International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 16 (2015) pp 36786-36790
© Research India Publications. http://www.ripublication.com

Classifier Models for Discrete Information Retrieval

A.Vinothkumar[1], Dr. M. Anand[2] and Dr. S.Ravi[3]

[1]Research Scholar, ECE Department, Dr.M.G.R.Educational and Research Institute, Chennai
Email: avinothme@gmail.com

[2]Professor , ECE Department, Dr.M.G.R.Educational and Research Institute, Chennai
Email: harshini.anand@gmail.com

[3]Professor & Head, ECE Department, Dr.M.G.R.Educational and Research Institute, Chennai
Email:ravi_mls@yahoo.com

Abstract- ***The thought that classifiers using machines could compose information is analogous to the human brain model. But, this could be possible with statistical and information theoretic models. The effect of classifier on two types of data generated models i.e. stochastic (probability based for ex: dice) and deterministic models i.e. the use of familiar sequences to generate information is presented in this paper. The classifier performance is studied with respect to varying signal to noise ratio values for different metrics.***

Keywords: Information Retrieval, Coded information, Scalable Support Vector Machine, Simple Integrated Modular Structures

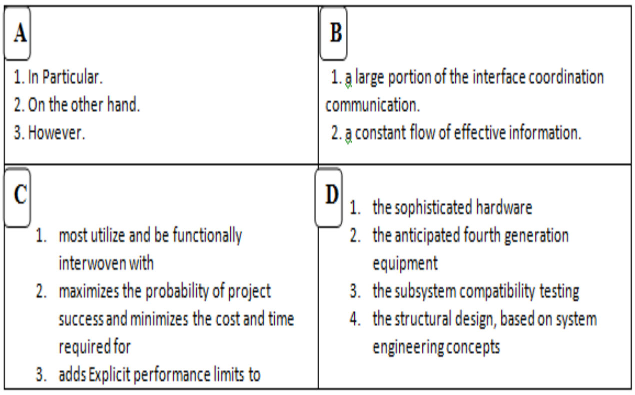
Introduction

Information is a combination of a finite number of discrete (expressible by a sequence of digits) and could represent music, a poem, a painting and so on. In coded form, a long string of digits can express each one of them distinctly. The only limitation is whether existing search algorithms are efficient to index such a large coded database. The advantage with coded information is it permits data manipulation and creation of new forms of the uncoded data without even the basic knowledge of music or poetry or painting. For eg: a painting can be coded into a matrix of minute cells on a base canvas and the code representing the precise color in each cell of the matrix. Thus, the coded information allows (i) scanning (ii) recreation (iii) generate new paintings with zero knowledge of painting. The same can be extended to symphony information which is hybrid data i.e. both continuous and discrete. However, as a discrete chain, the symphony is just a coded curve. In this paper, classifiers that use a set of combinatorial rules and perform recreation and generation of source information from coded discrete chains are discussed. The work helps to obtain significant new knowledge in almost every field simply by exploring all combinations of a small number of basic elements. Thus, a mixture of predictable patterns and elements of surprise can be achieved in this process.

Building Information Using Simple Integrated Modular Structures (SIMS)

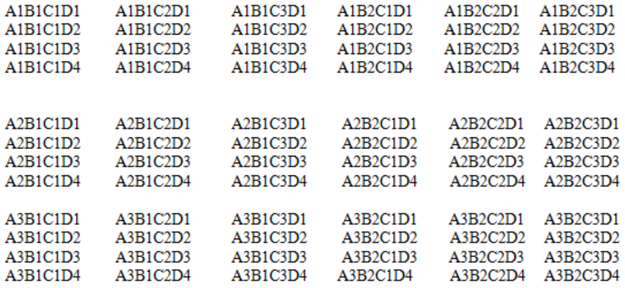
Information can be built using Simple Integrated Modular Structures (SIMS). Different base sets are constructed in such a way that any element of a set goes in conjunction with any element of the neighboring set when arranged sequentially. Example information constructed with four sets A, B, C and D for variable number of elements in each set is shown in Table 1. The size of each set could be non uniform. In Table 1, size (A) =3; size (B) =2; size(C) =3; size (D) =4. The number of distinct non-overlapping information that can be generated using this base sets are listed in table 2. For the sample size of S (A, B, C, D) = (3, 2, 3, 4) this is equal to seventy two i.e. 3x2x3x4. From Table 2, thus the information corresponding to A1B2C2D1 is "In particular, a constant flow of effective information adds explicit performance limits on the sophisticated hardware".

