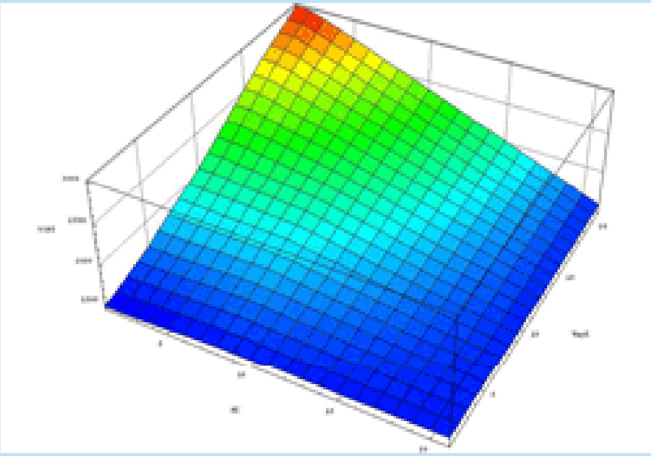
Ternary Decoding of Error Correcting Output Codes



Sergio Escalera

Oriol Pujol

Petia Radeva

Iberoamerican Congress on Pattern Recognition 2006

Index

1. ECOC
2. Ternary decoding
3. Results
4. Conclusions

ERROR CORRECTING OUTPUT CODES

A general framework for solving **multiclass** categorization problems.

Journal of Artificial Intelligence Research 2 (1995) 263-286 Submitted 8/94; published 1/95

Solving Multiclass Learning Problems via Error-Correcting Output Codes

Thomas G. Dietterich TGD@CS.ORST.EDU
Department of Computer Science, 303 Dearborn Hall
Oregon State University
Corvallis, OR 97331 USA

Ghulum Bakiri EB004@ISA.CC.UOB.BH
Department of Computer Science
University of Bahrain
Isa Town, Bahrain

Ensemble strategy based on the reduction of the multi-class problem in different sets of binary problems.

How are the sets defined?

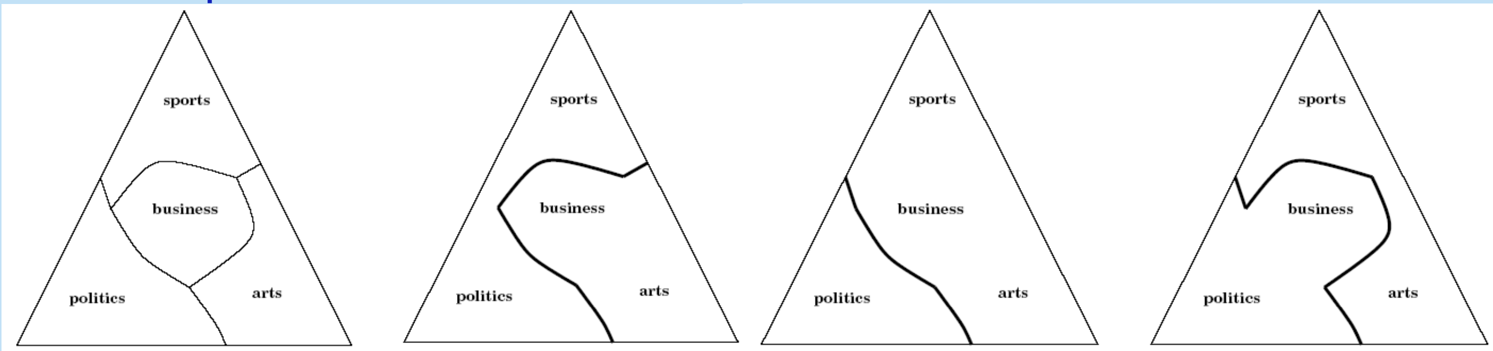
How are the classifiers combined?

It is a label perturbation technique that works in the following way:

Coding step: How many base classifiers? Which ones? Strategy to decompose a multiclass problem into complementary two “super-class” problems (a “super-class” a set of the original classes).

Decoding step: How do we decide the class of a new sample from the results of base classifiers? We expect that the decoding will be robust to error from learning algorithm, features and training samples.

Example | Classifier 1 | Classifier 2 | Classifier 3



C1= sports	1	1	1
C2=business	-1	1	-1
C3=politics	1	-1	-1
C4=arts	-1	1	1

coding matrix

code for class C4

Given a test sample we obtain a code according to the output of each classifier and find the “closest” code.

