International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

An Efficient Data Compress Algorithm and Data Mining

Yang Li[1]

[1]School of Electronic Engineering, Xidian University, Xi"an, 710126, China

Abstract: ***Large data size, bottleneck in the speed of the cloud uploads and program loads, redundant data, inaccurate data is the resent block towards data researches. in the paper, we propose a practical data compression algorithm, in order to merge the time information, delete redundant data, round the over precision, etc. aiming at the difficulty of data mining, we propose apriori algorithm and support vector machine (svm). in the light of insufficient data analyses and management, this paper proposes rough sets theory, contribute to computing data study. these numerical experiments show that the algorithms below can effectively accelerate data load speed, reduce data size and efficiently mining data.***

Keywords: Data compression, Apriori, Data mining, Rough sets, Support vector machine

1. Introduction

The result data of the mid-processing computation of vector form intrinsic finite element(VFIFE)[1-4] analysis has a large size. That is a considerable bottleneck in the speed of the Cloud uploading and program loading, which should be highly compressed. The behavior data of the vector form intrinsic finite element is studied. After analyzing the characteristics of high redundancy。invalid data and numerical notation format of the data, a practical compression algorithm[5-9] is proposed to merge the time information, delete redundant data, round the over precision and use the line element incremental expression. The numerical experiments indicate that, this algorithm highly compresses the behavior data size and accelerate the data load speed. It greatly improves the usage efficiency of the vector form intrinsic finite element behavior analysis software.

Rough sets theory is a soft computing tool to deal with vagueness and uncertainty. This paper introduces development, researches the data mining method based on rough sets theory-CRCG(Classification based on Rough sets and Concept Generalization), and applies CRCG to die design quality evaluation.

This paper proposes Support Vector Machine (SVM). The method can resolve the problem of imbalanced dataset and improve the classification performance of SVM. Experiment results with artificial dataset show the algorithm is effective for imbalanced dataset, especially for the minority class samples. Apriori algorithm in this paper based on Association Rules, once it came into common sight, it received wide attention and high academic recognition.

2. The Proposed Algorithm

2.1 Some Definitions

Behavior data node element (BDNE) is the basic unit of vector form intrinsic finite model, it represents one of the nodes of BDNE, and it is temporal geometrical model data. Assume there is one behavior data node element P, it consists time, ID and coordinate, etc. These constituent[10] denoted as $P\ (T,A,C_X,C_Y,C_Z)$, and $A=\{a_1,a_2,\cdots,\ \}$; $C=\{c_1,c_2,\cdots,\ \}$. Element a in ID set A of BDNE represents the node"s ID code. Mapping out 3-dimensional coordinate of element c_i in C, C_X,C_Y and C_Z denote nodes coordinate position on x, y, z axis.

BDNE mathematical model: The whole model consists m time informations, and it appears i different BDNEs. The length of n-line data is l_n bytes, and the data size is:

$$V_P=\sum_{n=1}^{mi} ln \qquad (1)$$

Every vector in set P is denoted by 24 bytes scientific notation, every two vectors are separated by one space symbol. Thus every data line is fixed length, that is $l_n=5\times24+5=125$ bytes.

Behavior data line element (BDLE) is another element of vector form intrinsic finite model, it represents bridge structure in this model, and it is temporal topology model dataset. Assume we have one behavior data line element E, it consists time, ID, etc. These constituent denoted as $E\ (T,B,A_1,\ A_2)$, and $B=\{\ b_1,b_2,\cdots,b_r\}$; $A_1\subseteq A$; $A_2\subseteq A$. Element b in ID-set B represents the ID code of the line element.

BDLE mathematical model: The whole model consists m time informations, it appears r different BDLEs, and there are j_t BDLEs at time t. The length of n-th data is l_n bytes, then the data size is:

$$V_l=\sum_{t=1}^{m}(\sum_{n=1}^{jt} ln),\ 0<j_t\leq r \qquad (2)$$

Every vector in set E is denoted by 24 bytes scientific notation, every two vectors are separated by one space symbol. Data length is $l_n=4\times24+4=100$ bytes, and the line element data size is $V_l=100\sum_{t=1}^{m} jt$, $0<j_t\leq r$; average data size is $\overline{Vl}=50rm$ bytes.

2.2 Comparison of compression rate between different algorithms:

Due to BDNE and BDLE compression algorithm is aimed at VFIFE behavior data compression, thus we choose Huffman, LZw, LZ77 universal algorithms to compare with the compression algorithms shown in this paper. Tab 1 is

based on bridge collapse model, comparing different compression rate.

