The Partial Completion and Reduction of Incomplete Decision Tables

Chen Wu

School of Computer Science and Engineering,
Jiangsu University of Science and Technology,
Zhenjiang City, 212003, Jiangsu Province, P.R.China

Abstract

An approach to partly erase null values in incomplete decision systems is proposed. A formally similar reduction approach to incomplete decision table is introduced, compared to the case in complete system, but it is rather different in forming the positive region which is not defined in [9] and the reduction approach which is not the same as in [9].

1 Introduction

Rough Set Theory (RST)[1-3] is put forward by Palwak Z. in 1980s. It ,as a mathematical tool to deal with non-determination, fuzziness, and uncertainty, has been massively used in the research fields of Artificial Intelligence, Data mining[4,5], Pattern Recognition[6], Knowledge Acquisition[7], Machine Learning[8] and so on. RST proposed originally by Palwak Z. is based on the assumption that all objects have definitive values in every attribute in a complete decision system and the classification is made by an indiscernibility relation defined by Pawlak Z. But in incomplete decision systems, it is not always be able to establish an indiscernibility relation due to the existence of null values. So the original RST can no longer be immediately applied in incomplete decision systems which could be seen everywhere in the real world. Therefore many new expanded RST models are suggested, mainly in expanding indiscernibility relation to non-indiscernibility relation such as tolerance relation [9], similarity relation [10], limited tolerance relation [11], etc.

For the reason of decreasing some null values, a partial completion method to fill in incomplete decision systems is introduced.

The paper is organized as follows. In section 2, a partial completion approach to full out incomplete decision systems is suggested. Section 3 defines a reduction approach to incomplete decision table, but it is rather different in forms from the definition by Kryszkiewicz, M in [9].

2 Partial Completion to IDT

Definition 1 An incomplete information system(IIS) is a quadruple: S =< U, AT, V, f >, where U is a non-empty finite set of objects and AT is a non-empty finite set of attributes, such that for any attribute a in AT, a: U->Va where Va is called the value set of attribute a. Attribute domain Va may contain a null value, meaning unknown or uncertain, denoted by special symbol "*". V $= \bigcup_{a \in AT} V_a$ represents the value set of all attributes in S. a(x) represents the value of x at attribute a.

In incomplete decision systems, null values are divided into two types, one is existed, and the other is non-existing. In the present paper, the assumption of existing type of null values is considered. Let a∈AT and x∈U. If the value of x at attribute a is null, i.e. a(x) =*, and the highest frequent appearance value, not a null value, at attribute a is unique, we use it to replace a(x)

Manuscript received April 5, 2010
Manuscript revised April 20, 2010

(=*). If the highest frequent appearance value at attribute a is not unique, we do not use any to replace a(x) (=*), because it is not reasonable to randomly apply any one of the highest frequent appearance values to substitute for. So the result obtained may remain incomplete. We call this procedure a partial completion. It differs from some other papers in randomly applying any one of the highest frequent appearance values to substitute for the null.

In the following, we first give some definitions and then show how to perform the partial completion.

Definition 2 Let S is an incomplete decision system, a in AT, Va={a(x): x in U, a(x)<>*}, |Va| denote the cardinal number of Va. c(a)= $\sum_{x\in U, a(x)\neq *} 1$ represents the number of non-null value at attribute a. If a(x)<>*,then c(a(x))= $\sum_{y\in U, a(y)=a(x)} 1$ expresses the appearance number of a(x) at attribute a.

Definition 3 Let S be an incomplete decision system, a∈AT, x, y in U. Then the probability of the same of x and y at attribute a, denoted by pa(x, y) is defined as follows:

$$p_a(x,y)=\begin{cases} 1, & if \quad a(x)=a(y) \\ c(a(x))/|U|, & if \quad a(x)\neq *\wedge a(y)=* \\ c(a(y))/|U|, & if \quad a(x)=*\wedge a(y)\neq * \\ 0, & if \quad a(x)\neq *\wedge a(y)\neq *\wedge a(x)\neq a(y) \end{cases}$$

Obviously, pa(x, y) is symmetric function about x and y, that is, pa(x, y) = pa(y, x). In the above definition, if a(x)<>* and a(y)=*, pa(x, y) can be also defined to be equal to c(a(x))/c(a); if a(x) =* and a(y)<>*, pa(x, y) can be also defined to be equal to c(a(y))/c(a). But computing c(a) consumes more time than computing |U|. So here pa(x, y) takes the above form and is convenient to calculate.