***X*=[-1 1 1]**

Standard strategies

Coding

One-vs-one

One-vs-all

Dense Random

Sparse Random

1 versus All

Code length: Nc

$$\begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix}$$

Ternary codes

1 versus 1: "All pairs"

Code length: Nc (Nc-1)/2

$$\begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & -1 \end{bmatrix}$$

Random Dense ECOC

Code length: 10 log Nc

$$\begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \end{bmatrix}$$

Random Sparse ECOC

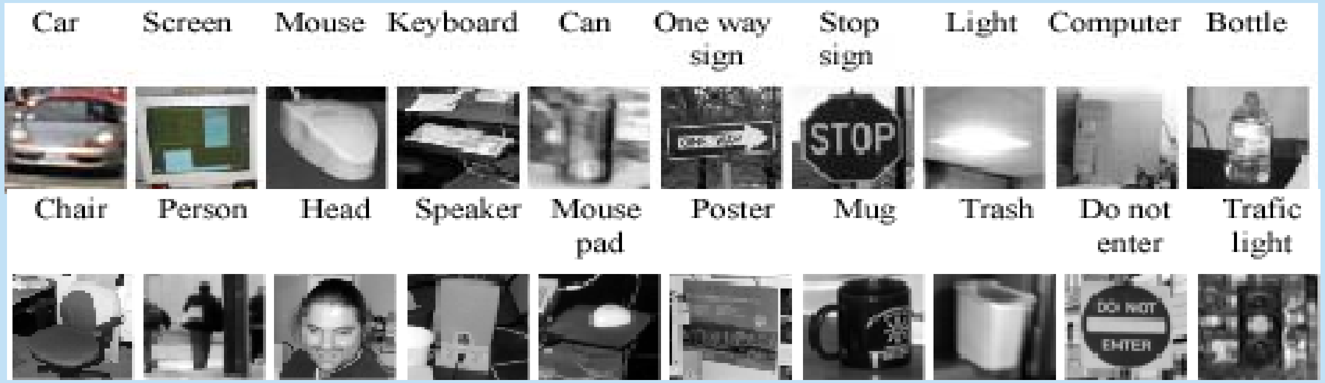
Code length: 15 log Nc

$$\begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}$$

Motivation

Many real problems involve a great number of classes.

•one-versus-all is the dominant strategy (e.g. shared boosting).



Question: how can we increase the technique performance while keeping the codeword length small?

Answer: problem dependent codification (the codeword length depends on the ensemble performance instead of being pre-fixed)

Traditional decoding

Hamming distance $\quad d(x, y^i) = \sum_{j=1}^{n} \mid (x_j - y_j^i) \mid /2$

Euclidean distance $\quad d(x, y^i) = \sqrt{\sum_{j=1}^{n} (x_j - y_j^i)^2}$

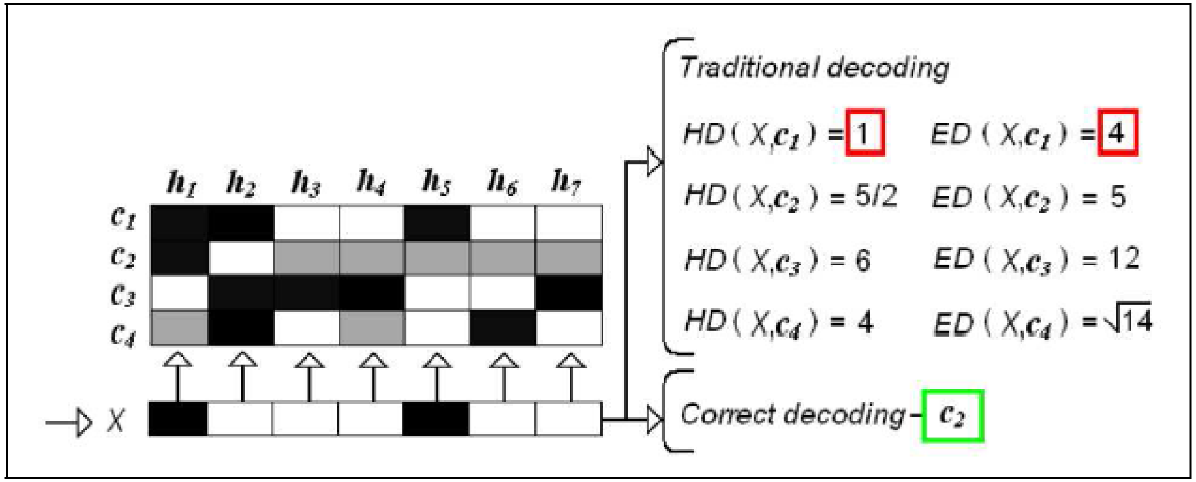


Fig. 1. Example of ternary matrix M for a 4-class problem. A new test codeword is missclassified due to the confussion of using the traditional decoding strategies.

