Rough Set-Based Information Dilution by Non-deterministic Information

著者	Sakai Hiroshi, Wu Mao, Yamaguchi Naoto, Nakata Michinori
journal or publication title	Lecture Notes in Computer Science
volume	8170
page range	55-66
year	2013-10-11
URL	http://hdl.handle.net/10228/00006074

doi: info:doi/10.1007/978-3-642-41218-9_7

Rough Set-Based Information Dilution by Non-deterministic Information

Hiroshi Sakai[1], Mao Wu[2], Naoto Yamaguchi[2] and Michinori Nakata[3]

[1] Department of Basic Sciences, Faculty of Engineering,
Kyushu Institute of Technology Tobata, Kitakyushu 804-8550, Japan
sakai@mns.kyutech.ac.jp
[2] Graduate School of Engineering, Kyushu Institute of Technology
Tobata, Kitakyushu, 804-8550, Japan
wumogaku@yahoo.co.jp, KITYN1124@gmail.com
[3] Faculty of Management and Information Science,
Josai International University
Gumyo, Togane, Chiba 283, Japan
nakatam@ieee.org

Abstract. We have investigated rough set-based concepts for a given *Non-deterministic Information System* (NIS). In this paper, we consider generating a NIS from a *Deterministic Information System* (DIS) intentionally. A NIS Φ is seen as a diluted DIS ϕ, and we can hide the actual values in ϕ by using Φ. We name this way of hiding *Information Dilution* by non-deterministic information. This paper considers information dilution and its application to hiding the actual values in a table.

Keywords: Rough sets, NIS-Apriori algorithm, Information dilution, Privacy preserving, Randomization, Perturbation.

1 Introduction

In our previous research, we coped with rule generation in *Non-deterministic Information Systems* ($NISs$) [7, 11–13]. In contrast to *Deterministic Information Systems* ($DISs$) [9, 14], $NISs$ were proposed by Pawlak [9], Orłowska [8] and Lipski [5, 6] in order to better handle information incompleteness in data. We have proposed *certain* and *possible* rules in $NISs$, and proved an algorithm named *NIS-Apriori* is *sound* and *complete* for defined certain and possible rules. We have also implemented *NIS-Apriori* [10].

This paper considers the connection between information incompleteness and information hiding (or the randomization and the perturbation in privacy-preserving [2]). We intentionally add information incompleteness, i.e., non-deterministic values, to a DIS for hiding the actual values, then a DIS is translated to a NIS. For such a NIS, we can apply our previous framework including *NIS-Apriori*.

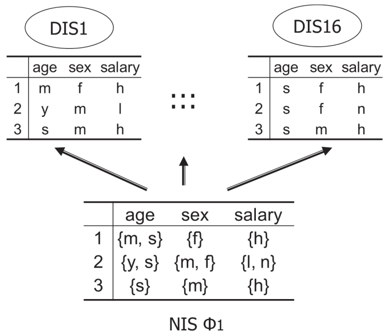

Fig. 1. $NIS\ \Phi_1$ and 16 derived $DISs$. Here, $Domain_{age}$=$\{\underline{y}oung, \underline{m}iddle, \underline{s}enior\}$, $Domain_{sex}$=$\{\underline{m}ale, \underline{f}emale\}$, $Domain_{salary}$=$\{\underline{l}ow, \underline{n}ormal, \underline{h}igh\}$. The number of derived $DISs$ is finite. However, it usually increases in the exponential order with respect to the level of incompleteness of $NIS's$ values.

The paper is organized as follows: Section 2 recalls rule generation in $NISs$. Section 3 introduces the framework of information dilution, and considers properties. Section 4 considers an algorithm for dilution and its relation to reduction [9], and Section 5 concludes the paper.

2 Apriori-Based Rule Generation in NISs

We omit any formal definition. Instead, we show an example in Figure 1. We identify a DIS with a standard table. In a NIS, each attribute value is a set. If the value is a singleton, there is no incompleteness. Otherwise, we have a set of possible values. We can interpret this situation by saying that each set includes the actual value but we do not know which of them is the actual one.