Table 1: Comparison of compression rate between different algorithms

Document	*Original size/KB*	*Huffman algorithm/KB*	*LZW algorithm /KB*	*LZ77 algorithm /KB*	*BDNE algorithm /KB*	*BDLE algorithm /KB*
BridgeNode.txt	79839	28732	11849	13410	10076	
BrigdgeElement.txt	25960	7953	2089	2169		11.9
Compression rate/%		65.33	86.82	85.27	87.38	99.95

The results show that, BDNE and BDLE compression algorithms can highly and effectively compress data size.

2.3 APRORI Data Mining Algorithm

In the beginning, R·Agrawal, etc. worked in IBM suggest data mining based on association rules. In terms of the customers‟ commodity preference analyses, they figure out some of the merchandise always bought together. The analyses show that, eighty per cent customers must have bought mike if they buy bread and butter; sixty per cent customers would have bought printers after they bought computers. There are too many shopping examples based on association rules. Once this theory came into common sight, it received wide attention and high academic recognition.

2.4 Association Rules

We regarded all items as $I=\{i_1,i_2,\ldots,i_m\}$, defined I as the items set, called element $i_p(p=1,2,\ldots,m)$ in set I items. Assume affairs data samples, denoted as $T=\{t_1,t_2,\ldots,t_m\}$, and define T as affairs data base, where element $t_p=\{Tid_p.IS_p\}(p=1,2,\ldots,n)$ called record or affair, and Tid_p is affair code, $IS_p \subseteq I$ is set of items.

Assume we have two nonempty data sets, denoted as X,Y; satisfy $X \subseteq I$. $X \neq \varphi$, and $Y \subseteq I$, $Y \neq \varphi$, $X \cap Y = \varphi$, title form like $X \geq Y$ association rules, which means items set X lead to appearance of set Y, where „≥‟ is association, X is prerequisite, Y is result of rules.

Support: Assume we have a nonempty set of items, denoted as X, and $X \subseteq I$, $X \neq \varphi$. One of the items denoted as $t_p=\{Tid_p.IS_p\}$, if $X \subseteq ISP$, then t_p support X, define the support as $S_r(X)$, which denoted as $S(X)=\{t/t \in T, X \subseteq ISP\}$, S(X) is the number of X appearance times inT.

Frequent Item Sets: Assume we have a nonempty set of items, denoted as X, and $X \subseteq I$, $X \neq \varphi$. Preset MinSup as minimum support, if $S(X) \geq MinSup$, title X frequent item sets; if X consist K items, title X K-dimensional frequent item sets.

2.5 Apriori algorithm

The main idea of Apriori algorithm:

(1) Mining 1-dimensional frequent item set L_1, computing each item sets support, and figure out all items which greater than MinSup, make them up as 1-dimensional frequent item set.

(2) Circulate step (1), mining K-dimensional frequent item set L_k, ($k \in [2, m]$), until no frequent item sets appear.