Table 1: Showing set A, B, C and D in SIMS



A 1. In Particular. 2. On the other hand. 3. However.	B 1. a large portion of the interface coordination communication. 2. a constant flow of effective information.
C 1. most utilize and be functionally interwoven with 2. maximizes the probability of project success and minimizes the cost and time required for 3. adds Explicit performance limits to	D 1. the sophisticated hardware 2. the anticipated fourth generation equipment 3. the subsystem compatibility testing 4. the structural design, based on system engineering concepts

Table 2: Information generated using SIMS of Table 1



A1B1C1D1	A1B1C2D1	A1B1C3D1	A1B2C1D1	A1B2C2D1	A1B2C3D1
A1B1C1D2	A1B1C2D2	A1B1C3D2	A1B2C1D2	A1B2C2D2	A1B2C3D2
A1B1C1D3	A1B1C2D3	A1B1C3D3	A1B2C1D3	A1B2C2D3	A1B2C3D3
A1B1C1D4	A1B1C2D4	A1B1C3D4	A1B2C1D4	A1B2C2D4	A1B2C3D4
A2B1C1D1	A2B1C2D1	A2B1C3D1	A2B2C1D1	A2B2C2D1	A2B2C3D1
A2B1C1D2	A2B1C2D2	A2B1C3D2	A2B2C1D2	A2B2C2D2	A2B2C3D2
A2B1C1D3	A2B1C2D3	A2B1C3D3	A2B2C1D3	A2B2C2D3	A2B2C3D3
A2B1C1D4	A2B1C2D4	A2B1C3D4	A2B2C1D4	A2B2C2D4	A2B2C3D4
A3B1C1D1	A3B1C2D1	A3B1C3D1	A3B2C1D1	A3B2C2D1	A3B2C3D1
A3B1C1D2	A3B1C2D2	A3B1C3D2	A3B2C1D2	A3B2C2D2	A3B2C3D2
A3B1C1D3	A3B1C2D3	A3B1C3D3	A3B2C1D3	A3B2C2D3	A3B2C3D3
A3B1C1D4	A3B1C2D4	A3B1C3D4	A3B2C1D4	A3B2C2D4	A3B2C3D4

The above methodology of coding can be easily extended to include emotions in addition to content in the coded information (for ex: the Parsons code for information storage of melody).

Objectives of This Work

- Retrieving information coded in the form described in section 1 of this paper can perform picture/audio based recognition and assist medical care, surveillance, process automation etc. The patient centered medical home is a way of organizing primary care that emphasizes care coordination and communication to transform primary care into "what patients want it to be." Medical homes can lead to higher quality and lower costs, and can improve patients' and providers' experience of care.

- To facilitate archiving for extended case based analysis
- To build modular algorithmic structures and provide ease of application extensibility
- Information indexing, retrieval and generation

Dimension Reduction for Classifier Design

Dimension reduction is an important step in information processing and classifier design. Too many feature data formed in the pre-classifier stage can increase the system storage and render the search algorithm useless. In this paper, dimensional reduction algorithms using three methods are presented namely;

I. Stepwise forward selection
II. Stepwise backward elimination and
III. Hybrid optimization method

I. Stepwise Forward Selection

In this method, the initial value of the dimension of the optimal feature set is chosen as null step by step, the essential features are added till a threshold is reached (dimension of the set is said to be converged). This is illustrated in the flowchart of Figure 1.

The basic concept of dimensional data reduction for the initial feature set is illustrated in Figure 2. High dimensional data can typically have many irrelevant dimensions. The data increasingly become sparse as the dimensionality increases. In such situations, the distance measurement between pairs of point become meaningless as the average density of point anywhere in the data is low. Hence, there is a need to optimize the dimensionality of feature set.

