A Heuristic Knowledge Reduction Algorithm Based on Partition Subdivision and Consistent Degree

Wen Huo and Xiaoguang Hong

College of Computer Science and Technology,Shandong University,
hw112200@hotmail.com

Abstract: In this paper, a new knowledge reduction definition based on partition subdivision is proposed, its equivalence to the classic attribute reduction definition based on positive region is proved, and a consistent degree is introduced to evaluate the importance of condition attribute for decision attribute. Based on the above results, a heuristic knowledge reduction algorithm is designed.

Keywords: data mining, knowledge reduction, rough set, decision table, positive region, partition subdivision, consistent degree

1. Introduction

Knowledge reduction is one of the key problems of knowledge discovery in data mining. There are two classical definitions of knowledge reduction in rough set theory: one is based on positive region, the other is based on condition information entropy, but they are not equivalent when they deal with inconsistent decision table. In 2005 a definition based on the new condition information entropy was proposed [1], and its equivalence to the definition based on positive region was explained by a knowledge reduction algorithm. In 2006 a definition based on the average decision power was proposed [2]. Afterwards, the average decision power was amended to the decision power [3] in 2007, but the definition based on the decision power is not equivalent to the classical definition based on positive region.

Being illuminated by all the above research, this paper proposed a new knowledge reduction definition based on partition subdivision, and its equivalence to the classical definition based on positive region is proved. Furthermore, a consistent degree is introduced to evaluate the importance of condition attribute for decision attribute. Based on the above results, a heuristic knowledge reduction algorithm based on partition subdivision and consistent degree is designed.

2. Basic Notations and Definitions

[Definition 1][4,5] The notion of *information system* is formally defined as S=(U,A,V,f) ,and U,A,V,f is defined as follows: U: nonempty set of objects, called universe; A: nonempty set of attributes; $V=\bigcup_{a\in A} Va$, Va is the range of attribute a; $f:U\times A\to V$ is an information function, $\forall x\in U, a\in A, f(x,a)\in Va$.

$\forall P\subseteq A$, $IND(P)=\{(x,y)|(x,y)\in U\times U$ and $\forall a\in P, f(x,a)=f(y,a)\}$ is an equivalence relation on U, so U/IND(P), shortly written as U/P, forms an partition on U.

If attribute set A can be divided into condition attribute set C and decision attribute set D, namely, $C\cup D=A, C\cap D=\varnothing$, then S is termed decision table, shortly written as S=(U,C,D).

[Definition 2] [4,5] In decision table T=(U,C,D), $P\subseteq C\cup D$, $\forall X\subseteq U$, $\underline{P}X=\cup\{Y|Y\in U/P$ and $Y\subseteq X\}$ is termed the lower approximation of X. $POS_P(D) = \bigcup_{X\in U/D}\underline{P}X$ is termed the P positive region of D.

3. The Classical Knowledge Reduction Definition Based on Positive Region

[Definition 3] [4, 5] In decision table T=(U,C,D),$A\subseteq C$, if $POS_A(D)=POS_C(D)$, and $\forall a\in A$, $POS_{A-\{a\}}\neq POS_C(D)$, then A is termed a knowledge reduction of C with respect to D.

4. The Knowledge Reduction Definition Based on Partition Subdivision

[Definition 4][1] In decision table T=(U,C,D),$U/D=\{Y_1, Y_2,\ldots, Y_m\}$,$A\subseteq C$, let $Y_0=U-POS_A(D)$, then the set cluster $R_A=\{\underline{A}Y_0,\underline{A}Y_1,\underline{A}Y_2,\ldots,\underline{A}Y_m\}$ is termed a partition on U educed by A.

Definition 4 is explained as follows: $\underline{A}Y_0=Y_0=U-POS_A(D)$ can be proved easily, and $\bigcup_{i=1}^{m}\underline{A}Y_i=POS_A(D)$, if $\varnothing$ exists in R_A, then R_A is still a partition on U after delete the $\varnothing$, so we can suppose there is no $\varnothing$ in R_A.

[Definition 5] $U/P=\{P_1, P_2,\ldots, P_m\}$,$U/Q=\{Q_1, Q_2,\ldots, Q_n\}$, if $\forall Q_i\in U/Q, \exists P_k\in U/P, Q_i\subseteq P_k$, then we say U/Q is a partition subdivision of U/P.