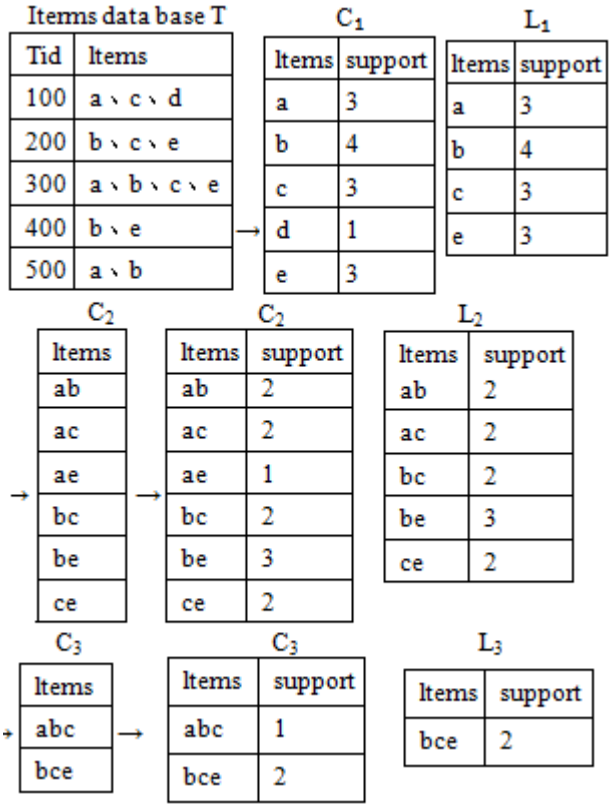

Figure 1: Apriori executive process

2.6 Algorithm based on Rough set theory

In 1970, famous savant Z. Pawlak in Poland and scientists in polish academy researched the logic characteristics of information system and these savants analyzed the inaccurate, uncertain, insufficient data. In 1982, Z.Pawlak first suggested Rough Sets[11] Theory in his paper. In 1991, monograph “Rough Sets” [12] published by Z.Pwalak is the milestone of rough set theory. In 1992, publication of handbook of application of the rough sets[13] theory made the conclusion of contemporary theory, and promoted computing study, knowledge, strategy analyses, process control, etc[14].

Rough sets theory is a soft computing tool to deal with vagueness and uncertainty. This paper introduces development, researches the data mining method based on rough sets theory-CRCG(Classification based on Rough sets and Concept Generalization), and applies CRCG to die design quality evaluation.

Definition of Rough Set is defined follows: We Given a data set $X \subseteq U$, equivalence relation is R. If X is some of basic

categories of R，denote X as R- definable，otherwise denote X as R- undefinable. R-definable set is called R-Accurate set，and can be accurately defined within database K, X is accurate set；R-undefinable set is called R-Rough set, it cannot be defined within database K, X is rough set.

CRCG algorithm: Data mining based on Classification based on Rough sets and Concept Generalization.Research of Classification based on Rough sets and Concept Generalization contain: classification, regression, gather, generalization. In this paper, we mainly consider about data classification.

In data classification, we separate all property into two groups, that is conditional property and strategic property, denoted as $c_1 \ldots c_n$ and $d_1 \ldots d_m$[15]. The rules of classification is:

IF $(c_1=I_1) \wedge \ldots \wedge (Cn = In)$ THEN $(d_1=J_1) \wedge \ldots \wedge (dm = Jm)$, where $I_i, J_j (i=1,2,\ldots,n; j=1,2,\ldots,m)$ denoted as sets.

Theorem 1: If rule "IF $(c_1=I_1) \wedge \ldots \wedge (Cn = In)$ THEN $(d_1=J_1) \wedge \ldots \wedge (dm = Jm)$" is true, for each k=1,2，…,n, in c_k corresponding concept generalization, if c_k is interior node, then there must be a child node C_k, to make rule "IF $(c_1=I_1) \wedge \ldots \wedge (Ck = Ik) \wedge \ldots \wedge (Cn = In)$ THEN $(d_1=J_1) \wedge \ldots \wedge (dm = Jm)$" prove true.

Theorem 2: If rule "IF $(c_1=I_1) \wedge \ldots \wedge (Cn = In)$ THEN $(d_1=J_1) \wedge \ldots \wedge (dm = Jm)$" is true, consider about strategic property, in d_k corresponding concept generalization, D_k is parent node of d_k, then the rule "IF $(c_1=I_1) \wedge \ldots \wedge (Cn = In)$ THEN $(d_1=J_1) \wedge \ldots \wedge (Dk = Jk) \wedge \ldots \wedge (dm = Jm)$" prove true.

Allowing for all above analyses, aiming at huge, high-dimensional data base, we choose mining strategy below in order to quickly mine data:

CRCG（Classification based on Rough sets and Concept Generalization）
1)Select Task Relevant Dataset;
2)New Dataset=Transform(Dataset);
3)Rule Set=Rough It(New Dataset);
4)IF Rule Set Interested or at Prime Concept Level THEN Stop ELSE Go to Step 2.

Algorithm introduction:
1)In terms of mining demand, choose correlative data sets;
2)In terms of corresponding concept generalization, transform correlative data sets into designated level concept data;
3)*Rough It* consists three constituents: Brief of attribute; Brief of element; Derivation rules.

3. Imbalanced data algorithm based on Support Vector Machine(SVM)

In order to reduce the effect of imbalanced dataset on Support Vector Machine(SVM)[15] classification performance, a new under -sampling algorithm[16-18] based on the twice support vector machine is proposed for imbalanced data classification. For samples of majority class, this algorithm deletes the samples far from the classification hyperplane. And for samples of minority class, this algorithm use over -sampling algorithm[19-20] to add new samples.