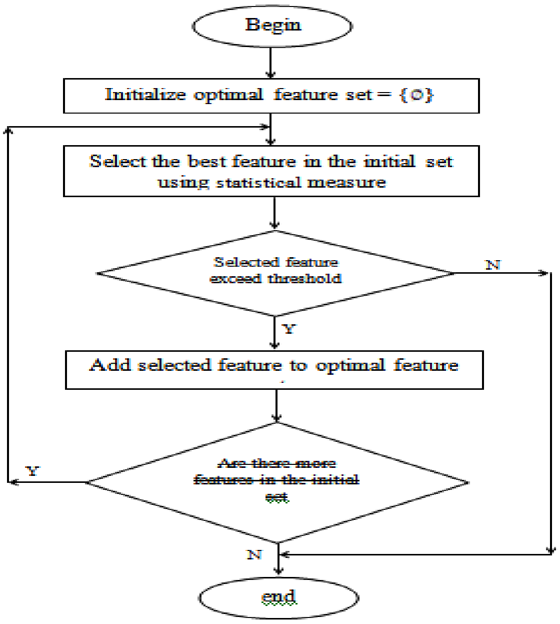

Begin
Initialize optimal feature set = {∅}
Select the best feature in the initial set using statistical measure
Selected feature exceed threshold
N
Y
Add selected feature to optimal feature
Are there more features in the initial set
Y
N
end

Figure 1: Flow chart for Stepwise Forward Selection

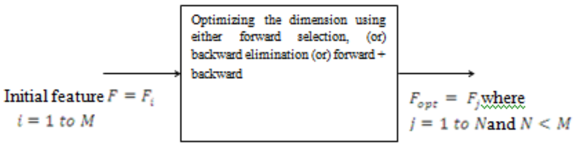

Optimizing the dimension using either forward selection, (or) backward elimination (or) forward + backward
Initial feature $F = F_i$
$i = 1\ to\ M$
$F_{opt} = F_j$ where
$j = 1\ to\ N$ and $N < M$

Figure 2: Dimensional Data Reduction for the Initial Feature Set

II. Backward Elimination

This differs from the stepwise forward selection in that the initial value of the optimal feature set dimension is set equal to the total number of features and in every step, the redundant features are eliminated. Flow chart for Stepwise backward Selection is shown in Figure 3.

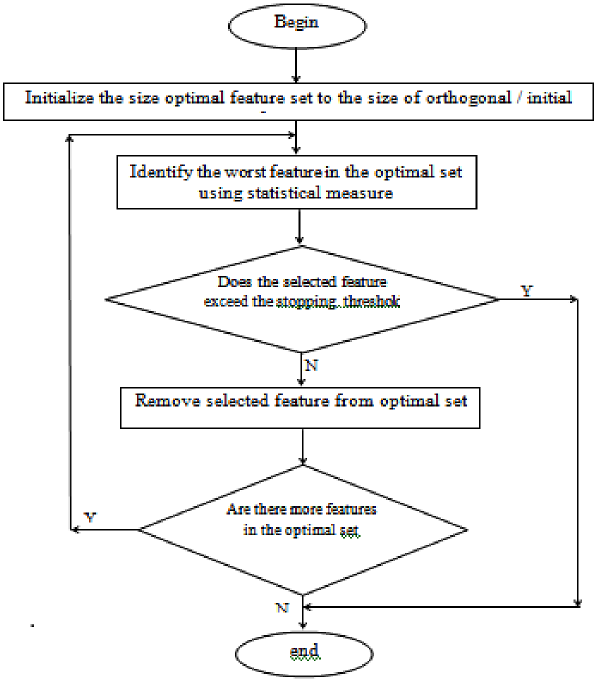

Begin
Initialize the size optimal feature set to the size of orthogonal / initial
Identify the worst feature in the optimal set using statistical measure
Does the selected feature exceed the stopping threshold
Y
N
Remove selected feature from optimal set
Are there more features in the optimal set
Y
N
end

Figure 3: Flow chart for Stepwise backward Selection

III.Hybrid Optimization Method

This method of dimensional data reduction is chosen in this work, and uses a combination of stepwise forward selection (step 1) and stepwise backward elimination (step 2) to form the optimal feature set. Flow chart for Hybrid Optimization Method is shown in Figure 4.

