Generation and Mapping of Multi-Reducts Based on Nearest Neighbor Relation

Naohiro Ishii [*†]
Aichi Institute of Technology
Toyota,470-0392 Japan
E-mail: ishii@aitech.ac.jp

Ippei Torii
Aichi Institute of Technology
Toyota, 470-0392 Japan
E-mail: mac@aitech.ac.jp

Toyoshiro Nakashima
Sugiyama Jyogakuen University
Nagoya, 464-8662 Japan
E-mail: nakasima@sugiyama-u.ac.jp

Hidekazu Tanaka
Daido University
Nagoya, 457-8530 Japan
E-mail: hitanaka@daido-it.ac.jp

Abstract

Dimension reduction of data is an important theme in the data processing to represent and manipulate higher dimensional data. Rough set is fundamental and useful to process higher dimensional data. Reduct in the rough set is a minimal subset of features, which has almost the same discernible power as the entire features in the higher dimensional scheme. Combination of multi-reducts is effective for parallel processing of the classification. Nearest neighbor relation between different classes has a basic information for classification. We propose here a multi-reduct parallel processing classification scheme with efficient and higher accuracy by using nearest neighbor relation. To improve the classification ability of reducts, we develop a generation method of reducts and its graph mapping method by using the nearest neighbor relation, which is based on characteristics of the weighted reducts for the classification. Further, a dependency relation and an embedding of nearest neighbor relation are proposed to improve the classification accuracy.

Keywords: classification; reduct generation; mapping of reducts; multi- reducts; nearest neighbor relation; embedding of nearest neighbor

1. Introduction

Rough sets theory firstly introduced by Pawlak [1,2,3,4] provides us a new approach to perform data analysis, practically. Up to now, rough set has been applied successfully and widely in machine learning and data mining. The need to manipulate higher dimensional data in the web and to support or process them gives rise to the question of how to represent the data in a lower-dimensional space to allow more space and time efficient computation. Thus, dimension reduction of data still remains as an important problem [5]. An important task in rough set based data analysis is computation of the attributes or feature reducts for the classification. By Pawlak's rough set theory [1,2,3,4], a reduct is a minimal subset of features, which has the discernibility power as using the entire features. Then, the reduct uses a minimum number of features and represents a minimal and complete rules set to classify new objects. Thus, reducts use partial data to classify objects, while conventional methods use all data. Finding all reducts of an information system is combinatorial NP-hard computational problem [2,3,4]. For the classification, multi-reducts are useful for the improvement of the classification accuracy, which is a kind of parallel processing scheme for the classification. Then, for the efficient and speedy classification of the parallel processing, nearest neighbor method by Cover and Hart [8] is simple, efficient and speedy with multi-reducts. We propose here a parallel processing scheme of multi-reducts followed by nearest neighbor classification [10,11,12,13]. Nearest neighbor relation [11,12] between different classes has a basic information for classification. We develop here reduct generation based on the nearest neighbor relation. Then, it is not necessarily efficient to apply reducts directly to nearest neighbor classification method. Here, we propose a modified reduct, which are added redundant attributes to reducts for nearest neighbor classification. There are problems when reducts are used for the classification. As the classification, nearest neighbor method is simple and effective one. But, only attributes in the reducts are not sufficient for the application o the nearest neighbor. To cope with this problem, a graphical mapping of nearest neighbor relation is developed here for generating modified reduct. The mapping procedure and the relations of weighted attribute values are developed. It is shown that the weighted modified reduct improves the classification accuracy, which is followed by the nearest neighbor classification. Finally, nearest neighbor classification method is closely related to neighbor data. So, to improve the accuracy actively, some methods to effect to neighbor data are expected. As a solution to this problem, an embedding of nearest neighbor is proposed to improve the classification accuracy. It is shown that the method improves classification accuracy in the UCI data set [14].

2. Generation of Reducts Based on Nearest Neighbor Relation

Classification is fundamental, important and useful for data processing of documents, learning machines and web searching. There are problems remained to cope with the intelligent and effective classification under the condition of the developed computer environments. To cope with huge data, it is expected to reduce the total data as small as possible. Reduct conception is useful method for the classification. The proposed classification is shown in Fig.1, which consists of some reducts followed by respective Nearest Neighbor(NN) system [10,11]. Nearest neighbor is different from other classification methods such as linear discriminate analysis, neural networks, support vector machine or kernel methods [13]. The nearest neighbor focuses on the nearest neighbor local points around the given point. This method produce a new concept of mapping and embedding proposed here. Then, to improve the performance of the nearest neighbor classification, mapping and embedding of nearest neighbor strategy is proposed and experimented.

A knowledge representing system containing the set of attributes, called condition attributes and the set of decision attributes, is called a decision table. The decision table is useful for classification. Here, a simple example of the decision table is shown in Table 1.