3.1 Support Vector Machine(SVM)

SVM algorithm
Given a data sample:

$$T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_i,y_i)\}, x_i \in R^n, y_i \in \{1, -1\}$$

SVM mainly propose to construct a classification hyper plane to cut apart two different samples, in order to figure out the maximum classification interval and keep the minimum error rate. We can use formula (1) to figure out decision function:

$$\min \varphi(\omega) = \frac{1}{2}<\omega, \omega> c\sum_{i=1}^{l} \varepsilon_i$$
$$s.t.\ y_i(<w,x_i>+b) \geq 1-\varepsilon_i,\ \varepsilon_i \geq 0,\ i=1,2,\cdots,l \quad (3)$$

By using Lagrange operator can we get the pairing question of question (3):

$$maxW(a) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} * \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_i K(x_i y_i)$$
$$s.t.\ \sum_{i=1}^{l} \alpha_i y_i = 0,\ 0 \leq \alpha_i \leq c,\ i=1,2,\cdots,l \quad (4)$$

Where $K(x_i y_i)$ is kernel function, $K(x_i, y_i)=<\Phi(x_i), \Phi(x_j)>$ use non linear mapping $\varphi: R^k \rightarrow F$ to map training sample from input space to other feature space. Finally, we can obtain decision function:

$$f(x)=sgn(\sum_{x \in SVs} \alpha_i y_i K(x_i,x)+b) \quad (5)$$

3.2 Distance from point to hyperplane

Distance from sample x to classification hyperplane:

$$d(x)=\frac{\omega}{\|\omega\|(x-x0)} \quad (6)$$

Where x_0 is mapping of sample x, ω is normal vector of hyperplane, $\|\omega\|$ is the second order norm of ω.

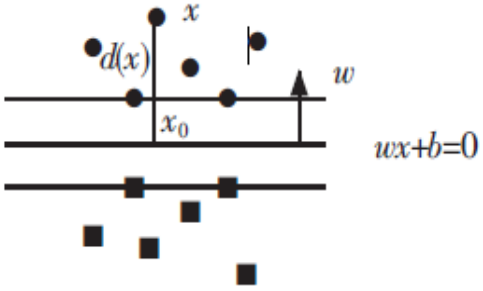

Figure 2: Distance from point to plane

Formula (6) can be transformed:

$$d(x)=\frac{\omega}{\|\omega\|}(x-x_0)=\frac{1}{\|\omega\|}(\omega x-\omega x_0)=\frac{1}{\|\omega\|}[(\omega x+b)-\omega x_0+b)] \quad (7)$$

Which satisfy $f(x_0)=\omega x_0+b=0$, then formula (5):

$$d(x)=\frac{\omega x+b}{\|\omega\|} \quad (9)$$

In terms of SVM derivation process, we can figure out that $\omega = \sum_{x \in SV} y_i a_i x_i$; Thus, for linear separable question, the distance from sample x to hyperplane is:

$$d(x) = \frac{\sum_{x \in SV} yiaiK(xi,\ x)}{\|\omega\|} \quad (10)$$

$K(xi, x)$ is kernel function $K(x_i, y_i)=<\Phi(x_i), \Phi(x_j)>$.
Distance from class to classification hyperplane $D(c_i)$ is distance from class c_i to classification hyperplane:

$$D(c_i)=\frac{1}{ni}\sum_{x\in C} d(x_j) \quad (11)$$

DSVM Under-Sampling Algorithm

There is a large amount of redundant information or helpless information for classification in most of the samples（such as samples far from classification hyperplane）. These redundant information lead to the imbalance of the training samples, and then affect the final classification performance of the separator. So a common method is to remove these redundant information by using a certain strategy, that is under-sampling, such as DROP, CNN, clustering algorithm. However, these methods delete some boundary samples as well. In this paper, we propose under-sampling algorithm based on sample-classification hyperplane, the algorithm is described below.

Step	Under-sampling algorithm based on sample-classification hyperplane
1	Aiming at training data set T, trained by using Support Vector Machine(SVM), and get classification hyperplane f(x), normal vector ω, support vector set *SV*, and corresponding coefficient of each *SV*.
2	In terms of formula (9) and (10): Linear separable question use (9); Linear inseparable question use (10); and compute the distance $d(x_i)$ between samples and classification hyperplane.
3	Compute the distance $D(c_i)$ between class and classification hyperplane by using formula (11).
4	For most classes of samples, in terms of given control parameters a, delete sample point $d(x)>a*D(c_i)$, and get new training set T'.
5	Train T′, if the classification effect is reached to the ideal state, then we can get final classification hyperplane and decision function; Otherwise, reset control parameters a, back to step 4.
6	Use interpolation method, increase samples.