A *rule* (more correctly, a candidate for a rule) is an implication τ in the form of $Condition\_part \Rightarrow Decision\_part$. In a NIS, the same τ may be generated from different tuples, so we use notation τ^x to express that τ is generated by an object x. For example in Φ_1, an implication $\tau : [age, senior] \Rightarrow [salary, high]$ occurs in objects 1 and 3. Therefore, there are τ^1 and τ^3. If τ^x is the unique implication from an object x, we say τ^x is $definite$, and otherwise we say τ^x is $indefinite$. In this example, τ^1 is indefinite and τ^3 is definite.

In a DIS, the following holds for each $y \in [x]_{CON} \cap [x]_{DEC}$ (CON: condition attributes, DEC: decision attributes).

$support(\tau^y)$=$support(\tau^x)$, $accuracy(\tau^y)$=$accuracy(\tau^x)$.

Therefore, we may identify τ^x with τ. However in a NIS, this may not hold. The property of each τ^1 and τ^3 is slightly different, namely the one is indefinite and the other is definite. If there is at least one τ^x satisfying constraint, we see this τ^x is the *evidence* for causing τ is a rule. There may be other τ^y not satisfying the constraint. We employ this strategy for rule generation in a NIS.

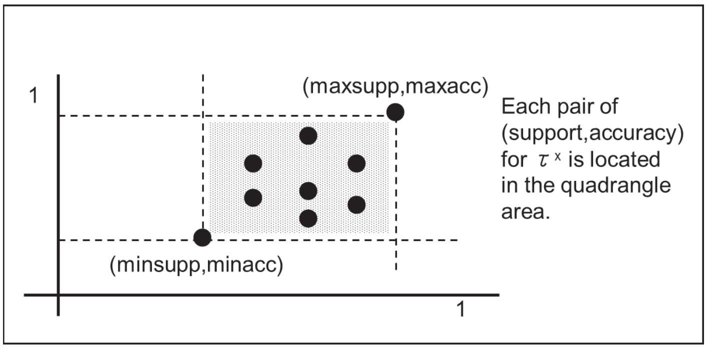

Fig. 2. A distribution of pairs (*support*,*accuracy*) for τ^x. There exists $\phi_{min} \in DD(\tau^x)$ which makes both $support(\tau^x)$ and $accuracy(\tau^x)$ the minimum. There exists $\phi_{max} \in DD(\tau^x)$ which makes both $support(\tau^x)$ and $accuracy(\tau^x)$ the maximum. We denote such quantities as *minsupp*, *minacc*, *maxsupp* and *maxacc*, respectively.

Let $DD(\Phi)$ and $DD(\tau^x)$ denote $\{\phi \mid \phi$ is a derived $DISs$ from $NIS\ \Phi\}$ and $\{\phi \in DD(\Phi) \mid \tau^x$ occurs in $\phi\ \}$, respectively. According to rule generation (employing *support* and *accuracy*) in $DISs$ [9], rule generation with missing values [3, 4] and data mining in transaction data [1], we defined the next tasks in rule generation in $NISs$ [11].

Specification of the rule generation tasks in a NIS
Let us consider the threshold values α and β $(0 < \alpha, \beta \leq 1)$.
(The Lower System: Certain rule generation task) Find each definite implication τ^x such that $support(\tau^x) \geq \alpha$ and $accuracy(\tau^x) \geq \beta$ hold in each $\phi \in DD(\tau^x)$. We say such τ is a *certain* rule and τ^x is an evidence of supporting τ in a NIS.
(The Upper System: Possible rule generation task) Find each implication τ^x such that $support(\tau^x) \geq \alpha$ and $accuracy(\tau^x) \geq \beta$ hold in some $\phi \in DD(\tau^x)$. If such τ is not certain rule, we say τ is a *possible* rule and τ^x is an evidence of supporting τ in a NIS.