Definition 4 Let S be an incomplete decision system, a in AT, x in U, a(x)=*. The set of most similar values to the value of x at attribute a, denoted by Pa (x), is

$P_a(x)=\{a(y): p_a(x, y)=\max_{z\in U}\{ p_a(x, z): z\diamond x\}$

Definition 5 Let *S* be an incomplete decision system, *a* in *AT*, *x* in *U*, *a*(*x*)=*. If | P_a (*x*)| =1, then we replace *a*(*x*), a null value, with the unique value in P_a (*x*). If | P_a (*x*)|>1, *a*(*x*) remains unchangeable.

By Definition 5, when *a*(*x*) is null, it can be replaced if and only if P_a (*x*) is consisted of only one element. If P_a (*x*) contains several different elements, it is not allowed to be substituted. Although each element in *P*(*x*) can be regarded as a feasible candidate to replace *a*(*x*), a null value, it is not reasonable to select randomly any one to fill in, because such a replacement will extremely influence on the classification about the universe according to the relation among objects. Suppose *y*, *z* in P_a (*x*). If we replace *a*(*x*) with *a*(*y*), not with *a*(*z*), *x* will be indiscernible with *y*, not with *z*. Oppositely, *x* will be indiscernible with *z*, not with *y*. In this case, we are in a bewilder situation. So the best way is to keep *a*(*x*) null so as to get balance between *y* and *z*. If P_a (*x*) contains more elements, persisting on this balance sometimes is more necessary.

Definition 6 Let *S* be an incomplete decision system, $A\subseteq AT$, *x*, *y* in *U*. Then the probability of the same of *x* and *y* at attribute subset *A*, denoted by $p_A(x, y)$ is defined as follows:

$$p_A(x, y)=\sum_{a\in A}\alpha_a p_a(x, y)$$

where $\sum_{a\in A}\alpha_a=1$ and $0<=\alpha_a<=1$. For any *a* in A, α_a can be considered to be the significance of *a* in attribute subset *A*. Because $0<=\alpha_a<=1$, $0<=p_A(x, y)<=1$ also holds.

If there is no decision about the significances of all attributes in *A*, α_a may be equal to 1/|*A*|, the average value.

Definition 7 Let S be an incomplete decision system, $A \subseteq AT$, x, y in U, $a(x)$=*, The set of most similar values to the value of x in attribute subset A , denoted by $P_A(x)$, is

$P_A(x)=\{a(y): p_A(x, y)= \max_{z \in U} \{ p_A(x, z): z <> x\}$.

According to Definition 7, when $a(x)$ is null, it can be replaced if and only if $P_A(x)$ is composed of only one element, otherwise $a(x)$ remains null.

A partial completion to an incomplete decision system may be fulfilled by following the above definitions. We call this procedure a partial completion, because S may still be an incomplete decision system, not a complete one. But after this partial completion, there are not so many null values in S. This consideration is more appropriate to the real world problem solving because when there are several candidates existed, it not suitable to select any one of them to act.

Example 1 Let we have an incomplete decision system $S=<U, AT, V, f>$ shown in table 1, where the universe $U=\{x_1,x_2,\ldots,x_{10}\}$, the attribute set $AT=\{a_1,a_2,a_3\}$,* means null.

Table 1 an incomplete information system

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
a1	1	1	3	1	*	3	3	3	2	3
a2	2	*	2	2	2	1	2	1	3	1
a3	1	3	3	*	1	*	*	2	*	2

In order to discriminate the position of *, we use *ij represents object xi has a null value at attribute aj. P(*ij) states the value set of the values that may be chosen to replace *ij .