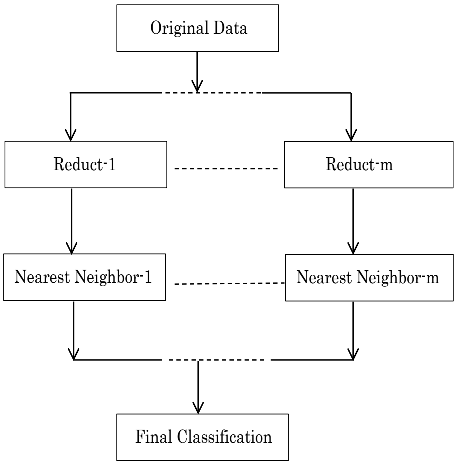

Fig. 1. Classification by using multi-reducts

The left side data in the column in Table 1 as shown in, $\{x_1, x_2, x_3, .., x_7\}$ is a set of instances, while the data { a, b, c, d} on the upper row, shows the set of attributes of the instance. The contents of each row in Table 1 shows numeral values of the corresponding instance. The class shows that the corresponding instance belongs to the numeral class value.

Table 1. Data example of decision table

Attribute	a	b	c	d	class
x_1	0	0	2	1	1
x_2	1	0	2	0	1
x_3	2	2	0	0	2
x_4	1	2	2	1	2
x_5	2	1	0	1	2
x_6	1	1	1	0	1
x_7	2	1	2	1	2

2.1. *Generation of Reducts*

Skowron[4, 6,7] proposed to represent decision table in forma of a discernibility matrix. This representation has many advantages, in particular it enables simple computation of the core, reducts other concepts. The discernibility matrix is defined as follows. Let $T = \{U, A, C, D\}$ be a decision table, with $U = \{x_1, x_2, ...x_n\}$. By a discernibility matrix of T, denoted by $M(T)$, which is $n \times n$ matrix defined as

$$m_{ij} = \{a \in C : a(x_i) \neq a(x_j) \\ \wedge (d \in D, d(x_i) \neq d(x_j))\}\ i, j = 1, 2, ...n \tag{1}$$

,where U is the universe of discourse, C is a set of features,

A is a subset of C called condition, and D is a set of decision features.

In Fig.2, the dicernibility matrix of the decision table in Table 1 is shown. In case of instance x_1, the value of the attribute a, is a(x_1)=1. Similarly, that of the attribute b, is b(x_1)=0.

Since a(x_1)=1 and a(x_5)=2, a(x_1) ≠ a(x_5) holds.

	x_1	x_2	x_3	x_4	x_5	x_6
x_2	—					
x_3	b,c,d	b,c				
x_4	b	b,d	—			
x_5	a,b,c,d	a,b,c	—	—		
x_6	—	—	a,b,c	a,b,c,d	c,d	
x_7	a,b	a,b,d	—	—	—	c,d

Fig.2. Dicernibility matrix of the decision table in Table 1

The dicernibility function is represented by taking the combination of the disjunction expression of the discerniblity matrix. In Fig.2, the item (b,c,d) in the second row and the first column, implies b ∨ c ∨ d in the Boolean expression, which shows the attribute b, c and d appear together for the discrimination between instances x_1 and x_3 [4,6,7]. By using nearest neighbor relation, generation of reducts is developed in the next session, in which nearest neighbor plays an important role.

2.2. *Generation of Reducts Based on Nearest Neighbor*

Nearest neighbor points have useful information in data manipulation. Reduct is a fundamental concept in rough set[1,2], which is a reduced sub-set characterizing total data set. In this paper , reducts generation is proposed using neighbor relation. We assume here two classes A and B.

Algorithm of Generation of Reducts Based on NN

Step1 On the discernibility matrix, nearest neighbor relations are picked out, which consists of nearest neighbor pairs as the set of tuple{(a,b)}, a ∈ A, while b ∈ B between different classes A and B.

Step2 Nearest neighbor relations is defined to be a set E on the discernibility matrix in Fig.2, in which each term of E is shown in shading. Then, the discerniblity sub-function $f_{NN}(E)$ is made to be the Boolean conjunction expression of each shading term in Fig.2.

Step3 We can here define a sub-core concept on the derived Boolean expression. By using sub-cores, $f_{NN}(E)$ with Boolean expression is simplified in the expression.

Step4 By using the sub-core, a simplified expression outside of the nearest neighbor relations is carried out. The remained simplified Boolean expression becomes reducts of the original discernibility matrix.

An example of reducts generated by the nearest neighbor relations is shown in the following. First, in Table 1, nearest neighbor relations are detected as follows and shown in the shading cells

$$\{(x_1,x_7),(x_5,x_6),(x_6,x_7)\} \qquad (2)$$

In the relation (x_1,x_7) , the x_1 in the class 1 is nearest to the x_7 is in the class 2. Similarly, (x_5,x_6) and (x_6,x_7) are nearest relations. Then, these relations are shown in shading in Fig.2 of the discernible matrix. The $f_{NN}(E)$ of the Boolean product of these three terms of (2) becomes

$$(a \vee b) \wedge (c \vee d) = a \wedge (c \vee d) \vee b \wedge (c \vee d) \qquad (3)$$

These terms are candidates of reducts. Second, attribute a or b is searched except the nearest relations (5) in the discernible matrix, Fig.2. Thus, { b } is found in the relation (x_1,x_4) . The Boolean product (b) and other terms except nearest relations becomes b . The Boolean product b and obtained candidates (5) of reducts from nearest relations, becomes

$$\begin{aligned} & b \wedge (a \wedge (c \vee d) \vee b \wedge (c \vee d)) \\ & = (b \wedge c) \vee (b \wedge d) \end{aligned} \qquad (4)$$

Thus, reducts $b \wedge c$ and $b \wedge d$ are obtained by using nearest neighbor and $\{b\}$ becomes the core of reducts.