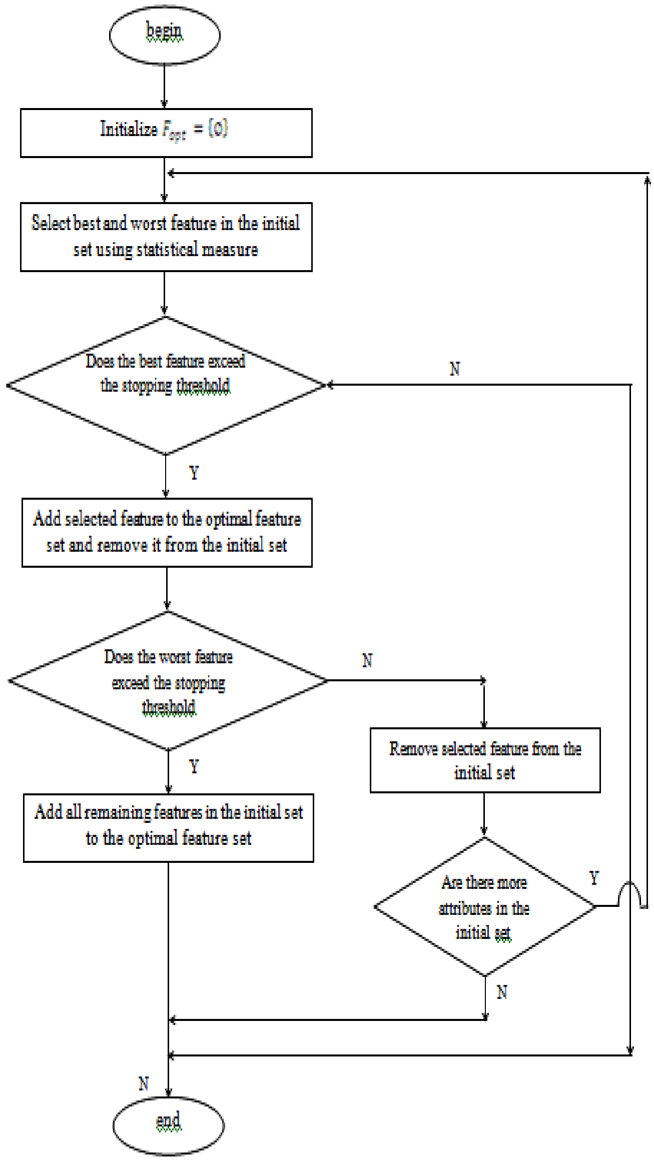

begin
Initialize F_{opt} = {0}
Select best and worst feature in the initial set using statistical measure
Does the best feature exceed the stopping threshold
N
Y
Add selected feature to the optimal feature set and remove it from the initial set
Does the worst feature exceed the stopping threshold
N
Remove selected feature from the initial set
Y
Add all remaining features in the initial set to the optimal feature set
Are there more attributes in the initial set
Y
N
N
end

Figure 4: Flow chart for Hybrid Optimization Method

Information Class Retrieval

Two types of classifiers (k-NN and scalable SVM) are presented in this work along with their comparison metrics. The implementation details for two classifiers are discussed later in the paper.

K-NN classifier

INPUT
Let 'U' → Unknown samples to be assigned different classes
Let 'T' → Training set containing the training samples

$$T_1 = \{t_{1,1}, t_{1,2}, \dots \dots t_{1,n}\}$$

$$T_2 = \{t_{2,1}, t_{2,2}, \dots \dots t_{2,n}\}$$

$$T_m = \{t_{m,1}, t_{m,2}, \dots \dots t_{m,n}\}$$

Let feature $t_{i,n}$ be the class label of T_i
Let 'm' → number of training samples
Let 'n' → number of features describing each sample
Let 'k' → number of nearest neighbors to the determined.

OUTPUT

Class label corresponding to 'U'

K-NN Algorithm

Step 1: Array a[m][2]
'M' represents the rows containing data regarding 'm' training samples. The first column represents the Euclidean distance between 'U' and that row's training samples. The second column refers to the Sample. The second column refers to that training samples index.
Note: the index needs to be saved, since when sorting the array (according to Euclidean distance), there need to be some method to determine to which training set the Euclidean distance refers.
Step 2:

$$for\ i = 1\ to\ m\ do\{$$

Step 3:

$$a[i][1] = eucllidean\_distance(U, T_i)$$

Step 4:

$$a[i][2] = i;$$

In this step the index is saved, as rows will be sorted later.
Step 5:
Sort the row of 'a' by their Euclidean distances saved in

$$a[i][1]$$

In this step sorting is done in ascending order
Step 6: Array

$$b[k][2];$$

The first column holds the distinct class labels of the k-nearest neighbours. The second column holds their respective counts. In the worst case, each k-nearest neighbour will have a different class label, hence there is a need to allocated space for k class label
Step 7:

$$for\ i = 1\ to\ k\ do\ \{$$

Step 8:
If class label $t_{a[i][2],n}$ already exists in array 'b' then perform step 9.
Step 9:
Find that class labels row in array 'b' and increment its count
Step 10:
else add the class label into the next available row of array 'b' and increment its count;}
Step 11:
Sort array 'b' in descending order
This sorting is done from class label with largest count down to that with smallest count.
Step 12:
Return $(b[1]);$
The most frequent class label of the k-nearest neighbors of 'U' is returned as the class prediction
Note: Euclidean distance is defined as

$$d(i,j) = \sqrt{(x_{i1}-x_{j1})^2 + (x_{i2}-x_{j2})^2 + \ldots\ldots + (x_{in}-x_{jn})^2}$$

Where $i = (x_{i1}, x_{i2}, \ldots x_{in})$ and $j = (x_{j1}, x_{j2}, \ldots x_{jn})$ are two n-dimensional data objects

Alternately, Manhattan distance defined as $d(i,j) = |x_{i1}-x_{j1}| + \cdots + |x_{in}-x_{jn}|$ (or)

Minkowski distance

$$d(i,j) = \left(|x_{i1}-x_{j1}|^q + |x_{i2}-x_{j2}|^q + \cdots + |x_{in}-x_{j1}|^p\right)^{1/p}$$

can be used.

Scalable SVM

In this research work, to overcome the limitation of conventional SVM classifiers (i.e. large training time particularly when data samples are large), a scalable SVM is used. The features of scalable SVM include;

(i) Using microclusters
(ii) Training the SVM on the centroids of the microclusters
(iii) Decluster entries near the boundary
(iv) Step (ii) and (iii) is repeated for additional entries
(v) Step (i) and (iv) is repeated till convergence

Steps for finding K nearest neighbors

Step 1:

For $i = 1$ to number of data classes do

Step 2:

Find the distances of the i^{th} object to all other objects

Step 3:

Sort these distances in descending order. In this step, which class is associated with each distance is tracked

Step 4:

Return the classes associated with the first 'k' distances of the sorted list

Step 5: End

Limitations to be Overcome in KNN

(i) Need to avoid dependency on the order of duplicate classes
(ii) Classifier time is high

Metrics for Class Retrieval

Accuracy, Precision, F-measure and recall are used as the metric in this work to assess the performance of the two classifiers used for the information class retrieval.

$$Accuracy = \frac{True\ Positive + True\ Negative}{Positive + Negative}$$

$$= \frac{True\ Positive}{denominator} + \frac{True\ Negative}{denominator}$$

$$= \frac{True\ Positive}{denominator} X \frac{Positive}{Positive} + \frac{True\ Negative}{denominator} X \frac{Negative}{Negative}$$

$$= Sensitivity\ X \frac{Positive}{True\ Positive + True\ Negative} + Sensitivity\ X \frac{Negative}{True\ Positive + True\ Negative}$$

Performance of the two classifiers

In this research, using z-statistic let, the SVM and KNN classifier is compared. For a given data set size 'N' let KNN_A& SVM_A be the accuracy of the respective classifiers. Then

$$Z = \frac{KNN_A - SVM_A}{\sqrt{\frac{2A(1-A)}{N}}} \quad \text{where } A = \frac{(KNN_A - SVM_A)}{2}$$

is calculated. The KNN classifier performs better than SVM classifier if Z >1.96

Results and Discussion

It is observed that at lower SNR values of information retrieval, the KNN classifier is better than SVM classifier. The classifier performance variation with SNR is given in figure 5. However, with coded information storage, where the SNR value has no effect, the SVM classifier outperforms K-NN classifier. Variation of Z-measure with SNR for K-NN and scalable SVM classifier is shown in Figure 5.