Both the above tasks depend on $|DD(\tau^x)|$. In [11], we proved some simplifying results illustrated by Figure 2. We also showed how to effectively compute $support(\tau^x)$ and $accuracy(\tau^x)$ for ϕ_{min} and ϕ_{max} independently from $|DD(\tau^x)|$. Due to Figure 2, we have the following equivalent specification.

Equivalent specification of the rule generation tasks in a NIS
(The Lower System: Certain rule generation task) Find each definite τ^x such that $minsupp(\tau^x) \geq \alpha$ and $minacc(\tau^x) \geq \beta$ (see Figure 2).
(The Upper System: Possible rule generation task) Find each implication τ^x such that $maxsupp(\tau^x) \geq \alpha$ and $maxacc(\tau^x) \geq \beta$ (see Figure 2).

Example. In $NIS\ \Phi_1$, we at first generate two blocks inf and sup for each descriptor. These two blocks are the extensions from Grzymała-Busse's *blocks* [3, 4], and inf defines the minimum equivalence class. On the other hand, sup defines the maximum equivalence class. For example,

$inf([age,s])=\{3\}$, $sup([age,s])=\{1,2,3\}$,
$inf([salary,h])=\{1,3\}$, $sup([salary,h])=\{1,3\}$.

Since $sup([age,s]) \cap sup([salary,h])=\{1,3\}$, we know there are τ^1 and τ^3 for an implication $\tau : [age,s] \Rightarrow [salary,h]$. As for τ^3, $3 \in inf([age,s]) \cap inf([salary,h])$ holds, so we know τ^3 is definite. In this case, we have the following.

$$minsupp(\tau^3)=(|inf([age,s]) \cap inf([salary,h])|)/3=|\{3\}|/3=1/3.$$
$$minacc(\tau^3)=\frac{|inf([age,s]) \cap inf([salary,h])|}{(|inf([age,s])|+|OUT|)}=|\{3\}|/(|\{3\}|+|\{2\}|)=1/2.$$
$$maxsupp(\tau^3)=(|sup([age,s]) \cap sup([salary,h])|)/3=|\{1,3\}|/3=2/3.$$
$$maxacc(\tau^3)=\frac{|sup([age,s]) \cap sup([salary,h])|}{(|inf([age,s])|+|IN|)}=|\{1,3\}|/(|\{3\}|+|\{2\}|)=2/2=1.0.$$
$$OUT=(sup([age,s]) \setminus inf([age,s])) \setminus inf([salary,h]),$$
$$IN=(sup([age,s]) \setminus inf([age,s])) \cap sup([salary,h]).$$

In the above calculation, we do not handle $DD(\Phi_1)$ at all. By using blocks inf and sup, it is possible to calculate four criterion values. We extended rule generation to $NISs$ and implemented a software tool with NIS-*Apriori* algorithm [11]. NIS-*Apriori* does not depend on the number of derived $DISs$, and the complexity is almost the same as the original *Apriori* algorithm [1].

3 Information Dilution

This section considers a framework of information dilution.

3.1 An Intuitive Example

We at first consider DIS_{16} and Φ_1 in Figure 1. Since a DIS is a special case of a NIS, we can apply NIS-*Apriori* to each DIS. In this case, the lower and the upper systems generate the same rules. The following is the real execution under the decision attribute *salary*, α=0.5 and β=0.6.

```
?-step1. /* Rule generation in DIS16 under α=0.5 and β=0.6 */
File Name for Read Open: dis16.pl.
SUPPORT:0.5, ACCURACY:0.6
===== Lower System ========================================
[1] MINSUPP=0.667, MINACC=0.667
[age,senior] ==> [salary,high] [1,3] /* Obtained rule */
[2] MINSUPP=0.333, MINACC=0.5
(Lower System Terminated)
```

```
===== Upper System ========================================
[1] MAXSUPP=0.667, MAXACC=0.667
          :   :   :
EXEC_TIME=0.0 (sec)
```

We obtained an implication $[age, senior] \Rightarrow [salary, high]$ from DIS_{16}. Now, we consider the 2nd person's tuple $(senior, female, normal)$. If we employ the following replacement,

$senior$ to $[young, senior]$ (semantically $young$ or $senior$),
$female$ to $[male, female]$,
$normal$ to $[low, normal]$,

the 2nd person's tuple is changed to

$([young, senior], [male, female], [low, normal])$.