3. Mapping of Nearest Neighbo on Reducts

Characterization of nearest neighbor relations is developed in the weighting of the coordinate in the data expression. Then, classification accuracy is objective function for the design of the weighting of the coordinates. Classification accuracy of reduct {b,c} which is applied to nearest neighbor becomes 0.50. Also, the classification accuracy of the reduct {b,d} applied to nearest neighbor becomes 0.58. The Powlak's reducts [1,2,3] connected directly to nearest neighbor method shows insufficient classification accuracy. To improve the classification accuracy by using reducts connected to nearest neighbor, the modified reducts and weighting of the distance measure are proposed based on the mapping graph of the nearest neighbor. First, the weighted distance measure is introduced as shown in equation (5).The equation (5) shows a weighted Euclidean distance. To isolate neighbor points between different classes, the weight of the respective coordinate is compared between coordinates [10].

$$Weighted\ Distance = \sqrt{\sum_{i=1}^{n} \omega_i (x_i - y_i)^2} \qquad (5)$$

The weight of the orthogonal coordinate in graphical mapping is discussed as follows. The instances (0,2) and (1,1) in the class 1 are nearest to the (1,2) and (1,0) in the class 2, respectively based on the reduct {b,c}

1) It is expected to isolate the (0,2) from the (1,2) by the weight of the mapping coordinate. The first b-component is different between (0,2) in the class 1 and (1,2) in the class 2. Thus, the weight ω_b is taken as $\omega_b > 1$ for the isolation of these instances. This ω_b operation is counted once.

2) It is expected to isolate the (1,1) from the (1,2), also, the (1,1) from the (1,0). Thus, the weight ω_c is taken as $\omega_c > 1$ for the isolation of these instances.

3) It is expected to bring the (1,2) and the (2,2), also (1,0) and (2,0), since these instances are in the class 2. Thus, the weight ω_b is desired to be smaller value. This operation ω_c is counted twice which shows weight ω_c is set more value than ω_b .

By the above discussions 1) , 2) and 3), the following equation is derived for the reduct {b,d}.

$$\omega_c > \omega_b > 1 \qquad (6)$$

Similarly, the nearest neighbor relation is mapped on the reduct {b,d}. The instances are graphically mapped in the order of the nearest neighbor relation with Euclidean distance 1. The weight of the orthogonal coordinate is estimated in the following steps.

4) It is expected to isolate the (0,1) in the class 1 from the (1,1) in the class 2, also the (1,0) from the (2,0). Then, the weight ω_b is taken as $\omega_b > 1$. Thus, the following weight relation is derived,

$$\omega_b > \omega_d = 1 \qquad (7)$$

4. Extension to Modified Reducts

Reducts for the application to the nearest neighbor classifi-cation is developed. Based on the reducts {b, c} and {b, d}, the following three modified reducts are derived on the discernibility matrix,

{b,c,a} , {b,c,d} and {b,d,a}

Based on these modified reducts, an extended reduct {b,c,d,a} is derived, which consists of total variables. The relation of these modified and the extended reducts is shown in Fig.3

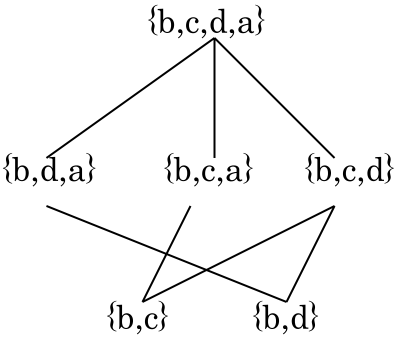

Fig. 3. Relation among reducts, modified reducts and the extended reduct

4.1. *Evaluation Based on Reduct {b,c}*

First, the modified reduct {b,c,a} is generated from the reduct {b,c} as shown in Fig.3. The a-weight relation between {b,c,a} and {b,c} is analyzed in the following.

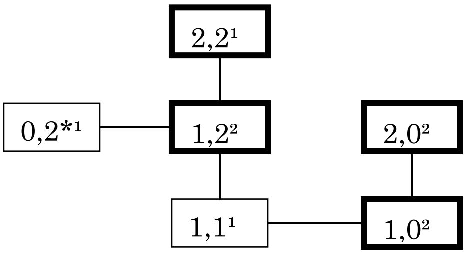

Fig. 4. Evaluation of attribute {a} based on reduct {b,c}

The upper suffix of attribute a is shown in each instance box. The a value of (b,c)= (2,2) is the suffix value 1. To isolate the bold instance box in the class 2 from the thin instance in the class 1, the weight value of the attribute a is important.