According to Definition 5, we can obtain, P(*22)= P_{a_2} (*22)={2}, P(*43)= P_{a_3} (*43)={1,2,3}, P(*51)= P_{a_1} (*51)={3}, P(*63)= P_{a_3} (*63)={1,2,3}, P(*73)= P_{a_3}(*73)={1,2,3}, P(*93)= P_{a_3}(*93) ={1,2,3}. Therefore, we get a partial completion system as in Table 2.

Furthermore, let A=AT={a1,a2,a3}, $\alpha_{a_1} = \alpha_{a_2} = \alpha_{a_3} = 1/3$, PA(*ij) state the value set of the values that may be chosen to replace *ij by attribute subset A, then according to Definition 6, we obtain, PA(*43)={1,3}, PA(*63)={2},PA(*73)={1,3}, PA(*93)={1,2,3}. We get a further partial complete decision system shown in Table 3, from Table 2

Table 2 partial complete IIS from table 1

	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10
a1	1	1	3	1	3	3	3	3	2	3
a2	2	2	2	2	2	1	2	1	3	1
a3	1	3	3	*	1	*	*	2	*	2

Table 3 further partial complete IIS from table 2

	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10
a1	1	1	3	1	3	3	3	3	2	3
a2	2	2	2	2	2	1	2	1	3	1
a3	1	3	3	*	1	2	*	2	*	2

In this way, some null values are deleted except *43, *73, and *93. It is still incomplete.

3 A New Reduction Method to IDT

Although you may do some completion work to your incomplete decision table so as to remove missing data or noises, you may still confront a incomplete one. Therefore, we have a urgent requirement to research approaches to reduce incomplete information system or decision tables. [9] puts forward an efficient method to obtain reductions based on discernibility matrix. But the matrix is much different from that in complete systems in formation, for example, one is symmetric, the another is non-symmetric. Now we give a description of reductions in incomplete decision tables. Most of terms are cited from [9].

Let S =< U, AT, V, f > is as in definition 1. $A \subseteq AT$, the tolerance relation is defined as $SIM(A) = \{(x, y) \in U \times U \mid \forall a \in A, f(x, a)$

$= f(y,a) \vee f(x,a) = * \vee f(y,a) = *\}.$

$S_A(x) = \{y \in U \mid (x,y) \in SIM(A)\}$ is the tolerance class of x. $U / SIM(A) = \{S_A(x) \mid x \in U\}$ is the collection of all tolerance classes with respect to $A \subseteq AT$. It forms a covering, but maybe not a partition of U. $A \subseteq AT$ is a reduction of S if both $SIM(A) = SIM(AT)$ and for any $B \subset S, SIM(B) \neq SIM(AT).$

For any $X \subseteq U$, $\underline{A}(X) = \{x \in U \mid S_A(x) \subseteq X\}$ is called the lower approximation of X, while $\overline{A}(X) = \{x \in U \mid S_A(x) \cap X \neq \varnothing\}$ is called the upper approximation of X. Note that forms of them are very similar to those in complete systems, but the contents are much different. In S =< U, AT, V, f >, if $AT = C \cup D, \quad C \cap D = \varnothing,$ C is called condition attribute set, D decision attribute set. Ordinarily, $* \notin V_d (d \in D)$. For convenience to speaking, D is always supposed to be a single decision attribute set, i.e. $D = \{d\},$ d is the decision attribute. SIM(D) is a equivalence relation because $* \notin V_d (d \in D)$. $pos_A(D) = \cup \underline{A}(Y)(Y \in U / SIM(D))$ is called the positive region of A with respect to D. $\gamma_A(D) = card(pos_A(D)) / card(U)$ is called dependence degree of attribute set $A \subseteq AT$ with respect to D.

If $pos_A(D) = pos_{A-\{a\}}(D)(a \in A)$, then a is called dispensable on A with respect to D. Otherwise a is called indispensable on A with respect to D. If all attributes in A are indispensable, then A is called to be independent with respect to D.