Based on *Definition 4* and *Definition 5*, we propose a new knowledge reduction definition based on partition subdivision as follows:

[Definition 6] In decision table $T=(U,C,D)$,$U/D=\{Y_1,Y_2,\ldots,Y_m\}$, $R_C=\{\underline{C}Y_0,\underline{C}Y_1,\underline{C}Y_2,\ldots,\underline{C}Y_m\}$, $A \subseteq C$, if U/A is a partition subdivision of R_C, and $\forall a \in A$,$U/(A-\{a\})$ is not the partition subdivision of R_C, then A is termed a knowledge reduction of C with respect to D.

5 The Equivalence between Definition 6 and Definition 3

[Lemma 1][1] In decision table $T=(U,C,D)$,$U/D=\{Y_1, Y_2,\ldots, Y_m\}$,$A \subseteq C$, then $POS_A(D)=POS_C(D) \Leftrightarrow \underline{A}Y_i=\underline{C}Y_i$, $\forall i \in \{0,1,2,\ldots,m\}$.

[Lemma 2] In decision table $T=(U,C,D)$,$A \subseteq C$,$R_A=\{\underline{A}Y_0,\underline{A}Y_1,\underline{A}Y_2,\ldots,\underline{A}Y_m\}$, then U/A is a partition subdivision of R_A.

Proof：Suppose $U/A=\{A1, A2,\ldots, A n\}$,$U/D=\{Y_1, Y_2,..., Y_m\}$, because of $\underline{A}Y_j=\cup\{A_i|A_i \subseteq Y_j\}$, j=1,2…,m, some partition blocks should be a subdivision of $\{\underline{A}Y_0,\underline{A}Y_1,\underline{A}Y_2,\ldots,\underline{A}Y_m\}$, and the other partition blocks should be a subdivision of $\underline{A}Y_0=U-POS_A(D)$, so U/A is a subdivision of R_A.

[Theorem 1] In decision table $T=(U,C,D)$, $U/D=\{Y_1, Y_2,\ldots, Y_m\}$, $A \subseteq C$, $R_C=\{\underline{C}Y_0,\underline{C}Y_1,\underline{C}Y_2,\ldots,\underline{C}Y_m\}$, then $POS_A(D)=POS_C(D) \Leftrightarrow$ U/A is a partition subdivision of R_C.

Proof: 1) First, we prove "$\Rightarrow$": If $POS_A(D)=POS_C(D)$, then according to *Lemma 1* we can get $\underline{A}Y_i=\underline{C}Y_i$, $\forall i \in \{0,1,2,\ldots,m\}$, namely, $R_A=R_C$. Moreover, U/A is a partition subdivision of R_A according to *Lemma 2*, so U/A is a partition subdivision of R_C too.

2) Second, we prove "$\Leftarrow$": If U/A is a partition subdivision of R_C, let $U/A=\{A_1, A_2,\ldots, A_n\}$, and

$\bigcup_{i=1}^{k_1} A_i = \underline{C}Y_0$, $\bigcup_{i=k_1+1}^{k_2} A_i = \underline{C}Y_1,\ldots,$ $\bigcup_{i=k_m+1}^{n} A_i = \underline{C}Y_m$.(*) Then we can get the following conclusions:

Conclusion 1: $\forall i \in \{k_1+1,k_1+2,\ldots,n\}, \exists j \in \{1,2,\ldots,m\}, A_i \subseteq Y_j$.

Conclusion 2: $\forall i \in \{1,2,\ldots, k_1\}, \forall j \in \{1,2,\ldots,m\}, A_i \not\subset Y_j$.

Proof of Conclusion 1: For $\forall i \in \{k_1+1,k_1+2,\ldots,n\}$,according to (*) we can get that $\exists j \in \{1,2,\ldots,m\}$, $A_i \subseteq \underline{C}Y_j$, because of $\underline{C}Y_j \subseteq Y_j$, so $A_i \subseteq Y_j$.