To isolate (0,2) from (1,2) , the weight of the a is increased. (* implies 0 value in the (0,2)). Since * and 1 in (0,2) from 2 in (1,2) are counted twice, there are two pairs by the increase of the a-weight. Similarly, since 1 in (1,1) from 2 in (1,0) is counted once, there is a pair by the increase of the a-weight. Also,1 in (1,1) from 2 in (1,2) is counted once, there is a pair by the increase of a-weight. Thus, there are 4 pairs by the increase of the a-weight. Between (2,2) and (1,2) in the same 2 class, the value of the a-weight is expected to be reduced , since 1 in (2,2) and 2 in (1,2) is expected to be reduced. Thus, only one pair decrease is counted. As the total value of the a-weight, there are 3 pairs increase (by the difference of 4 times increase – one decrease) of the a-weight value. Here, the 4 pairs by the increase of the a-weight is closely related to the nearest neighbor relation. Next, the modified reduct{b,c,d} is generated from the reduct{b,c} as shown in Fig.5. The d-weight relation between {b,c,d} and {b,c} is analyzed in the following.

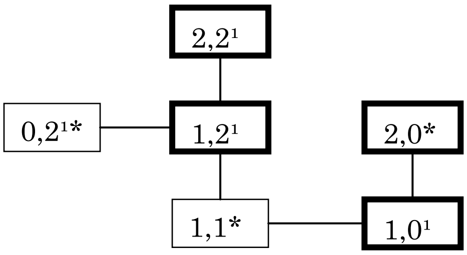

Fig. 5. Evaluation of attribute {d} based on reduct {b,c}

In Fig.5, there are 3 pairs by the increase of the d-weight , while 2 pairs by the decrease of the d-weight is observed. Thus, by the difference between the increase and the decrease counts, there is one time of the increase of the d-weight. Here, the 3 pairs by the increase of the d weight is closely related to the nearest neighbor relation. By comparing the result of the counts in the nearest neighbor relation derived in Fig.4 with that in Fig.5, the a-weight, ω_a is taken1 to be larger than the d-weight, ω_d corresponding to respective counts. Thus, the following a-weight and d-weight relation is obtained,

$$\omega_a > \omega_d \qquad (8)$$

4.2. *Evaluation Based on Reduct {b,d}*

Similar analysis is carried out on the reduct {b,d}. Modified reduct {b,d,a} is generated from the reduct {b,d} as shown i [11,12]n Fig.3. The a-weight relation between {b,d,a} and {b,d} is analyzed in the following.

In Fig.6, there are 3 pairs by the increase of the a-weight, while 3 times of the decrease of the a-weight is observed. The symbol,* shows 0 value in Fig.6. By the difference between the increase and the decrease counts, there is zero time of the increase of the d-weight. Here, the 3 pairs by the increase is closely related to the nearest neighbor relation. Further, the modified reduct {b,d,c} is generated from the reduct {b,d} as shown in Fig.3. The c-weight relation between {b,d,c} and {b,d} is analyzed in the following.

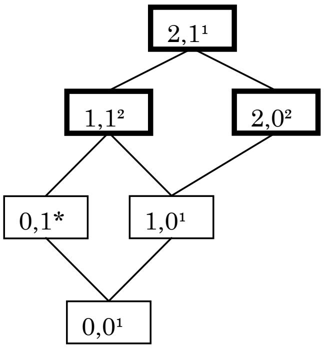

Fig. 6. Evaluation of attribute {a} based on reduct {b,d}

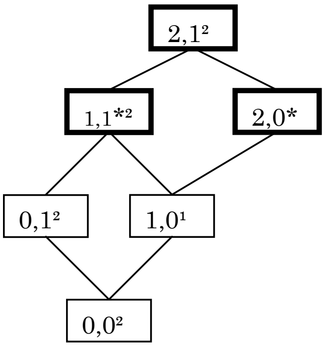

Fig. 7. Evaluation of attribute {c} based on reduct {b,d}

In Fig.7, there are 4 pairs by the increase of the c-weight, while 4 pairs of the decrease of the c-weight are observed. The symbol, * shows 0 value in Fig.7. By the difference between the increase and the decrease counts, there is zero time of the increase of the c-weight. Here, the 4 pairs by the increase is closely related to the nearest neighbor relation. By comparing the result of the nearest neighbor relation derived in Fig.6 with that in Fig.7, the a-weight, ω_a is taken to be larger than the c-weight, ω_c corresponding to respective counts. Thus, the following a-weight and c-weight relation is obtained,