Definition 8 $A \subseteq AT$ is a reduction of AT with respect to D, if and only if A is independent on AT and $pos_A(D) = pos_{AT}(D).$

Theorem 1 $A \subseteq AT$ is a reduction of AT with respect to D, if and only if

$\gamma_A(D) = \gamma_{AT}(D) \wedge (\forall a \in A)(\gamma_{A-\{a\}}(D) \neq \gamma_{AT}(D)).$

Proof Suppose that $A \subseteq AT$ is a reduction of AT with respect to D. Then A is independent on AT and $pos_A(D) = pos_{AT}(D).$ So $\gamma_A(D) = card(pos_A(D)) / card(U)$ $= card(pos_{AT}(D)) / card(U) = \gamma_{AT}(D)$ and for any $a \in A,$ a is indispensable , thus $pos_A(D) \neq pos_{A-\{a\}}(D)(a \in A)$ .Therefore, $\gamma_{A-\{a\}}(D) \neq \gamma_{AT}(D).$ Conversely , if $\gamma_A(D) = \gamma_{AT}(D) \wedge (\forall a \in A)(\gamma_{A-\{a\}}(D) \neq \gamma_{AT}(D)),$ then $pos_A(D) = pos_{AT}(D),$ and $(\forall a \in A) pos_A(D) \neq pos_{A-\{a\}}(D)$ for $(\forall a \in A)(\gamma_{A-\{a\}}(D) \neq \gamma_{AT}(D))$. That is A is a reduction of AT with respect to D. The theorem is finished proving.

Use this method to solve the same example described in [9], we can also get that {P,S,X} is a reduction of the incomplete decision table. The computation process is omitted here so as to save the space.

4. Conclusions

A procedure of partial completion to an incomplete decision system is suggested in section 2. Its aim is to erase a null value of an object which may be replaced by the most frequent value at the same attribute. To some extend, it decreases uncertainties. Although the result system may still be an incomplete one as the procedure supposes to do so, the approach is meaningful for reminding that people .do not need to hurry up in dealing with data rather than preprocessing. In section 3, a formally similar reduction approach to incomplete decision table is introduced, compared to the case in complete system, but it is rather different from the definition by Kryszkiewicz, M in [9]. The next research step ofr us is to develop efficient algorithms to find reductions of incomplete decision tables.

References

[1] Pawlak, Z. Rough sets. International Journal of Computer and Information Sciences. 11(1982) 341~356

[2] Pawlak, Z. Rough set theory and its applications to data analysis. Journal of Cybernetics and Systems. 29 (1998) 661~688

[3] Pawlak, Z. Rough sets and intelligent data analysis. Journal of Information Sciences. 147 (2002) 1~12

[4] Chien-Chung, Chan.: A rough set approach to attribute generalization in data mining. Journal of Information Sciences. 107 (1998) 169~176

[5] Ananthanarayana, V.S. Narasimha Murty, M. Subramanian, D.K.: Tree structure for e±cient data mining using rough sets. Pattern Recognition Letters. 24 (2003) 851~862

[6] Roman, W.Swiniarski. Andrzej, Skowron.: Rough set method in feature selection and recognition. Pattern Recognition Letters. 24 (2003) 833~849

[7] Ju-Sheng Mi,Wei-Zhi Wu,Wen-Xiu Zhang: Approaches to knowledge reduction based on variable precision rough set model. Journal of Information Sciences. 159 (2004) 255~272.

[8] Tzung-Pei, Hong. Li-Huei, Tseng. Shyue-Liang, Wang.: Learning rules from incomplete training examples by rough sets. Expert Systems with Applications. 22 (2002) 285~293

[9] Kryszkiewicz, M.: Rough set approach to incomplete information systems. Journal of Information Sciences. 112 (1998) 39~49

[10] Jerzy, Stefanowski.: Incomplete information tables and rough classification. Journal of Computational Intelligence. 17 (2001) 545~566

[11] Guoyin, Wang.: Extension of rough set under incomplete information system. Journal of Computer Research and Development. 39 (2002) 1238~1243

[12] Y.Y.Yao: Information granulation and rough set approximation. International Journal of Intelligent Systems. 16 87~104

[13] William, Zhu. Fei-Yue, Wang.: Reduction and axiomization of covering generalized rough sets. Journal of Information Sciences. 15 (2003) 217~230

[14] Wanli Chen, Jiaxing Cheng, Chijian Zhang.: A generalization to rough set theory based on tolerance relation, Journal of computer engineering and applications, 16(2004)26~28