This is an example of information dilution. There are 8 possible tuples and one of the tuple is actual, so in such case we say the actual tuple is *diluted* with 1/8 degree. Similarly, DIS_{16} is diluted to Φ_1 with 1/16 degree in Figure 1. The following is the real execution of rule generation in Φ_1.

```
?-step1. /* Rule generation in Φ1 under α=0.5 and β=0.6 */
File Name for Read Open: Phi1.pl.
SUPPORT:0.5, ACCURACY:0.6
===== Lower System ========================================
(Lower System Terminated)
===== Upper System ========================================
[1] MAXSUPP=0.667, MAXACC=1.0
[age,senior] ==> [salary,high] [1,3] /* Obtained rule */
[2] MAXSUPP=0.333, MAXACC=1.0
[3] MAXSUPP=0.333, MAXACC=1.0
(Upper System Terminated)
EXEC_TIME=0.0 (sec)
```

In this execution, we know that the results (an obtained rule) in Φ_1 is the same as the original DIS_{16}. Namely, DIS_{16} and Φ_1 are equivalent in rule generation, but some actual values are hidden in Φ_1. Even though this example depends on threshold values α=0.5 and β=0.6, these DIS_{16} and Φ_1 give an example of *information dilution with obtainable rules preserved.*

Figure 3 shows the chart of information dilution, namely the relation between a DIS, a NIS and obtained rules. In data mining, we usually do not open the original data set, namely a DIS, to save privacy-preserving. However, we may open the diluted data set, namely a NIS, because some data in a NIS are changed to disjunctive information. We may consider diluting some specified

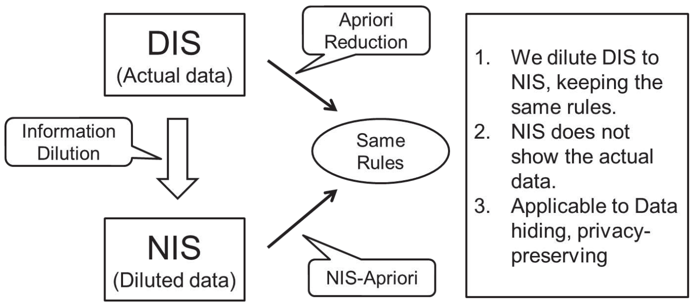

Fig. 3. Formalization of information dilution with constraint

person's data intentionally. Like this, information dilution may take the role of hiding the actual values in a table.

3.2 Some Properties and a Formalization of a Problem

Now, we confirm the following facts.

(Fact 1) A DIS ϕ is diluted to a NIS Φ.
(Fact 2) NIS-*Apriori* is applicable to Φ.
(Fact 3) For Φ diluted from ϕ, each rule in ϕ is obtainable by the upper system in Φ.

(Fact 3) is the key background. Let us suppose an implication τ^x satisfies $support(\tau^x) \geq \alpha$ and $accuracy(\tau^x) \geq \beta$ in ϕ, and ϕ is diluted to Φ. Then, we know $\phi \in DD(\tau^x) \subseteq DD(\Phi)$. According to the specification of the upper system, τ satisfies the condition of a possible rule, namely τ is obtainable in the upper system. However, we also have a problem. For $\phi' \in DD(\Phi)$ ($\phi' \neq \phi$), the upper system may pick up another implication η as a possible rule. Therefore, we need to know the next fact.

(Fact 4) For Φ diluted from ϕ, some rules not related to ϕ may be obtained by the upper system in Φ. We name such rules *unexpected rules*.
(Fact 5) If we dilute much more attribute values, we may have much more unexpected rules. On the other hand, if we dilute less attribute values, we will have less unexpected rules.

