An Algorithm for Finding Equivalence Relations from Tables with Non-deterministic Information

Hiroshi SAKAI and Akimichi OKUMA

Department of Computer Engineering
Kyushu Institute of Technology, Tobata, Kitakyushu 804, Japan
sakai@comp.kyutech.ac.jp

Abstract. Rough sets theory depending upon DIS (Deterministic Information System) is now becoming a mathematical foundation of soft computing. Here, we pick up NIS (Non-deterministic Information System) which is more general system than DIS and we try to develop the rough sets theory depending upon NIS. We first give a definition of definability for every object set X, then we propose an algorithm for checking it. To find an adequate equivalence relation from NIS for X is the most important part in this algorithm, which is like a resolution. According to this algorithm, we implemented some programs by prolog language on the workstation.

1 Introduction

Rough sets theory is seen as a mathematical foundation of soft computing, which covers some areas of research in AI, i.e., knowledge, imprecision, vagueness, learning, induction[1,2,3,4]. We recently see many applications of this theory to knowledge discovery and data mining[5,6,7,8,9].

In this paper, we deal with rough sets in NIS (Non-deterministic Information System), which will be an advancement from rough sets in DIS (Deterministic Information System). According to [1,2], we define every DIS = (OB, AT, {VAL\(_a\) | \(a \in AT\)}, f), where OB is a set whose element we call object, AT is a set whose element we call attribute, VAL\(_a\) for \(a \in AT\) is a set whose element we call attribute value and f is a mapping such that \(f: OB \times AT \rightarrow \bigcup_{a \in AT} VAL_{a}\), which we call classification function. For every \(x, y (x \neq y) \in OB\), if \(f(x, a) = f(y, a)\) for every \(a \in AT\) then we see there is a relation for \(x\) and \(y\). This relation becomes an equivalence relation on OB, namely we can always define an equivalence relation EQ on OB. If a set \(X(\subset OB)\) is the union of some equivalence classes in EQ, then we call \(X\) is definable in DIS. Otherwise we call \(X\) is rough [1].

Now we go to the NIS. We define every NIS = (OB, AT, {VAL\(_a\) | \(a \in AT\)}, g), where g is a mapping such that \(g: OB \times AT \rightarrow P(\bigcup_{a \in AT} VAL_{a})\) (Power set for \(\bigcup_{a \in AT} VAL_{a}\))[3,4]. We need to remark that there are two interpretations for mapping g, namely AND-interpretation and OR-interpretation. For example, we can give the following two interpretations for \(g(\text{tom}, \text{language}) = \{\text{English, Polish, Japanese}\}\).

© Springer-Verlag Berlin Heidelberg 1999
(AND-interpretation) Tom can use three languages, English, Polish and Japanese. Namely, we see $g(tom, language) = English \land Polish \land Japanese$.

(OR-interpretation) Tom can use either one of language in English, Polish or Japanese. Namely we see $g(tom, language) = English \lor Polish \lor Japanese$.

The OR-interpretation seems to be more important for $g$. Because, it is related to incomplete information and uncertain information. Furthermore, knowledge discovery, data mining and machine learning from incomplete information and uncertain information will be important issue. In such situation, we discuss $NIS$ with OR-interpretation. We have already proposed incomplete information and selective information for OR-interpretation [10], where we distinguished them by the existence of unknown real value. In this paper, we extend the contents in [10] and develop the algorithm for finding equivalence relations in $NIS$.

2 Aim and Purpose in Handling NIS

Now in this section, we show the aim and purpose in handling $NIS$. Let’s consider the following example.

**Example 1.** Suppose the next $NIS_1$ such that $OB = \{1, 2, 3, 4\}$, $AT = \{A, B, C\}$, $\bigcup_{a \in AT} VAL_a = \{1, 2, 3\}$ and $g$ is given by the following table.

<table>
<thead>
<tr>
<th>$OB$</th>
<th>$A$</th>
<th>$B$</th>
<th>$C$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 ∨ 2</td>
<td>2</td>
<td>1 ∨ 2 ∨ 3</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1 ∨ 2 ∨ 3</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1 ∨ 2</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>2</td>
<td>2 ∨ 3</td>
</tr>
</tbody>
</table>

**Table 1.** Non-deterministic Table for $NIS_1$

In this table, if we select an element for every disjunction then we get a $DIS$. There are $72(=2^3 \times 3^3 \times 2^2)$ $DIS$s for this $NIS_1$. In this case, we have the following issues.

**Issue 1:** For a set $\{1, 2\} (\subset OB)$, if we select 1 from $g(1, A)$ and 3 from $g(1, C)$, $g(2, C)$ and $g(4, C)$ then $\{1, 2\}$ is not definable. However, if we select 1 from $g(1, C)$ and $g(2, C)$ then $\{1, 2\}$ is definable. How can we check such definability for every subset $X$ of $OB$?

**Issue 2:** How can we get all possible equivalence relations from 72 $DIS$s? Do we have to check 72 $DIS$s sequentially?

**Issue 3:** Suppose there are following information for attribute $D$: $g(1, D) = \{1\}$, $g(2, D) = \{1\}$, $g(3, D) = \{2\}$ and $g(4, D) = \{2\}$, respectively. In this case, which $DIS$ from $NIS_1$ makes $(A, B, C) \to D$ consistent? How can we get all $DIS$s which make $(A, B, C) \to D$ consistent?

These issues come from the fact such that the equivalence relation in $DIS$ is always unique but there are some possible equivalence relations for $NIS$. 