$$\omega_c > \omega_a \tag{9}$$

4.3. *Relation between a-weight and b-weight*

To make clear the relation between the a-weight and the b-weight, the following mapping is developed in Fig.6. The mapping of the attribute b based on the pair {c,d} is shown. To isolate $(2,1^1)$ from $(2,1^2)$, the increase of the b-weight is needed. To isolate $(0,0^2)$ from $(1,0^1)$, also that increase is needed. Similarly, the increases between $(2,0^*)$ and $(2,1^2)$, ($2,1^1$) and $(2,0^*)$, and ($2,1^1$) and $(2,1^*)$ are needed. Thus, 5 pairs in the nearest neighbor relation are obtained. The mapping of the attribute a based on the pair {c,d)} is obtained. To isolate $(2,1^*)$ from $(2,1^2)$, the increase of the b-weight is needed. To isolate $(0,0^2)$ from $(1,0^1)$, also that increase is needed. Similarly, the increases between $(2,0^1)$ and $(2,1^2)$ and between $(2,1^*)$ and $(2,1^2)$ are needed. Thus, 4 pairs in the nearest neighbor relation are obtained. By comparing the result of the nearest neighbor relation, the b-weight, ω_b is taken to be larger than the a-weight ω_a, corresponding to respective counts. Thus, the following b-weight and a-weight relation is obtained,

$$\omega_b > \omega_a \tag{10}$$

4.4. *Order of Mapping of Nearest Neighbor*

By the mapping of the nearest neighbor relation, the following attribute weight relations are obtained. From equations (6) and (7),

$$\omega_c > \omega_b > \omega_d = 1 \tag{11}$$

is satisfied. From equations (10) and (11),

$$\omega_c > \omega_b > \omega_a > \omega_d = 1 \tag{12}$$

As an example, the weights of attributes are set as $\omega_c = 1.3$, $\omega_b = 1.2$, $\omega_a = 1.1$ and $\omega_d = 1.0$.

1) In case of the reduct {b,d}, the weight of attribute is set as $\omega_b = 1.2$ and $\omega_d = 1.0$. Then, the classification of the instances x_5, x_6 and x_7 in Table 1 is misclassified. Thus, the accuracy of the classification becomes 4/7=0.56 (56%),

2) In case of the reduct {b,c}, the weight of attributes is set as $\omega_c = 1.3$ and $\omega_b = 1.2$. Then, the classification of the instances x_6 in Table 1 is misclassified. Nearest neighbor of the instance x_7 is x_2 in the class 1 and x_4 in the class2. Thus, the x_7 is evaluated as the 0.5 point. Thus the accuracy of the classification becomes 5.5/7=0.78(78%).

3) In case of the modified reduct {b,c,d}, the weight of attributes is set as $\omega_b = 1.2$, $\omega_c = 1.3$ and $\omega_d = 1.0$. Then, the accuracy becomes 6/7=0.86(86%).

4) In case of the modified reduct {b,c,a}, the weight of attributes is set as $\omega_b = 1.2$, $\omega_c = 1.3$ and $\omega_a = 1.1$ .Then, the accuracy becomes 6.5/7=0.95(95%).

5) In case of the modified reduct {b,d,a}, the weigt of attributes is set as $\omega_b = 1.2$, $\omega_d = 1.0$ and $\omega_a = 1.1$ Then, the accuracy becomes 7/7=1.0(100%).

By combining processing as shown in Fig.1, the majority voting is applied to the modified reducts {b,c,d}, {b,c,a} and {b,d,a} . Then the final classification accuracy becomes 1.0(100%). By gathering these three modified reducts, the extended modified reduct {a,b,c,d} is used. The weights of attributes become $\omega_c = 1.3$, $\omega_b = 1.2$, $\omega_a = 1.1$ and $\omega_d = 1.0$. In this case, the accuracy becomes 1.0(100%). The classification accuracy of reducts {b, d} and {b, c}, weighted modified reducts are computed. Further, the combining factors for the parallel processing of multi-reduct classification are derived from the dependency computations by Pollack [1,2] as shown in the next section.

5. Dependency Relations for Parallel Processing of Reducts

To make clear the dependency between the modified reduct and its original related reducts, the notion of the dependency of knowledge developed by Pawlak is applied here [1,2,3,4]. Then, the dependency implies here whether the modified reducts are stable , related or not by extending original reducts. In Fig.2, it is expected to show how much the modified reduct {b,c,d} is related to reduct {b,c} and {b,d}. By using Pawlak's notation, reduct {b,c} –NN classification is represented by the relation as

$$U / R_{\{b,c\}} = \{\{x_3, x_4\}, \{x_1, x_2, x_6, x_7\}, \{x_5\}\} \tag{13}$$

$$U / R_{\{b,d\}} = \{\{x_2, x_4\}, \{x_1, x_3, x_6\}, \{x_5, x_7\}\} \tag{14}$$

$$U / Mod\, R_{\{b,c,d\}} = \{\{x_1, x_2, x_3, x_4, x_5\}, \{x_6, x_7\}\} \tag{15}$$

,where U in equation (13) is the set of all instances, $R_{\{b,c\}}$ is the reduct {b,c} followed by nearest neighbor relation. Then $U / R_{\{b,c\}}$ becomes the correctly classified set, $\{x_3, x_4\}$, incorrectly classified set, $\{x_1, x_2, x_6\}$ and the undecided set, $\{x_5, x_7\}$. Similarly, the equation (14) holds in reduct {b,d} and the equation (15) in modified reduct {b,c,d}. The modified reduct relation with weight, $Mod\, R_{\{b,c,d\}}$ depends in a degree k $(0 \le k \le 1)$ from the reduct relation $R_{\{b,c\}}$ as follows

$$k_{bcd(bc)} = \frac{card\ \{POS_{R_{\{b,c\}}}(Mod\, R_{\{b,c,d\}})\}}{card\ \{U\}} \tag{16}$$

,where $card\{\ \}$ denotes cardinality of the set. Equation (8) is a numerical value.