According to five facts, we have the problem in the following.

(Problem of Information Dilution) Dilute a DIS ϕ to a NIS Φ so as not to generate any unexpected rules.

4 A Example on an Algorithm for Information Dilution

We are now starting this work, and we are considering how to dilute a DIS to a NIS. Therefore, we employ an exemplary DIS ϕ_1 in Table 1 for considering an algorithm. For simplicity, we fix constraint such that the decision attribute is D, $maxsupp(\tau^x) = \alpha > 0$ and $maxacc(\tau^x) = \beta = 1.0$. In this example, we dilute ϕ_1 to a NIS with obtainable 7 rules preserved in Table 2. We can easily obtain them by using a software tool [10].

Table 1. An exemplary DIS ϕ_1. Here, $Domain_A$=$\{1,2,3\}$, $Domain_B$=$\{1,2\}$, $Domain_C$=$\{1,2\}$ and $Domain_D$=$\{1,2\}$.

OB	A	B	C	D
1	3	1	1	1
2	2	1	1	1
3	1	1	1	2
4	3	1	2	1
5	3	1	1	1
6	2	2	2	2
7	1	2	1	2
8	2	2	2	2

Table 2. Seven rules in ϕ_1.

	Rules	*Objects*
`(Imp 1)`	`[A,1]==>[D,2]`	`[3,7]`
`(Imp 2)`	`[A,3]==>[D,1]`	`[1,4,5]`
`(Imp 3)`	`[B,2]==>[D,2]`	`[6,7,8]`
`(Imp 4)`	`[A,2]&[B,1]==>[D,1]`	`[2]`
`(Imp 5)`	`[A,2]&[C,1]==>[D,1]`	`[2]`
`(Imp 6)`	`[A,2]&[C,2]==>[D,2]`	`[6,8]`
`(Imp 7)`	`[B,1]&[C,2]==>[D,1]`	`[4]`

4.1 Reduction and Dilution

Reduction seems to be applicable to information dilution, namely we apply reduction to a table, and we replace non-necessary attribute values with the set of all attribute values. However, this way is not sufficient for preserving the rules.

In ϕ_1, the degree of data dependency from $\{A, B, C\}$ to $\{D\}$ is 1.0, and 8 objects are consistent for condition attributes A, B, C and decision attribute D. In reduction, we have a tuple $(3, -, -, 1)$ from object 1, 4, 5, and a tuple $(1, -, -, 2)$ from object 3, 7, because they are still consistent. After this reduction, it seems possible to replace each $-$ symbol with all attribute values, i.e., $[1, 2]$. Like this we have a tuple $(1, [1, 2], [1, 2], 2)$ from object 3. In this tuple, we need to consider four cases $(1, 1, 1, 2)$, $(1, 1, 2, 2)$, $(1, 2, 1, 2)$ and $(1, 2, 2, 2)$. An implication $\tau^2 : [B, 1]\&[C, 1] \Rightarrow [D, 1]$ contradicts to $\eta^3 : [B, 1]\&[C, 1] \Rightarrow [D, 2]$ related to the tuple $(1, 1, 1, 2)$. However in other three cases, τ^2 does not contradict to any implication, and τ^2 becomes the unexpected rule.

4.2 Base Step Dilution: Dilution in Each Attribute

We propose a dilution process related to reduction. We start with NIS Φ_2 in Table 3, then we fix some attribute values which induce 7 rules.

Table 3. NIS Φ_2 at the beginning.

OB	A	B	C	D
1	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
2	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
3	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
4	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
5	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
6	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
7	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
8	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$

(Step 1-1) In order to generate (Imp 1), (Imp 2) and (Imp 3), we fix $[A,3]$ and $[D,1]$ in object $1 \in [1,4,5]$, $[A,1]$ and $[D,2]$ in object $7 \in [3,7]$, $[B,2]$ and $[D,2]$ in object $8 \in [6,7,8]$.
(Step 1-2) In order to generate inconsistency, we fix $[A,2]$ and $[D,1]$ in object 2, $[A,2]$ and $[D,2]$ in object 6, $[B,1]$ and $[D,1]$ in object 2, $[B,1]$ and $[D,2]$ in object 3, $[C,1]$ and $[D,1]$ in object 2, $[C,1]$ and $[D,2]$ in object 3, $[C,2]$ and $[D,1]$ in object 4, $[C,2]$ and $[D,2]$ in object 6.