Inverse Hamming distance $\Delta(i,j) = d(y^i, y^j)$

$$Q = [q_1, q_2, ..., q_{N_c}]$$

$$Q = \Delta^{-1} D^T.$$

Attenuated Euclidean distance $d(x, y^i) = \sqrt{\sum_{j=1}^{n} | y_j^i | (x_j - y_j^i)^2}$

Loss-based decoding $d^i(\ell, i) = \sum_{j=1}^{n} L(M(i,j) \cdot f(\ell, j))$

Laplacian strategy We propose a Laplacian decoding strategy to give to each class the distance according to the number of coincidences between the input codeword and the class codeword, normalized by the errors without considering the zero symbol. In this way, the coded positions of the codewords with more zero symbols attain more importance. The distance is estimated by:

$$d(x, y^i) = \frac{C_i + 1}{C_i + E_i + K}$$

where C_i is the number of coincidences from the test codeword and the codeword for class i, E_i is the number of failures from the test codeword and the codeword for class i, and K is an integer value that codifies the number of classes considered by the classifier, in this case 2, due to the binary partitions of the base classiers. The offset $1/K$ is the default value (bias) in case that the coincidences and failures tend to zero. Note that when the number of C and E are sufficiently high, the factor $1/K$ does not contribute:

$$\lim_{C \to 0, E \to 0} d(x, y^i) = \frac{1}{K} \qquad \lim_{C >> K} d(x, y^i) = \frac{C}{C + E}$$