Proof of Conclusion 2: Suppose $U/C=\{C_1,C_2,\ldots, Cs\}$, $\underline{C}Y_0=\bigcup_{k=1}^{q} C_k$, $q \le s$, and $\forall k \in \{1,2,\ldots,q\}, \forall j \in \{1,2,\ldots,m\}, C_k \not\subset Y_j$ (**), thus $\bigcup_{i=1}^{k_1} A_i = \bigcup_{k=1}^{q} C_k$. Because of U/C is a partition subdivision of U/A [7],$\{C_1,C_2,\ldots,C_q\}$ is definitely a partition subdivision of$\{A_1,A_2,\ldots,A_{k1}\}$,so $\forall$ i $\in$ $\{1,2,\ldots, k_1\}$, A_i is the union of some $C_k(k\in\{1,2,\ldots,q\})$, then we can get that $\forall$ j $\in$ $\{1,2,\ldots,m\}$, $A_i \not\subseteq Y_j$ due to (**), otherwise, suppose that $\exists$ j $\in$ $\{1,2,\ldots,m\}$， $A_i \subseteq Y_j$, thus $C_k \subseteq Y_j$, which is inconsistent with (**).

According to *Conclusion 1*, *Conclusion 2* and the definition of *A positive region of D*, we can get that

$$U - POS_A(D) = \bigcup_{i=1}^{k_1} A_i = \underline{C}Y_0 = U - POS_C(D),\ \text{namely, } POS_A(D) = POS_C(D).$$

From *Theorem 1* we can easily get the conclusion that *Definition 6* is equivalent to *Definition 3*.

Now we validate it by the following examples [2]:

Table 1. Decision Table1.

U	a	b	c	d
1	1	0	0	1
2	2	1	1	3
3	3	1	2	0
4	3	1	2	0
5	3	1	2	1
6	3	1	2	1
7	3	1	1	0
8	3	1	1	0
9	3	1	1	1
10	3	1	1	1

Table 2. Decision Table2.

U	a	b	c	d
1	1	1	0	1
2	3	1	2	0
3	3	1	2	0
4	3	1	2	0
5	3	1	2	0
6	3	1	2	1
7	3	3	0	0
8	3	3	0	0
9	3	3	0	0
10	3	3	0	1

Table 3. Decision Table3.

U	a	b	c	d
1	1	1	0	1
2	3	1	2	0
3	3	1	2	0
4	3	1	2	0
5	3	1	2	0
6	3	1	2	3
7	3	3	0	2
8	3	3	0	2
9	3	3	0	3
10	3	3	0	3

Table 1, *Table 2*and *Table 3* are all inconsistent decision table, we reduce the condition attribute set by Definition 3 and Definition 6, the reductive results list in the following table:

Table 4. Reductive Results of *Table 1, Table 2* and *Table 3*.

	Table 1	Table 2	Table 3
Definition 3	{a}	{a},{b,c}	{a},{b,c}
Definition 6	{a}	{a},{b,c}	{a},{b,c}

If we restrict to get the minimal reduction, which contains attributes the least, then {b,c} should be removed.

6. The Consistent Degree of Condition Attribute Subset Relative to R_C

From *Theorem 1* and *Definition 6* we can get that if condition attribute subset A and B are not reduction, then they are not the partition subdivision of R_C, now, how can we decide which one between A and B is more important for decision attribute? Therefore, we introduce the following definition to evaluate the importance of condition attribute subset for decision attribute.

[Definition 7]In decision table T=(U,C,D),A $\subseteq$ C,U/D={Y_1, Y_2,…,Y_m}, U/A={A_1, A_2,…, A_n},Y_0=U－$POS_C(D)$, R_C ={$\underline{C}Y_0$,$\underline{C}Y_1$,$\underline{C}Y_2$,…,$\underline{C}Y_m$}, then $\sigma_A = \sum_{i=1}^{n}\sum_{j=0}^{m}\frac{|\underline{C}Y_j \cap A_i|}{|A_i|}\times\frac{|\underline{C}Y_j \cap A_i|}{|U|}$ is called *the consistent degree of A with respect to R_C.* (It is obvious that 0<σ_A≤1.)

Before we clarify the significance of consistent degree, we firstly prove that $\sigma_A = 1 \Leftrightarrow$ U/A is a partition subdivision of R_C.

[Lemma 3] let U/P={P_1, P_2,…, P_m}, U/Q={Q_1, Q_2,…, Q_n}, then $\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{|P_j \cap Q_i|}{|Q_i|}\times\frac{|P_j \cap Q_i|}{|U|} = 1 \Leftrightarrow$ U/Q is a partition subdivision of U/P.