Control parameters a is used to control delete the proportion of most samples, its value determined in terms of the ratio of minority samples amount and majority samples amount. That is $a=\frac{kni}{nj}$, where n_i is minority samples amount, n_j is majority samples amount, k is constant.

Conclusion and Experiment

Because of the random database is fortuitously emerged, so we experiment ten times in order to text the result. Tab 2 shows the results of ten experiments, the numbers in Tab2 is accuracy rate of experiments. Fig 3 shows the average accuracy rate of experiments while the kernel function is polynomial kernel function.

Table 2: Results of experiments DSVM and SVM

frequency/ times	*K=5.5,a=1.8*		*K=4.5,a=1.8*		*K=4.0,a=1.4*		*K=5.0,a=1.0*	
	DSVM	SVM	DSVM	SVM	DSVM	SVM	DSVM	SVM
1	93	85	93	83	91	86	97	94
2	95	87	94	88	91	86	95	89
3	92	88	95	90	90	85	97	84
4	98	90	95	89	93	88	93	86
5	93	89	91	86	90	84	94	87
6	97	82	95	88	97	92	96	92
7	99	94	95	90	93	83	98	94
8	98	95	95	91	93	89	86	81
9	97	90	97	82	97	87	98	91
10	92	89	92	83	91	88	98	89

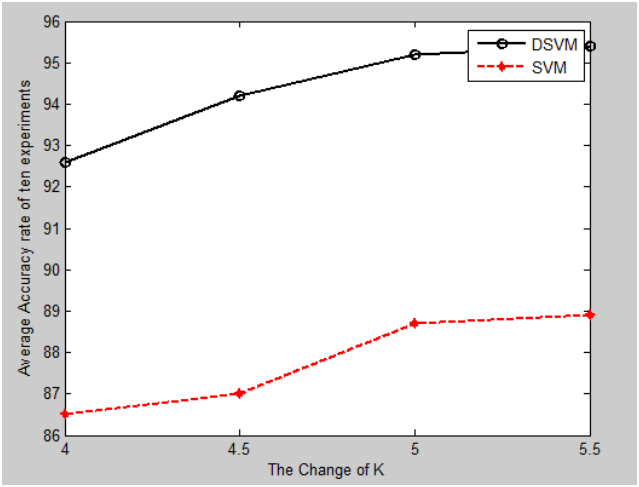

Figure 3: Average Accuracy rate of ten experiments

From Tab 2, we can propose that with the decrease of K, the amount of majority class samples decrease the meanwhile. Where K=0, majority class sample bit drop-out, the accuracy rate drops either. Where K=3, accuracy rate drops twice in ten times.

We used Fuzzy C-means Algorithm(FCM), etc. compared with our algorithm. The algorithm is shown in Tab 3.

Table 3: The FCM Algorithm

FCM Algorithm. Fuzzy C-means Algorithm (FCM)
1. Input: Given the dataset desire number of clusters, fuzzy parameters and stopping condition.
2. Calculate the cluster centroids and the objective value.
3. Compute the membership values stored in the matrix.
4. If the objective value of between consecutive iterations is less than the stopping condition ,then stop=true.
5. While(!stop).
6. Output:A list of c clusters and a partition matrix are produced.

In Tab 4, we list out quantitative and numerical evaluation for amount of algorithms.

Table 4: Quantitative and numerical Evaluation For Our Algorithm

CA				ARI				RI			
Ours	DEN	FCM	EM	Ours	DEN	FCM	EM	Ours	DEN	FCM	EM
82.13	66.13	71.78	79.85	77.91	58.85	62.63	72.02	82.97	77.15	79.36	81.76
81.72	66.43	70.67	80.13	73.79	55.36	60.79	71.44	88.75	76.29	81.11	84.07
79.85	63.29	77.66	78.26	72.77	53.01	66.25	70.33	83.98	76.56	80.29	81.14
80.12	67.53	73.73	82.33	69.63	59.37	63.10	66.98	80.11	76.57	73.29	79.26
87.26	64.66	79.77	77.01	78.99	61.23	62.15	76.31	81.11	74.46	76.13	80.08
86.73	64.45	70.76	79.73	74.71	50.93	62.97	73.07	80.66	79.29	76.12	75.31
81.18	68.98	74.10	80.92	71.76	55.47	62.60	70.56	84.62	75.50	81.80	79.73
88.33	64.62	77.36	85.98	76.23	53.10	70.21	75.23	86.51	81.21	82.33	84.04
88.92	66.86	77.45	87.22	79.46	51.21	70.15	77.12	88.51	86.22	80.11	85.68
83.94	69.94	74.31	80.69	72.92	50.35	69.07	71.81	87.76	80.22	86.13	82.75

Fig 4 shows the numerical evaluation for four algorithms in a comprehensive index. We can figure out that our algorithm proposed in this paper can highly and effectively execute data processing.

