Efficient Decoding of Ternary Error-Correcting Output Codes for Multiclass Classification

Sang-Hyeun Park and Johannes Fürnkranz

TU Darmstadt, Knowledge Engineering Group

Overview

TECHNISCHE UNIVERSITÄT DARMSTADT

1. Introduction

- Error-Correcting Output Codes are a well-known technique for handling multiclass classification
- transforms a k-class problem into a series of binary problems
- Typically the number of classifiers exceeds the number of classes, which yields a better error-correcting and detection ability
- increase in prediction accuracy comes in general with an computational increase at training and testing

Our Focus: Efficient Testing Phase

Generalization of QWeighted (Park and Fürnkranz, 2007) to arbitrary ternary ECOCs.

Definition, Decomposition

- Error-Correcting Output Codes (Dietterich and Bakiri, 1995) are a general framework for:
  - decomposition of a multiclass problem to a set of binary classification problems
  - decoding individual binary predictors to an unified multi-class prediction
  - all relevant information is summarized in a so-called coding matrix $(m_{i,j}) = M \in \{-1, 1\}^{k \times n}$

	f_1	f_2	f_3	f_4	f_5	f_6
c_1	1	1	1	-1	-1	-1
c_2	1	-1	-1	1	1	-1
c_3	-1	-1	-1	1	-1	1
c_4	-1	-1	1	-1	1	1

  - Allwein et al. (2000) generalized to "ternary" case, i.e. $M \in \{-1, 0, 1\}^{k \times n}$, where 0-symbol means that the corresponding class is ignored during training for the classifier in the column

Decoding

Traditional ECOC Decoding (Hamming Distance)

$$\underset{i \in k}{\text{argmin}}\, d_H(c\vec{w}_i, \vec{p})$$

$$d_H(c\vec{w}_i, \vec{p}) = \sum_{j=1}^{n} \frac{|m_{i,j} - p_j|}{2}$$

$$M = \begin{pmatrix} 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 \\ -1 & -1 & -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & -1 & 1 & 1 \end{pmatrix}$$

$$\vec{p} = (p_1, p_2, p_3, p_4, p_5, p_6)$$

Alternatives

- Euclidian Distance, Attenuated Euclidian & Hamming Distance
- (Allwein et al., 2000) Loss based (with ternary ECOC)
- (Escalera et al., 2006) Laplace Strategy and Beta Density Strategy

Code Types

- two types of exhaustive ternary codes
  1. (k, l)-exhaustive codes: all possible classifiers with l non-zero symbols
  2. a cumulative version, which subsumes in addition (k, j)-exhaustive codes with $j \leq l$
- two types of random ternary codes
  1. the probability of the zero symbol can be specified (sparsity)
  2. random selection from the set of all possible classifiers
- Coding Theory
  - BCH Codes as a representative
    - has been studied in depth
    - minimum Hamming distance can be specified

3. Efficient Decoding of ECOC

TECHNISCHE UNIVERSITÄT DARMSTADT

- Generalization of QWEIGHTED(Park and Fürnkranz, 2007), a simple efficient voting-based pairwise prediction algorithm, to ternary ECOC decoding
  - guaranteed to produce the same result as with original decoding
- QWEIGHTED
  - basic idea: not all pairwise comparisons are needed for determining only the winner class
    - imagine the voting process as a tournament of two-player games
    - consider dominant player who can not be caught up
  - algorithm in one sentence: *Evaluate always the classifier of the two best classes (w.r.t voting) until the best class has no evaluations left.*
    - quickly emerge current favorite as winner (and avoid unnecessary evaluations)
    - or quickly replace favorite - reasonable that the true winner is among the stronger competitors

Above points hold also for ECOC Decoding

QuickECOC algorithm in Steps

The QUICKECOC algorithm consists of the following steps:

1. Selection of the Next Classifier
2. Classifier Evaluation and Update of rel. statistics
3. Test Stopping Criterion
4. if Test fails, repeat with 1.

Selection of the next Classifier

QWEIGHTED (pairwise classification)

- Selection: favorite class c^* vs. strongest competitor class
- above selection is deterministic

QUICKECOC

- Problem: ECOC classifiers involve in general more than two classes
- Solution: Compute a score for all classifiers f_j involving c^*:

$$s(j) = \sum_{i \in K_j} r(i) \qquad \text{and select} \qquad j^* = \text{argmax}_j \, s(j)$$

- K_j is the set of classes which are involved in f_j and are discriminated against in it against c^*
- "select the classifier which discriminates the favorite class against the greatest number of strong competitors"

3. Efficient Decoding of ECOC

Stopping Criteria

- key idea: stop the evaluation process as soon as it is clear which class will be predicted; determine if current favorite can not be caught up by other classes
- Here: if in the worst case the hamming distance of c^* is smaller than all current hamming distances of all other classes $\rightarrow$ STOP