Table 4. NIS Φ_3 after the base step.

OB	A	B	C	D
1	$\{3\}$	$\{1,2\}$	$\{1,2\}$	$\{1\}$
2	$\{2\}$	$\{1\}$	$\{1\}$	$\{1\}$
3	$\{1,2,3\}$	$\{1\}$	$\{1\}$	$\{2\}$
4	$\{1,2,3\}$	$\{1,2\}$	$\{2\}$	$\{1\}$
5	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
6	$\{2\}$	$\{1,2\}$	$\{2\}$	$\{2\}$
7	$\{1\}$	$\{1,2\}$	$\{1,2\}$	$\{2\}$
8	$\{1,2,3\}$	$\{2\}$	$\{1,2\}$	$\{2\}$

After these two steps, we have Φ_3 in Table 4. Since three implications (Imp 1), (Imp 2) and (Imp 3) appear in each derived DIS, they are also rules in the upper system. We have the next important fact.

(Fact 6) Any implication $\tau^x : [A,1]\&Condition\_part \Rightarrow [D,2]$ in a derived DIS $\phi \in DD(\Phi_3)$ is redundant for (Imp 1). Therefore, $accuracy(\tau^x) = 1.0$ holds in this ϕ. Any implication $\eta^y : [A,1]\&Condition\_part \Rightarrow [D,1]$ in a derived DIS $\phi' \in DD(\Phi_3)$ is inconsistent, because (Imp 1) also appears in this ϕ'. Therefore, $accuracy(\eta^y) < 1.0$ holds in this ϕ'. According to the above consideration, we do not have to pay any attention to any implication with a descriptor $[A,1]$. The same holds for descriptors $[A,3]$, $[B,2]$.

4.3 Recursive Steps Dilution: Dilution in a Set of Attributes

Similarly to the base step, we fix some attribute values for (Imp 4), (Imp 5), (Imp 6) and (Imp 7).

(Step 2-1) The attribute values of (Imp 4) and (Imp 5) are fixed in Φ_3. We fix $[B, 1]$ in object 4 and $[C, 2]$ in object 6.

According to (Fact 6), we do not have to consider any implication including descriptors $[A, 1]$, $[A, 3]$ and $[B, 2]$. It is enough to consider descriptors $[A, 2]$, $[B, 1]$, $[C, 1]$ and $[C, 2]$. Then, we have 10 implications, where unexpected rules may exist.

```
(1) [A,2]&[B,1] ==> [D,1],  (2) [A,2]&[B,1] ==> [D,2],
(3) [A,2]&[C,1] ==> [D,1],  (4) [A,2]&[B,1] ==> [D,2],
(5) [A,2]&[C,2] ==> [D,1],  (6) [A,2]&[C,2] ==> [D,2],
(7) [B,1]&[C,1] ==> [D,1],  (8) [B,1]&[C,1] ==> [D,2],
(9) [B,1]&[C,2] ==> [D,1],  (10) [B,1]&[C,2] ==> [D,2].
```

(Step 2-2) Here, (1) is (Imp 4), (3) is (Imp 5). They are obtainable in object 2. (6) is (Imp 6), which is obtainable in object 6. (9) is (Imp 7), and we fix $[B, 1]$ in object 4. According to (Fact 6), any of (2), (4), (5) and (10) does not satisfy $accuracy(\tau^x)$=1.0 in any derived $DISs$. (7) in object 2 and (8) in object 3 are inconsistent in any derived $DISs$.

After (Step 2-1) and (Step 2-2), we have Φ_4 below.

Table 5. NIS Φ_4 after the 2nd step.