Beta Density Distribution Pessimistic Strategy

$$\psi(z,\alpha,\beta) = \frac{1}{K} z^{\alpha}(1-z)^{\beta}$$

$$a_i \ : \quad \int_{Z_i - a_i}^{Z_i} \psi_i(z) = \frac{1}{3}$$

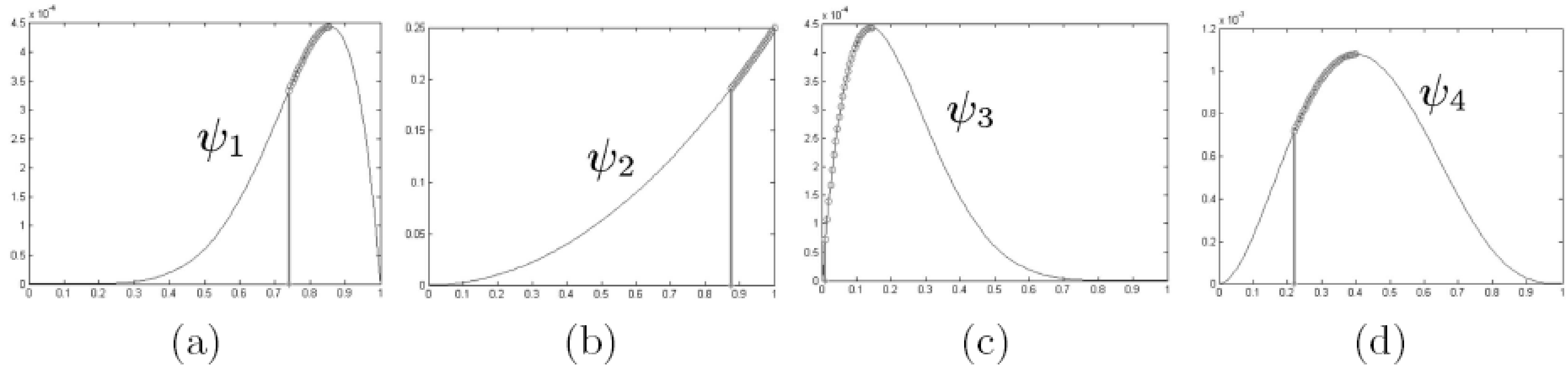


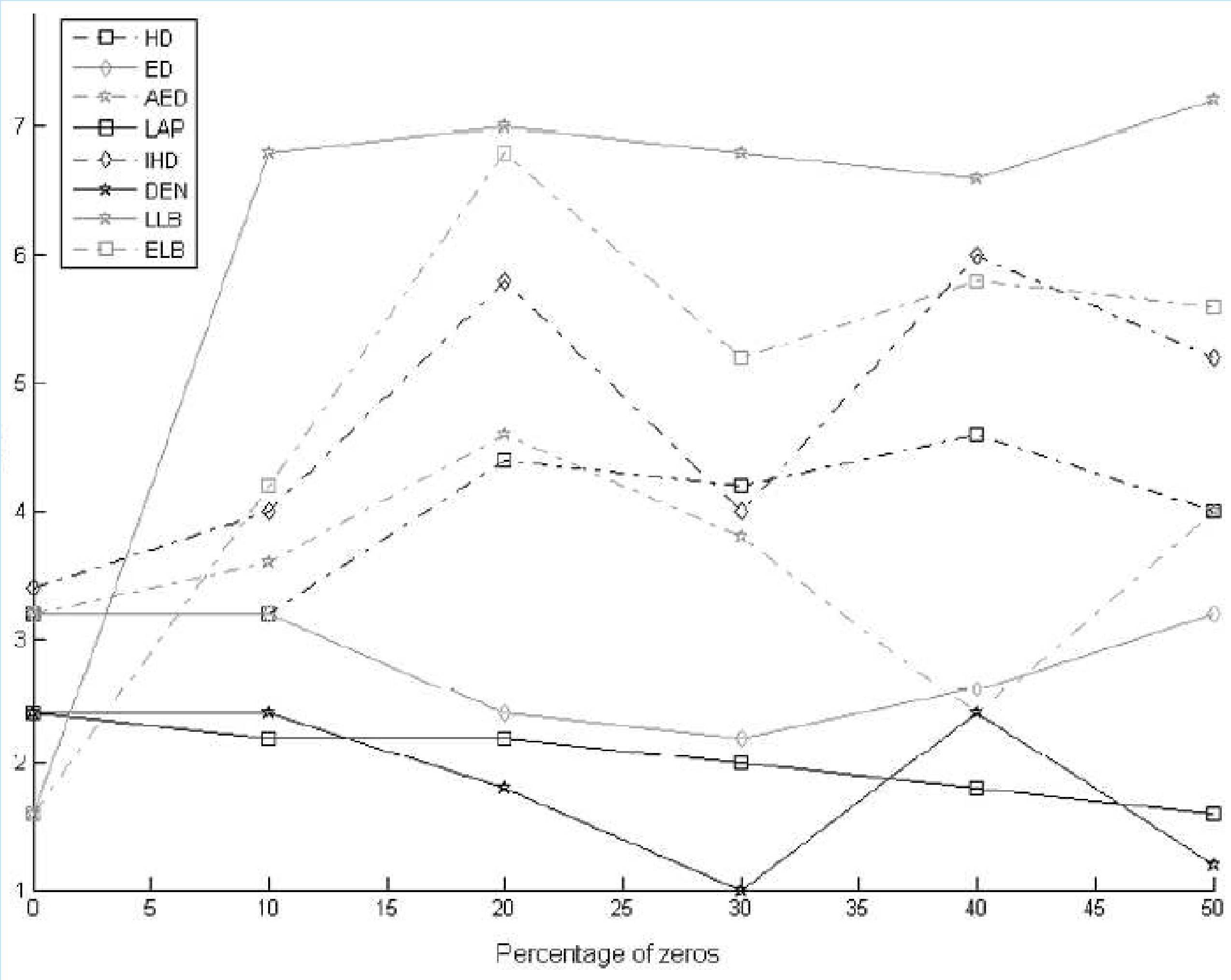
Fig. 2. Pessimistic Density Probability estimations for the test codeword x and the matrix M for the four classes of fig. 1. The probability for the second class allows a successful classification in this case.

Problem	#Train	#Test	#Attributes	#Classes
Dermathology	366	-	34	6
Ecoli	336	-	8	8
Glass	214	-	9	7
Vowel	990	-	10	11
Yeast	1484	-	8	10

Table 1. UCI repository databases characteristics.

Strategy	0% zeros	10% zeros	20% zeros	30% zeros	40% zeros	50% zeros	Global rank
HD	3.2	3.2	4.4	4.2	4.6	4.0	3.9
ED	3.2	3.2	2.4	2.2	2.6	3.2	2.8
AED	3.2	3.6	4.6	3.8	**2.4**	4.0	3.6
IHD	3.4	4.0	5.8	4.0	6.0	5.2	4.7
LLB	**1.6**	6.8	7.0	6.8	6.6	7.2	6.0
ELB	**1.6**	4.2	6.8	5.2	5.8	5.6	4.9
LAP	2.4	**2.2**	2.2	2.0	1.8	1.6	**2.0**
β-DEN	2.4	2.4	**1.8**	**1.0**	**2.4**	**1.2**	**1.9**

Table 2. Mean ranking evolution for the methods on the UCI databases tests when the number of zeros is increased.


HD
ED
AED
LAP
IHD
DEN
LLB
ELB
1
2
3
4
5
6
7
0
5
10
15
20
25
30
35
40
45
50
Percentage of zeros

Conclusions

- Special treatment of the zero symbol

- Laplian and Beta decoding improve state-of-the-art decoding strategies

Thank you!