Since $POS_{R_{\{b,c\}}}(Mod\, R_{\{b,c,d\}})$ = $\{x_3, x_4, x_5\}$ holds, $k_{bcd(bc)} = 3/7$. Similarly, $POS_{R_{\{b,d\}}}(Mod\, R_{\{b,c,d\}})$ = $\{x_2, x_4\}$ holds, Thus, $k_{bcd(bd)} = 2/7$. Then, k_{bcd} is taken as the average value of $k_{bcd(bc)}$ and $k_{bcd(bd)}$. Similar dependency discussions are carried out on the modified reducts {b,c,a} and {b,d,a}.

$$U / Mod\, R_{\{b,c,a\}} = \{\{x_1, x_2, x_3, x_4, x_5, x_6\}, \{x_7\}\}$$

$$U / Mod\, R_{\{b,d,a\}} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$$

$$k_{bca} = \frac{card\ \{POS_{R_{\{b,c\}}}(Mod\, R_{\{b,c,a\}})\}}{card\ \{U\}} = 3/7$$

$$k_{bda} = \frac{card\ \{POS_{R_{\{b,d\}}}(Mod\, R_{\{b,d,a\}})\}}{card\ \{U\}} = 1$$

The classification class is computed by using dependency factors { $k_{bca}, k_{bda}, k_{bcd}$ } of modified reducts {b,c,a}, {b,d,a} and {b,c,d}, respectively. When an instace is decided to be in a certain class by the modified {b,d,a} with $k_{bda} = 1$ and other two reducts determine another same class with $k_{bca} + k_{bcd} < 1$, then the classification class belongs to the class determined by {b,d,a}. Thus, the classification class is determined as the final integrated computation for the parallel processing of the multi-reduct. As the results of the final classification , the classification accuracies of reducts {b,d},{b,c}and weighted modified reducts {b,c,a},{b,d,a}, and {b,c,d} are compared in Fig.8. Fig 8 shows that the original reducts {b,d} and {b,c} followed by the nearest neighbor have lower accuracy, while the weighted modified reducts have higher accuracy for the classification. These multi-reducts are computed in the parallel computing scheme. After these parallel computations, final integrated computation is needed. Here, dependency relation developed by Powlack [1,2] was applied. Thus, the final classification (Final Class. in Fig.8) shows a higher accuracy value to be 1. Thus, the integrated computation shows the dependency will be a useful measure for combining the parallel processing of the classification as shown in Fig.8.

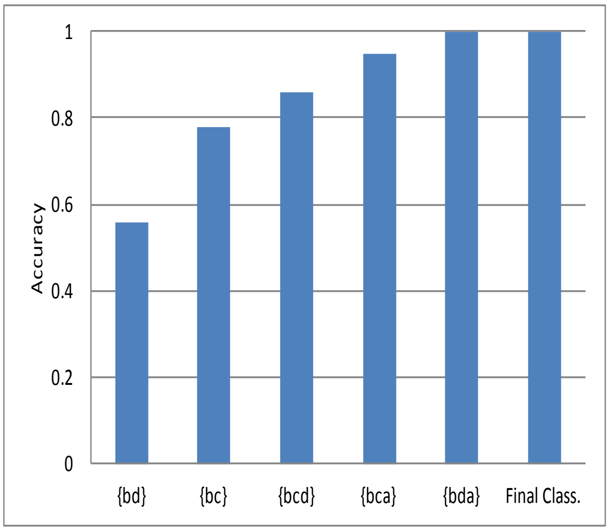

Fig. 8. Classification accuracy by reducts and weighted modified reducts by mapping

6. Embedding of Subspaces of Nearest Neighbor

Next characterization of nearest neighbors is developed as an embedding of subspaces. To improve the accuracy of the performance of nearest neighbor method, it is expected to develop any new method, which is different from such as the conventional kernel methods with applying nonlinear mapping from the neighborhood space. Here, an embedding of subspaces of nearest neighbor points is developed.

Embedding of subspace of nearest neighbor is interpreted as follows. We assume nearest neighbor points of the training set. In Fig.9, x and y are neighbor points, each of which belongs to different class. Since x and y are near, but in the different class, they will often cause missed classification. To prevent the error of the classification, newly generated points near the x and y, will help their classification.