OB	A	B	C	D
1	$\{3\}$	$\{1,2\}$	$\{1,2\}$	$\{1\}$
2	$\{2\}$	$\{1\}$	$\{1\}$	$\{1\}$
3	$\{1,2,3\}$	$\{1\}$	$\{1\}$	$\{2\}$
4	$\{1,2,3\}$	$\{1\}$	$\{2\}$	$\{1\}$
5	$\{1,2,3\}$	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$
6	$\{2\}$	$\{1,2\}$	$\{2\}$	$\{2\}$
7	$\{1\}$	$\{1,2\}$	$\{1,2\}$	$\{2\}$
8	$\{1,2,3\}$	$\{2\}$	$\{1,2\}$	$\{2\}$

In Φ_4, all 7 implications (Imp 1) to (Imp 7) are all obtainable. There is a conjunction of descriptors $[B, 1]\&[C, 1]$ which causes inconsistency, so we need to consider a conjunction of descriptors $[A, \_]\&[B, 1]\&[C, 1]$. However, such conjunction is redundant, and we do not have to consider it. The following is the real execution. If there is an implication τ^x, $maxsupp(\tau^x)$>0.1 holds. Therefore, we set α=0.1 instead of α>0.

```
?-step1. /* Rule p ⇒ q in Φ4 under α=0.1 and β=1.0 */
File Name for Read Open: Phi4.pl.
SUPPORT:0.1, ACCURACY:1.0
===== Lower System =========================================
           :   :   :
(Next Candidates are Remained) [[[1,1],[4,2]],[[1,2],[4,1]], :::
===== Upper System =========================================
[1] MAXSUPP=0.125, MAXACC=0.5
[2] MAXSUPP=0.375, MAXACC=1.0
[a,1] ==> [d,2] [3,7,8]   /* (Imp 1) in φ1 */
           :   :   :
[5] MAXSUPP=0.375, MAXACC=1.0
[a,3] ==> [d,1] [1,4,5]   /* (Imp 2) in φ1 */
           :   :   :
[10] MAXSUPP=0.5, MAXACC=1.0
[b,2] ==> [d,2] [5,6,7,8]   /* (Imp 3) in φ1 */
(Next Candidates are Remained) [[[1,2],[4,1]],[[1,2],[4,2]], :::
EXEC_TIME=0.0 (sec)

?-step2. /* Rule p1&p2 ⇒ q in Φ4 under α=0.1 and β=1.0 */
===== Lower System =========================================
           :   :   :
(Next Candidates are Remained) [[[1,2],[2,1],[4,1]],[[1,2], :::
===== Upper System =========================================
[1] MAXSUPP=0.375, MAXACC=1.0
[a,2]&[b,1] ==> [d,1] [2,4,5]   /* (Imp 4) in φ1 */
           :   :   :
[3] MAXSUPP=0.25, MAXACC=1.0
[a,2]&[c,1] ==> [d,1] [2,5]   /* (Imp 5) in φ1 */
           :   :   :
[6] MAXSUPP=0.375, MAXACC=1.0
[a,2]&[c,2] ==> [d,2] [5,6,8]   /* (Imp 6) in φ1 */
           :   :   :
[9] MAXSUPP=0.375, MAXACC=1.0
[b,1]&[c,2] ==> [d,1] [1,4,5]   /* (Imp 7) in φ1 */
           :   :   :
(Next Candidates are Remained) [[[2,1],[3,1],[4,1]], :::
EXEC_TIME=0.0 (sec)

?-step3. /* Rule p1&p2&p3 ⇒ q in Φ4 under α=0.1 and β=1.0 */
===== Lower System =========================================
[1] MINSUPP=0.125, MINACC=0.333
           :   :   :
[4] MINSUPP=0.0, MINACC=0.0
(Lower System Terminated)
```

```
===== Upper System ==========================================
(Upper System Terminated)
EXEC_TIME=0.0 (sec)
```