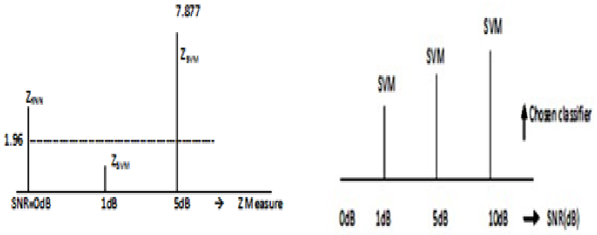


Figure 5: Variation of Z-measure with SNR for K-NN and scalable SVM classifier

Table 3: Summary of the Classifier Metrics Results with varying SNR

Classifier	SNR 0dB	SNR 1dB	SNR 5dB	Classifier Type	Information Class
Z-Measure For SVM	X	0.3186	7.877	SVM	Class-1
Z-Measure For KNN	2.3433	X	X	KNN	Class-1
Precision	X	X	0.6	KNN	Class-2
Recall			0.6696	KNN	Class-2
F-Measure			0.6328	KNN	Class-2
P			0.744	SVM	Class-2
R			0.788		Class-2
F			0.7653		Position-2

Discussion

It is inferred from figure 5 and table 3 that SVM classifier outperforms K-NN classifier for information class retrieval.

This implies that with SVM classifier uniform class level categorization is guaranteed. In coded information storage where SNR is the least affecting factor, SVM classifiers are far superior.

Conclusion

The analysis and algorithms presented in this work reveals that the deep structure of information must unfold linearly over time, and this puts pressure on composing the information. In this context, the classifier helps in presenting the form of the piece and balancing the fine line between expectation and surprise. Future direction of study shall include entropy based information class retrieval algorithms and comparison with the reported works.

References

[1]. AlexandrosKaratzoglou ,David Meyer ,Kurt Hornik, 2006, “Support Vector Machines in R”, Journal of Statistical Software, Vol. 15, Issue 9.

[2].Mohammad Faizal Ahmad Fauzi, 2009, “Optimal Discrete Wavelet Frames Features for Texture-Based Image Retrieval Applications”, Vol. 5857, pp 66-77.

[3].Juan Feng, Shengwei Wang, Gang Liu, LihuaZeng, 2012, “A Separating Method of Adjacent Apples Based on Machine Vision and Chain Code Information”, Vol.368, pp 258-267.

[4].S. Dobler, P. Meyer, H. W. Rühl, R. Collier, L. Vogten, K. Belhoula, 1992, “A Server for Area Code Information Based on Speech Recognition and Synthesis by Concept”, Konvens Informatikaktuell , pp 353-357.

[5].R. Eckhorn, O. -J. Grüsser, J. Kröller, K. Pellnitz, B. Pöpel,1976, “Efficiency of different neuronal codes: Information transfer calculations for three different neuronal systems”, Vol. 22, Issue 1, pp 49-60.

[6].Alberto Costa, Massimo Melucci, 2010, “An Information Retrieval Model Based on Discrete Fourier Transform”, Vol. 6107, pp 84-99.

[7].H. Doove, 1986, “Aids for Reading and for the Interaction with Coded Information Sources”, Vol. 47, pp 83-86.

[8].Bruce R. Schatz, 1997, “Information Retrieval in Digital Libraries: Bringing Search to the Net”, Science, Vol. 275

[9].Richard J. Lamberski, Francis M. Dwyer, 1983, “The instructional effect of coding (color and black and white) on information acquisition and retrieval”, ECTJ, Vol. 31, Issue 1, pp 9-21

[10].Yongzhen Huang, Tieniu Tan, 2014, “Enhancement via Integrating High Order Coding Information”, Springer Briefs in Computer Science , pp 59-70

[11].LiaofuLuo, 2009, “Law of genome evolution direction: Coding information quantity grows”, Vol. 4, Issue 2, pp 241-251.

[12].Tridenski, S ,Zamir, R, Ingber, A.,2015, “The Ziv–Zakai–Rényi Bound for Joint Source-Channel Coding”, IEEE Xplore

[13].Shih-Fu Chang, Thomas S. Huang, Michael S. Lew, Bart Thomee,2015, “Special issue on concept detection with big data”, International Journal of Multimedia Information Retrieval, Vol. 4, Issue 2, pp 73-74.

[14].SzilárdVajda, Daekeun You, Sameer Antani, George Thoma, 2015, “Large image modality labeling initiative using semi-supervised and optimized clustering”, International journal of multimedia information retrieval Vol. 4, Issue 2, pp 143-151

[15]. Hwanjo Yu, Jiong Yang, Jiawei Han, Xiaolei Li, 2005, “Making SVMs Scalable to Large Data Sets using Hierarchical Cluster Indexing”, Submission to data mining and knowledge discovery: an international journal.