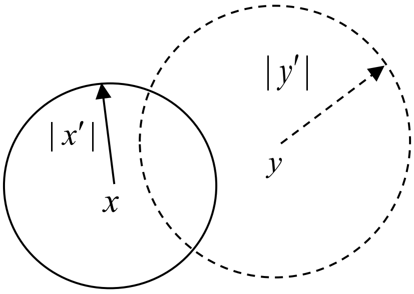

Fig.9. Embedding of subspace of nearest neighbor

The following steps are considered. The distance $|x'|=|x+\varepsilon|$ is the radius from the center point x, while the distance $|y'|=|y+\delta|$ is that from the center point y. A lso, we assume the generated sets $\{x''||x''|\le|x'|\}$ and $\{y''||y''|\le|y'|\}$. Then, by choosing the smaller value ε and δ, the intersection between generated sets becomes

$$\{x''||x''|\le|x'|\}\cap\{y''||y''|\le|y'|\}=\phi \qquad (17)$$

,i.e, disjoint. By the generation of newly data points, $\{x''\}$ and $\{y''\}$, the $|y'|$ and y can be classified correctly. This embedding process is realized by the following strategy of the embedding computation in the next section.

6.1. *Embedding Computation*

An embedding computation is shown in the schematic diagram in Fig.10. By using the leaning data, the first learning process is carried out by the learnt data. The classification process is done by using the testing data in (a) in Fig.10. The classified testing data in (a) is compared in the correctly classified table (b), given in advance. The misclassified data, instance 1 and instance 2, are applied in the second learning process as shown in Fig.10, which is called here embedding computation. Near the instance 1 and 2, the near data to those are embedded to support the instance 1 and 2. By the iteration of embedding of misclassified data including training data, it is expected to improve the classification accuracy when compared to the basic nearest neighbor system. Some experiments have been conducted in the following embedding strategy.

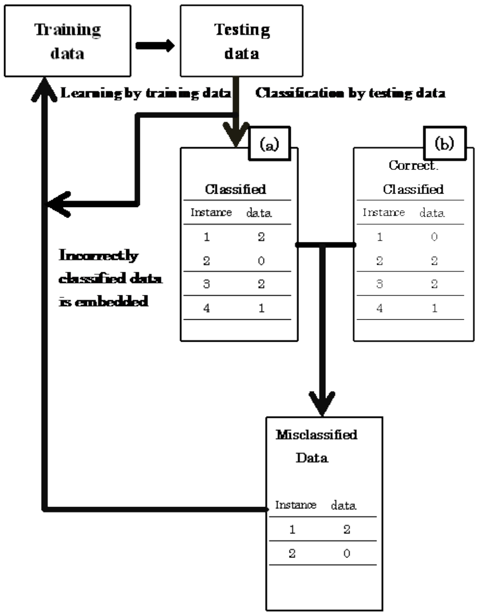

Fig. 10. An embedding computation strategy

Experiments to verify the embedding process have been conducted on some bench mark data sets from the UCI Machine Learning Repository [14]. The weighted reducts are computed by appplying Genetic Algorithm. Classification accuracy before and after is comared in Table 2. Classification before embedding means the processing by nearest neighbor method for the learnig, while classification after embedding means the application of embedding process for the learning. It is shown in Table 2. that the total average of the classification acuracy is improved 10% from 0.7704 to 0.8852 for the classification by applying embedding process. To compare the proposed embedding method with other conventional method, the classification accuracies by the well known algorithms are shown in Table 3, which is reported by Wilson [15]. In Table 3, the C4.5 system is an inductive decision tree algorithm [16]. The IB1 and IB2 are instance-based algorithms [17]. The IB1is a simple nearest neighbor classifier with k=1, while the IB2 prunes the training set. A naïve Bayesian classifier(Bayes) [18,19] and the back propagation algorithm (BP) in neural network [16,20] are also applied to these data. The IDBL [15] is an integrated decremental instance-based learning algorithm.

Table 2. Classification before and after embedding

Data set	Before	After	Data set	Before	After
bridges	0.6232	0.9859	hearlb	0.8301	0.8654
bupa	0.6134	0.9209	Heart swi	0.9385	0.9466
flag	0.5381	0.9601	hepat	0.7893	0.7903
glass	0.7439	0.9650	promotors	0.8153	0.8223
heart	0.7788	0.7663	wine	0.9739	0.9655
heartcle	0.7526	0.7524	zoo	0.9771	0.9930
hearthun	0.8447	0.8546	**Average**	**0.7704**	**0.8852**

Table 3. Accuracy by conventional algorithms [15]

DataSet	C4.5			IB		Bayes	BP	IDIBL
	Tree	P-Tree	Rule	IB1	IB2			
Bridges	68.00	65.30	59.50	53.80	45.60	66.10	67.60	63.20
Flag	59.20	61.30	60.70	63.80	59.80	52.50	58.20	57.70
Glass	68.30	68.80	68.60	70.00	66.80	71.80	68.70	70.60
Heart	73.30	72.10	80.00	76.20	68.90	75.60	82.60	83.30
Hepat	77.70	77.50	78.80	80.00	67.80	57.50	68.50	81.90
Promo	73.30	71.90	79.10	81.50	72.90	78.20	87.90	88.60
Zoo	91.00	91.00	91.40	96.40	97.50	97.80	95.60	92.20

By comparison with the conventional algorithms in Table 3, the embedding method based on nearest neighbor relation shows a better improvement for the classification accuracy.