Worst-Case Scenario

- naive: consider remaining classifiers of c^*, assume worst-case, w.r.t Hamming Distance $\rightarrow$ count remaining classifiers
- slightly better:
  - maintain for each class c_i a separate bound l_i (worst-case increase)
  - for all remaining evaluations f_j involving c^*:
    - if f_j discriminates c^* against c_i, increment l_i with 1 (assume negative result for c^*)
    - else if c_i is not involved in f_j increment l_i with 0.5
  - Stopping Criterion: STOP if for each class c_i : $d_H(c^*) + l_i < d_H(c_i)$

4. Experiments

TECHNISCHE UNIVERSITÄT DARMSTADT

Experimental Setup

- 5 encoding strategies: BCH Codes and two versions each of exhaustive and random codes.
- 7 decoding methods: Hamming, Euclidian, Att. Euclidian, linear loss-based, exponential loss-based, Laplacian Strategy and Beta Density Probabilistic Pessimistic
- 7 multiclass datasets with $k \leq 8$ arbitrarily selected from the UCI Machine Learning Repository (Asuncion and Newman, 2007)

All experiments were performed within WEKA with J48 as base-learner and 10-fold stratified cross-validation.

Reduction in Number of Evaluations

Table: QUICKECOC performance using Hamming decoding and exhaustive ternary codes. (The left value depicts the number of average classifier evaluations by QUICKECOC and the right value in italic shows the ratio of it to the full number of classifiers.)

l	vehicle	derm.	auto	glass	zoo	ecoli	machine
2	3.82 *0.637*	7.12 *0.475*	7.95 *0.379*	9.99 *0.476*	9.48 *0.451*	11.75 *0.420*	11.60 *0.414*
3	7.91 *0.659*	26.05 *0.434*	42.86 *0.408*	43.47 *0.414*	41.64 *0.397*	58.85 *0.350*	57.90 *0.345*
4	5.65 *0.808*	46.30 *0.441*	115.22 *0.470*	116.45 *0.475*	107.03 *0.437*	199.31 *0.407*	194.81 *0.398*
5		43.11 *0.479*	163.67 *0.520*	163.98 *0.521*	148.50 *0.471*	369.06 *0.439*	355.23 *0.423*
6		16.54 *0.534*	114.87 *0.529*	116.77 *0.538*	102.41 *0.472*	394.25 *0.454*	369.19 *0.425*
7			34.24 *0.543*	37.84 *0.601*	31.52 *0.500*	234.80 *0.466*	218.09 *0.433*
8						62.17 *0.490*	57.27 *0.451*

- in all cases a reduction was possible
- QUICKECOC performance seems to benefit of increased class-count resp. number of classifiers

Reduction in Number of Evaluations

TECHNISCHE UNIVERSITÄT DARMSTADT

Table: QuickECOC performance on BCH codes

	vehicle	derm.	auto	glass	zoo	ecoli	machine
7	0.764	0.774	0.851	0.880	0.834	-	-
15	0.646	0.656	0.699	0.717	0.659	0.670	0.648
31	0.571	0.564	0.607	0.662	0.581	0.602	0.558
63	0.519	0.506	0.567	0.616	0.517	0.540	0.509
127	0.489	0.447	0.522	0.565	0.477	0.493	0.459
255	0.410	0.380	0.450	0.467	0.397	0.417	0.388

Sparsity of Coding Matrices

TECHNISCHE UNIVERSITÄT DARMSTADT

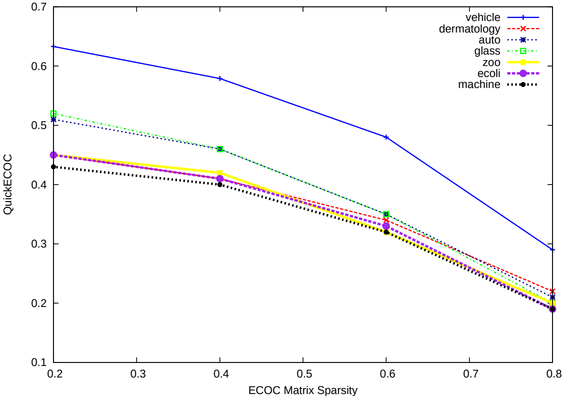

Figure: QUICKECOC performance of random codes in dependence of sparsity

Various Decoding Methods

Table: QUICKECOC performance on *ecoli* with all decoding methods and cumulative exhaustive ternary codes