Proof: 1) First, we prove "$\Leftarrow$": If U/Q is a partition subdivision of U/P, then $\forall Q_i \in U/Q, \exists P_k \in U/P, Q_i \subseteq P_k$, thus $P_k \cap Q_i = Q_i$, and for the other $P_j \in U/P$, $j \neq k$, $P_j \cap Q_i = \varnothing$. So

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{|P_j \cap Q_i|}{|Q_i|}\times\frac{|P_j \cap Q_i|}{|U|} = \sum_{i=1}^{n}\left(\frac{|P_k \cap Q_i|}{|Q_i|}\times\frac{|P_k \cap Q_i|}{|U|} + \sum_{j\neq k}\frac{|P_j \cap Q_i|}{|Q_i|}\times\frac{|P_j \cap Q_i|}{|U|}\right) = \sum_{i=1}^{n}\left(\frac{|Q_i|}{|U|}+0\right) = 1$$

2) Second, we prove "$\Rightarrow$": If U/Q is a partition subdivision of U/P, then $\exists Q_s \in U/Q$, that the elements of Q_s come from different P_j, suppose that there are x_1 elements come from P_{j1}, x_2 elements come from P_{j2},…, x_k elements come from P_{jk}, $x_1 + x_2 + ... + x_k = |Q_s|$, thus $\sum_{j=1}^{m}\frac{|P_j \cap Q_s|}{|Q_s|}\times\frac{|P_j \cap Q_s|}{|U|} = \frac{x_1^2 + x_2^2 + ... + x_k^2}{|Q_s|\times|U|}$, because of $x_1, x_2, ..., x_k$>0, it is obvious that $x_1^2 + x_2^2 + ... + x_k^2 < (x_1 + x_2 + ... + x_k)^2 = |Q_s|^2$, thus $\sum_{j=1}^{m}\frac{|P_j \cap Q_s|}{|Q_s|}\times\frac{|P_j \cap Q_s|}{|U|} < \frac{|Q_s|^2}{|Q_s|\times|U|} = \frac{|Q_s|}{|U|}$. Moreover, $\forall Q_i \in U/Q, |P_j \cap Q_i| \leq |Q_i|$, thus $\sum_{j=1}^{m}\frac{|P_j \cap Q_i|}{|Q_i|}\times\frac{|P_j \cap Q_i|}{|U|} \leq \frac{|Q_i|}{|U|}$. So $\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{|P_j \cap Q_i|}{|Q_i|}\times\frac{|P_j \cap Q_i|}{|U|} = \sum_{j=1}^{m}\frac{|P_j \cap Q_s|}{|Q_s|}\times\frac{|P_j \cap Q_s|}{|U|} + \sum_{i\neq s}\sum_{j=1}^{m}\frac{|P_j \cap Q_i|}{|Q_i|}\times\frac{|P_j \cap Q_i|}{|U|}$ $< \frac{|Q_s|}{|U|} + \sum_{i\neq s}\frac{|Q_i|}{|U|} < \sum_{i=1}^{n}\frac{|Q_i|}{|U|} = 1$.

According to *Lemma3* and *Definition 3* we can easily get the following theorem:

[Theorem 2] $\sigma_A = 1 \Leftrightarrow$ U/A is a partition subdivision of R_C.

Now we come to the significance of consistent degree σ_A : Rc={$\underline{C}Y_0$,$\underline{C}Y_1$,$\underline{C}Y_2$,…,$\underline{C}Y_m$} not only divide the consistent objects belong to different decision class Y_1,Y_2,…,Y_m into different partition blocks $\underline{C}Y_1$,$\underline{C}Y_2$,…,$\underline{C}Y_m$, but also put all the inconsistent objects in one partition block $\underline{C}Y_0$ .Therefore, when U/A is not a partition subdivision of R_C, it is positive that the consistent objects belong to different decision class are mixed in one partition block, or the consistent objects and the inconsistent objects are mixed in one partition block. From the proof of *Theorem 2* we can see that the more of these mixtures, the smaller of σ_A, and the less importance of A for decision attribute.

7. A Heuristic Knowledge Reduction Algorithm Based on Partition Subdivision and Consistent Degree

Searching for all reduction or minimal reduction of a decision table was already proved to be a NP-hard problem [6, 7]; therefore, making use of some heuristic information to reduce the searching space is the main idea in most of the algorithms, which can get a minimal reduction or a suboptimal reduction. An algorithm is designed in this paper as follows: Taking the consistent degree of every condition attribute with respect to R_C as the heuristic information, the condition attribute set is presented in the consistent degree's descending order, then we begin to search from the first attribute of this set until the searching result is a partition subdivision of R_C, which is the minimal reduction.