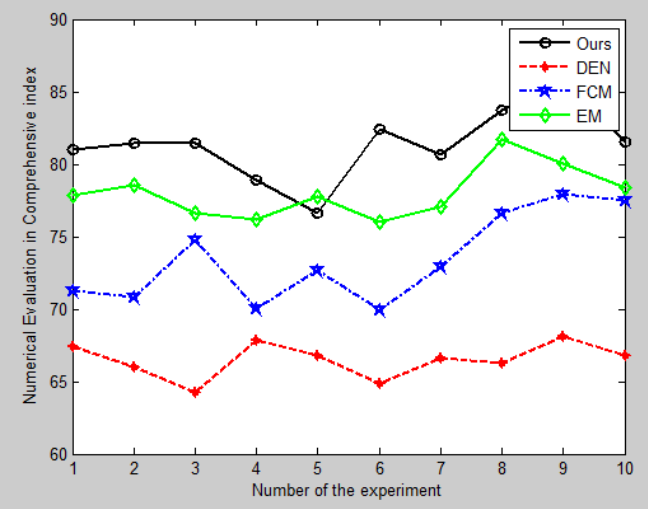

Figure 4: Numerical Evaluation for four Algorithms in Comprehensive index

References

[1] LU Zhegang, YAO Jian. Vector form intrinsic finite element-a new numerical method[J]. Spatial Structures,2012,18(1):85.
[2] Ting E C, Shih C, Wang Y K. Fundamentals of a vector form intrinsic finite element: part I.basic procedure and a plane frame element[J]; Journal of Mechanics,2004,20(2):133.
[3] Ting E C, Shih C, Wang Y K. Fundamentals of a vector form intrinsic finite element: part II. plane solid elements[J]. Journal of Mechanics,2004,20(2):123.
[4] YU Dan. An improved Thomas algorithm for linear finite element parallel computing[J]. Public Communication of Science,2013(1):112.
[5] LONG Haijun, YAO Guanglin. Study of compressing algorithm on historical data in NPP DCS[J]. Science and Technology Information,2013(8):29.
[6] LI Shuai, FANG Yuanmin, XI Wenfei. Rapid compression algorithm based on the no topological vector curve[J]. Science and Technology and Engineering,2011,11(18):4324.
[7] FU Peng. Compression algorithm for triangle meshes[D]: Hangzhou: Zhejiang University,2009.
[8] LIU Ying, HAN Zhongming, CHEN Yi. Connectivity compression for non-triangular meshes by context-based arithmetic coding[J]. Computer Engineering and Applications,2010,46(22):178.
[9] GU Tianlong. Formal methods of software development[M]. Beijing:China Higher Education Press,2005.
[10] Pawlak Z. Rough sets. International journal of information and computer science,1982,11(5)
[11] Pawlak Z. Rough sets. Theoretical journal of reasoning about data, Dordrecht. The Netherlands: Kluwer Academic Publishers,1991,1-168.
[12] Slowiski R, Intelligent decision support; Handbook of application of the rough sets theory. The Netherlands: Kluwer Academic Publishers,1992,1-235.
[13] WANG yu, MIAO duolian, ZHOU youjian. Rough sets theory and applications[J]. Pattern Identification and Intelligence,1996,9(4):337-344
[14] Vapnik V. The nature of statistical learning theory[M]. New York：Springer-verlag,1995
[15] He H, Garcia E A. Learning from imbalanced data[J]. IEEE Transactions on Knowledge and Data Engineering, 2009,21（9）：1263-1284
[16] Chen B, Ma L, Hu J. An improved multi-label classification method based on SVM with delicate decision boundary[J]. International Journal of Innovative Computing, Information and Control, 2010,(6):1605-1614
[17] Wang B X, Japkowicz N. Boosting support vector machines for imbalanced datasets[J]. Lecture Notes in Computer Science, 2008,4994:38-47
[18] Plawlak Z. Vagueness and uncertainty-a rough set perspective. Computational Intelligence. 1995,11(2):227-232.
[19] J.Han, et al. Generalization-based data mining in object-oriented databases using an object cube model, data & Knowledge Engineering,1998,25:55-97.