的其它步骤,也存在类似的计算量的优势。由于算法 1 和 LEM2 的主要计算量表现在步 3 的循环执行上,所以算法 1 在计算量上有较大优势。

5 结论

元信息是信息系统的紧凑描述。在分布式数据环境中,元信息的集成代价远小于原始数据的集成。因此在元信息上研究各种粗糙集方法是有重要意义的。分析表明,本文提出的基于元信息的最小决策规则集获取方法在计算代价上优于传统的 LEM2 方法。

在元信息基础上,大量传统的粗糙集方法都可以被改造,这方面的研究工作将另外发表。

参考文献

1 Pawlak Z. Rough set: Theoretical Aspects of Reasoning About Data. Kluwer Academic, Dordrecht, The Netherlands, 1991.

2 Komorowski J., Pawlak Z., Polkowski L., et al., Rough sets: A Tutorial. In: Pal S. K., Skowron A., ed., Rough fuzzy hybridization. A new trend in decision-making. Springer, 1999, 3-98.

3 Stefanowski J., On rough set based approaches to induction of decision rules. Skowron W. A., Polkowski L. (ed.), Rough Sets in Knowledge Discovery Vol 1, Physica Verlag, Heidelberg, 1998, 500-529.

4 Grzymala-Busse J. W., A new version of the rule induction system LERS, Fundamenta Informaticae, 1997, 31:27-39.