7. Conclusion

Dimension reduction of data is an important problem as in web, data mining and image processing. Concept of rough set theory is a useful method to reduce the higher dimensional features. Then, the nearest neighbor relation between different classes has a basic information for classification. To connect reducts and nearest neighbor relations, parallel processing approach is needed for multi-reducts computations. In this paper, the characterization of the parallel processing by multi-reducts is analyzed for the higher classification accuracy.

First, we develop here reduct generation based on the nearest neighbor relation. Second, to apply reducts for the nearest neighbor classification, the reducts do not necessarily show higher classification accuracy. This paper develops a graphical mapping method of nearest neighbor for classification by the weighted modified reduct. It is shown that the combined parallel processing by weighted modified multi-reducts show a higher classification accuracy. Further, to improve the classification accuracy by the nearest neighbor strategy, an embedding of nearest neighbor is developed for the learning.

References

1. Z.Pawlak, Rough Sets, *International Journal of Computer and Information Science*, **11** (1982) 341-356.
2. Z.Pawlak, *Rough Sets:Theoretical Aspects and Reasoning about Data*,(Kluwer Academic Publishers, Dordrecht, 1991)
3. Z.Pawlak and R.Slowinski, Rough Set Approach to Multi-attribute Decision Analysis, *European Journal of Operations Research* **72**,(1994), 443-459.
4. Z.Pawlak and A.Skowron, Rough sets and Boolean reasoning, *Information Sciences* **177**(2007) 41-73.
5. K.Thangavel and A. Pethalakshmi, Dimensionality reduction based on rough set theory: A review, *Applied Soft Computing* **9** (2009) 1-12.
6. A.Skowron and C. Rauszer,, The Discernibility Matrices and Functions in *Information Systems, in Intelligent Decision Support- Handbook of Application and Advances of Rough Sets Theory* (Kluwer Academic Publishers, Dordrecht, 1992),pp.331-362.
7. A.Skowron and L.Polkowski, Decision Algorithms, A Survey of Rough Set Theoretic Methods, *Fundamenta Informatica,* **30**/3-4 (1997) 345-358
8. T.M.Cover and P.E.Hart, Nearest Neighbor Pattern Classification, *IEEE Transactions on I nformation Theory*, Vol.13, No.1(1967) 21-27.
9. J.Zaluski, R.Szoszkiewicz, J.Krysinski, and J.Stefanowski, Rough set theory and decision rules in data analysis of breast cancer patients, *Transactions on Rough Sets 1, LNCS*, **3100** (2004) 375-391.
10. N.Ishii,Y. Morioka, Y.Bao and H.Tanaka, Control of Variables in Reducts-kNN Classification with Confidence, *KES2011, LNCS*, **6884**,(Springer 2011)98-107.
11. N.Ishii, I.Torii, Y.Bao, and H.Tanaka, Modified Reduct Nearest Neighbor Classification, in *Proc. ACIS-ICIS*, (IEEE Comp.Soc.2012)pp.310-315.
12. N.Ishii, I.Torii, Y.Bao and H. Tanaka, Mapping of Nearest Neighbor for Classification, in *Proc. ACIS-ICIS*, (IEEE Comp.Soc.2013)pp.71-76
13. N.Ishii, Y. Morioka, S.Suyama and Y.Bao, Classification by Rough Set Reducts , AdaBoost and SVM, *in Proc. SNPD*, (IEEE Comp. Soc.2010) pp.63-68.
14. C.J.Merz and P.M.Murphy, UCI Repository of Machine Learning Databases. Irvine, CA: Internet: http://www.ics.uci.edu/mearn/MLRepository.html,(1998)
15. D.R.Wilson andT.R.Martinez, An Integrated Instance-Based Learning Algorithm, *Computational Intelligence*, **16**(1) (2000)1-28.
16. J.R.Quinlan, *C4.5: Programs for Machine Learning* (Morgan Kaufmann, 1993).
17. D.W.Aha, D.Kibler and M.K.Albert, Instance-Based Learning Algorithms, *Machine Learning*, 6(1991) 37-66.
18. P.Lanley, W.Iba and K.Thompson, An Analysis of Bayesian Classifiers, in *Proc. Of 10th national Conf. on A.I.(AAAI-92)*,(AAAI/MIT press,1992) pp.223-228.
19. D.Michie, D. Spiegelhalter and C.Taylor, *Machine Learning, Neural and Statistical Classification* (Ellis Horwood, Hertfordshire, England, 1994).
20. J.L.McClelland and D.E.Rumelhart, *Explorations in Parallel Distributed Processing* (MIT press,1988).