[Algorithm 1] Input：decision table T=(U,C,D)

Output：a minimal reduction of T

Initialize REDU=$\varnothing$.

①Compute U/C, U/D, R_C;

②for each Ci∈C, compute U/Ci，if U/Ci is a partition subdivision of R_C, then REDU={Ci} and go to ⑧;

If every U/Ci is not the partition subdivision R_C, then go to ③;

③for each Ci∈C, calculate the consistent degree σ_{C_i} , and C is presented in the consistent degree's descending order as C= {A1,A2,…,An};

④REDU={A1,A2}， i=2; If U/REDU is a partition subdivision of R_C, then go to⑧. Else, go to ⑤;

⑤i=i+1, REDU=REDU∪{Ai};

⑥If U/REDU is a partition subdivision of R_C, then go to ⑦. Else, go to ⑤;

⑦for(k=i-1;k>=1;k--)

If U/(REDU-{Ak}) is a partition subdivision of R_C, then REDU = REDU-{Ak} and go to ⑧。

⑧Output REDU。

Supplementary explanation: Compute U/C, U/D, U/Ci and U/REDU by Algorithm 1 in reference [10];Compute $POS_C(D)$ by Algorithm 2 in reference[11], so R_C is obtained simultaneously.

We analyze *the time complexity of Algorithm 1* as follows: The time complexity of ① is O(|C||U|)[10,11]. In the worst circumstance, the whole C is minimal reduction, then the time complexity of ② is $O(|C|^2|U|)$,③ is $O(|C|^2|U|)$, ④⑤⑥ is $O(|C|^2|U|)$, ⑦ is $O(|C|^2|U|)$.Therefore, the time complexity of *Algorithm 1* is $O(|C|^2|U|)$, which is lower than the time complexity $O(|C|^2|U||\log|U||)$ of Algorithm 2 in reference[1].

The advantage of Algorithm 1:

1) There is no computation for $CORE_D(C)$.
2) Judging if REDU is a subdivision of R_C instead of judging if $POS_{REDU}(D)=POS_C(D)$, the calculation amount shrink evidently.
3) Even in the worst circumstance we only calculate the importance of each single condition attribute for decision attribute, so the calculation amount is less than calculating the importance of some attributes' combination. The calculation of consistent degree is easier than the calculation of condition information entropy too.

Now we clarify *Algorithm 1* by the following example[1] :

Table 5. Decision Table 4.

U	a	b	c	e	f	d
1	0	0	0	0	1	0
2	0	1	1	1	0	1
3	1	1	0	1	1	1
4	0	1	1	1	0	0
5	0	0	1	0	1	0
6	1	1	0	1	0	1
7	0	1	1	1	1	1
8	1	1	1	0	1	1
9	1	1	0	1	1	0
10	0	1	1	1	1	0

①U/D={{1,4,5,9,10},{2,3,6,7,8}},U/C={{1},{2,4},{3,9},{5},{6},{7,10},{8}}, R_C ={{1,5},{6,8},{2,3,4,7,9,10}}.

②U/{a}={{1,2,4,5,7,10},{3,6,8,9}},U/{b}={{1,5},{2,3,4,6,7,8,9,10}}, U/{c}={{1,3,6,9},{2,4,5,7,8,10}},U/{e}={{1,5,8},{2,3,4,6,7,9,10}}, U/{f}={{2,4,6},{1,3,5,7,8,9,10}}.
None of them is the partition subdivision of R_C.

③ $\sigma_{\{a\}}$ =0.533, $\sigma_{\{b\}}$ =0.7, $\sigma_{\{c\}}$ =0.4, $\sigma_{\{e\}}$ =0.695, $\sigma_{\{f\}}$ =0.467, then C is presented in the consistent degree's descending order as C={b,e,a,f,c}.

④⑤⑥U/{b,e},U/{b,e,a} are not the partition subdivision of R_C, U/{b,e,a,f} is the partition subdivision of R_C, so REDU={b,e,a,f}.