In step 1, we obtained three implications (Imp 1), (Imp 2) and (Imp 3) in the upper system. In step 2, we obtained four implications (Imp 4) to (Imp 7) in the upper system. In step 3, we obtained no implications. In view of the above results, we have the following:

$$\{\tau \mid \tau \text{ is either a possible rule or a certain rule in } \Phi_4\}=\{\tau \mid \tau \text{ is a rule in } \phi_1\}.$$

This means that Φ_4 and ϕ_1 are equivalent in rule generation, and they are satisfying the formalization of Figure 3. Each tuple of ϕ_1 stores the actual values, therefore we should not open ϕ_1. However, it may be possible to open Φ_4, because some attribute values are diluted. Especially, the tuple of object 5 is completely diluted.

5 Concluding Remarks

We have proposed a framework of information dilution, which depends on the research on *RNIA* (*Rough Non-deterministic Information Analysis*) and *NIS-Apriori* algorithm. This is an attempt to apply information incompleteness and *RNIA* to the randomization and the perturbation in privacy-preserving [2].

We investigated the formal algorithm of diluting a *DIS* and its implementation. In Figure 1, we unexpectedly obtained that rules in DIS_{16} and Φ_1 are the same under *support* ≥ 0.5 and *accuracy* ≥ 0.6. In this paper, we handled the most simple case *support* > 0 and *accuracy*$=1.0$. The procedure proposed in this paper is a preliminary work towards more general cases.

In Φ_4 and ϕ_1, 13 attribute values are diluted for totally 32 attribute values. The ratio is about 1/3. We figure that this ratio is depending on the number of rules and total number of objects. Furthermore, (Fact 6) seems very important. If most descriptors are fixed in the base step, the number of implications are reduced in the recursive steps. Like several variations of reduction with several constraints, there may be several variations of information dilution.

Acknowledgment
The authors would be grateful for anonymous referees for their useful comments.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules, In : Proc. of *VLDB*, pp.487–499 (1994).
2. Aggarwal, C., Yu, P.: *Privacy-Preserving Data Mining*, *Advances in Database Systems* **34**, Springer (2008).

3. Grzymała-Busse, J.: A new version of the rule induction system LERS, *Fundamenta Informaticae* **31**, pp.27–39 (1997).
4. Grzymała-Busse, J., Rząsa, W.: A local version of the MLEM2 algorithm for rule induction, *Fundamenta Informaticae* **100**, pp.99–116 (2010).
5. Lipski, W.: On semantic issues connected with incomplete information data base, *ACM Trans. DBS.* **4**, pp.269–296 (1979).
6. Lipski, W.: On databases with incomplete information, *Journal of the ACM* **28**, pp.41–70 (1981).
7. Nakata, M., Sakai, H.: Twofold rough approximations under incomplete information, *International Journal of General Systems* **42**(6), pp.546–571 (2013).
8. Orłowska, E., Pawlak, Z.: Representation of nondeterministic information, *Theoretical Computer Science* **29**, pp.27–39 (1984).
9. Pawlak, Z.: *Rough Sets*, Kluwer Academic Publishers (1991).
10. RNIA software logs: `http://www.mns.kyutech.ac.jp/~sakai/RNIA`
11. Sakai, H., Ishibashi, R., Nakata, M.: On rules and apriori algorithm in non-deterministic information systems, *Transactions on Rough Sets* **9**, pp.328–350 (2008).
12. Sakai, H., Okuma, H., Nakata, M.: Rough non-deterministic information analysis: Foundations and its perspective in machine learning, *Smart Innovation, Systems and Technologies* **13**, Springer, Chapter 9, pp.215–247 (2013).
13. Sakai, H., Okuma, H., Wu M., Nakata, M.: Rough non-deterministic information analysis for uncertain information, *The Handbook on Reasoning-Based Intelligent Systems*, World Scientific, Chapter 4, pp.81–118 (2013).
14. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems, In: Intelligent Decision Support - *Handbook of Advances and Applications of the Rough Set Theory*, Kluwer Academic Publishers, pp.331–362 (1992).