	Hamming	Euclidian	A. Euclidian	LBL	LBE	Laplace	BDDP	$\|N\|$
$l = 2$	*0.420*	*0.420*	*0.420*	*0.399*	*0.398*	*0.406*	*0.426*	28
$l = 3$	*0.331*	*0.331*	*0.331*	*0.335*	*0.350*	*0.332*	*0.333*	196
$l = 4$	*0.377*	*0.377*	*0.377*	*0.383*	*0.402*	*0.374*	*0.375*	686
$l = 5$	*0.400*	*0.400*	*0.400*	*0.414*	*0.439*	*0.399*	*0.401*	1526
$l = 6$	*0.421*	*0.421*	*0.421*	*0.437*	*0.466*	*0.419*	*0.418*	2394
$l = 7$	*0.427*	*0.427*	*0.427*	*0.444*	*0.475*	*0.426*	*0.425*	2898
$l = 8$	*0.428*	*0.428*	*0.428*	*0.446*	*0.477*	*0.427*	*0.426*	3025

5. Conclusions

TECHNISCHE UNIVERSITÄT DARMSTADT

Conclusions

- presented a general algorithm for reducing the number of classifier evaluations for arbitrary ternary ECOC matrices.
- applicable to a broad spectrum of code types and decoding methods
- increasing sparsity and dimension of ECOC matrix favors reduction performance
- resulting predictions are guaranteed to be equivalent with the original decoding strategy

Future Work

- Improvement potential for next classifier selection, e.g. active learning ideas
- in-depth analysis of fast decoding methods in Coding Theory because they seem to share some similarities

References

Erin L. Allwein, Robert E. Schapire, and Yoram Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1:113–141, 2000.

A. Asuncion and D.J. Newman. UCI machine learning repository, 2007. URL `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

Thomas G. Dietterich and Ghulum Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.

Sergio Escalera, Oriol Pujol, and Petia Radeva. Decoding of ternary error correcting output codes. In José Francisco Martínez Trinidad, Jesús Ariel Carrasco-Ochoa, and Josef Kittler, editors, *CIARP*, volume 4225 of *Lecture Notes in Computer Science*, pages 753–763. Springer, 2006.

Sang-Hyeun Park and Johannes Fürnkranz. Efficient pairwise classification. In Joost N. Kok, Jacek Koronacki, Ramon López de Mántaras, Stan Matwin, Dunja Mladenic, and Andrzej Skowron, editors, *ECML*, volume 4701 of *Lecture Notes in Computer Science*, pages 658–665. Springer, 2007.

Further Evaluations

TECHNISCHE UNIVERSITÄT DARMSTADT

Table: QuickECOC Performance on datasets with high number of classes.

	k	exh. *l* = 3		exh. *l* = 4		cum. exh. *l* = 4	
yeast	10	105.92	*0.294*	524.13	*0.357*	631.02	*0.337*
vowel	11	139.42	*0.282*	797.32	*0.345*	937.43	*0.328*
soybean	19	443.89	*0.153*	5351.90	*0.197*	5804.73	*0.192*

	k	random1		random2	
yeast	10	844.02	*0.296*	1351.94	*0.474*
vowel	11	880.89	*0.306*	1388.74	*0.482*
soybean	19	8481.74	*0.282*	12964.83	*0.431*

QuickECOC algorithm

TECHNISCHE UNIVERSITÄT DARMSTADT

Require: ECOC Matrix $\vec{M} = (m_{i,j}) \in \{-1, 0, 1\}^{k \times n}$, binary classifiers $f_1, \dots, f_n$, testing instance $\vec{x} \in X$

$\vec{l} \in \mathbb{R}^k \Leftarrow 0$ ▷ Hamming distance vector

$c^* \leftarrow \text{NULL}\ ,\ N \leftarrow \{1, \dots, n\}$

while $c^* = \text{NULL}$ **do**

  $j \leftarrow \text{SELECTNEXTCLASSIFIER}(\vec{M}, \vec{l})$

  $p \leftarrow f_j(\vec{x})$ ▷ Evaluate classifier

  for each $i \in K$ **do**

    $l_i \leftarrow l_i + \frac{|m_{i,j} - p|}{2}$

  $\vec{M} \leftarrow \vec{M} \backslash \vec{M}_j\ ,\ N \leftarrow N \backslash \{j\}$

  $i_0 = \operatorname{argmin}_{i \in K} l_i$

  ▷ First stop Criterion

  abort ← **true**

  for each $i \in K \backslash \{i_0\}$ **do**

    $e_F \leftarrow |\{j \in N | m_{i,j} \times m_{i_0,j} = -1\}|$

    $e_H \leftarrow |\{j \in N | m_{i,j} \neq 0$ **and** $m_{i_0,j} = 0\}|$

    if $l_{i_0} + e_F + \frac{1}{2} e_H > l_i$ **then**

      abort ← **false**

  ▷ Second stop Criterion

  if abort **or** $\forall j \in N . m_{i_0,j} = 0$ **then**

    $c^* \leftarrow c_{i_0}$

return c^*