⑦U/{b,e,f}, U/{b,a,f} are not the partition subdivision of R_C, U/{e,a,f} is the partition subdivision of R_C, so REDU={e,a,f}.

⑧ Output REDU={e,a,f}.

8 Experimental Results

We choose Decision table 4 in this paper and some decision tables in UCI machine learning database, and implemented Algorithm 1 in this paper and Algorithm 2 in reference[1] by Java language on our PC(Intel(R) Core(TM)2 2.33GHz, 1.96GB RAM,WINXP). The experimental results are as follows:

Table 6. Experimental Results Table.

Decision table	If it is a consistent decision table	The number of instances	The number of condition attributes before reduction	The number of condition attributes in the minimal reduction	Algorithm 1in this paper		Algorithm 2 in reference[1]	
					The number of condition attributes after reduction	Execution time /s	The number of condition attributes after reduction	Execution Time /s
Table 5	No	10	5	3	3	0.01	3	0.02
Voting-records	Yes	435	16	9	9	0.12	9	0.15
Tic-tac-toe	Yes	958	9	8	8	0.32	8	0.38
zoo	No	101	17	10	11	0.06	10	0.07
mushroom	Yes	8124	22	4	4	3.23	4	3.80
chess end-game	Yes	3196	36	29	29	2.73	29	3.09

From Table 6. we can see that the execution time of Algorithm 1 in this paper is less than that of Algorithm 2 in reference[1].

9. Conclusions

Being illuminated by the set cluster R_C [1] and the decision power[3], this paper has found and proved the following laws by *Theorem 1* and *Theorem 2* : In decision table T=(U,C,D), $A \subseteq C$, $POS_A(D)=POS_C(D) \Leftrightarrow$ U/A is a partition subdivision of $R_C \Leftrightarrow \sigma_A = 1$ (σ_A is the consistent degree of A with respect to R_C).

Consequently, a heuristic knowledge reduction algorithm, *Algorithm 1,* is designed. Making use of *Theorem 1*, this algorithm judges if REDU is a subdivision of R_C instead of judging if $POS_{REDU}(D)=POS_C(D)$, so the calculation amount shrink evidently. From the proof of *Theorem 2* we can see that the smaller of σ_A, the less importance of A for decision attribute, so it is rational that this algorithm takes the consistent degree σ_A as the heuristic information to reduce the searching space. And the calculation of consistent degree is easier than the calculation of condition information entropy. The time complexity of this algorithm is lower too.

Finally, The results of experiment show that this algorithm is more efficient than Algorithm 2 in reference[1] actually.

References

1. LIU Qi-he, LI Fan, MIN Fan, YE Mao, and YANG Guo-wei,"An Efficient Knowledge Reduction Algorithm Based on New Conditional Information Entropy", Vol.20, No.8, pp.878-882,Control and Decision, 2005.
2. JIANG Si-yu and LU Yan-sheng,"Two New Reduction Definitions of Decision Table", Vol.27, No.3, pp.512-515, Mini- Micro Systems,2006.
3. XU Zhang-yan, SONG Wei, YANG Bing-ru, GAO Jing and HOU Wei,"Note on 'Two New Reduction Definition of Decision Table' ", Vol.28 ,No.9, pp.1686-1689,Journal of Chinese Computer Systems, 2007.
4. WANG Guo-yin: Rough Set Theory and Knowledge Acquisition, Xi'an Jiaotong University Publishing Company, Xi'an2001.
5. LIU Qing: Rough sets and Rough Reasoning, Science Publishing Company, Beijing 2001
6. Hu X.H. and Nick C. "Learning in relational databases: A rough set approach", Vol.11, No.2, pp.323-338,International Journal of Computational Intelligence, 1995
7. XU Zhang-yan, LIU Zuo-peng, YANG Bing-ru and SONG Wei,"A Quick Attribute Reduction Algorithm with Complexity of max $(O(|C||U|), O(|C|^2|U/C|))$", Vol.20, No.3, pp.391-399,Chinese Journal of Computers, 2006．
8. Pawlak Z, "Rough sets", Vol.11, No.5, pp.341-356, International Journal of Computer and Information Science, 1982.
9. Pawlak Z: Rough Sets：Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers，Dordrecht,1991
10. Pawlak Z, Slowinski R, "Rough set approach to multi-attribute decision analysis", Vol.72, No.1, pp.443-459, European Journal of Operational Research, 1994.