

# Exploiting System Knowledge to Improve ECOC Reject Rules

Paolo Simeone, Claudio Marrocco, Francesco Tortorella

DAEIMI

Università degli Studi di Cassino

Cassino (FR), Italy

{paolo.simeone, c.marrocco, tortorella}@unicas.it

**Abstract**—Error Correcting Output Coding is a common technique for multiple class classification tasks which decomposes the original problem in several two-class problems solved through dichotomizers. Such classification system can be improved with a reject option which can be defined according to the level of information available from the dichotomizers. This paper analyzes how this knowledge is useful when applying such reject rules. The nature of the outputs, the kind of the employed classifiers and the knowledge of their loss function are influential details for the improvement of the general performance of the system. Experimental results on popular benchmark data sets are reported to show the behavior of the different schemes.

**Keywords**-Error Correcting Output Coding; Reject option

## I. INTRODUCTION

Error Correcting Output Coding (ECOC) has emerged in the field of Pattern Recognition as an effective technique to solve multi class classification problems. The original idea [1] was to break the original  $M$  classes problem into  $L$  different binary problems. A  $M \times L$  coding matrix  $\mathbf{C}$  is used to create such sub-problems. For each unknown samples the resulting outputs are combined into an output word which has to be decoded. The decoding is not only a matching process between output and original words, but can be done through a wide variety of decoding techniques which make more robust the decision.

Even though the original motivation for this method founded on the error correcting capabilities of the codes used to group classes, it has also been proved that ECOC provides a reliable probability estimation and a concurrent reduction of both bias and variance [2] which motivate its good generalization capabilities. For such reasons it has been successfully applied to a wide range of real applications such as text and digit classification [3], face recognition and verification [4] or fault detection [5].

A common aim in the realization of these classification systems is the reduction of errors, because a wrong prediction can produce serious consequences in some critical applications, e.g. automated disease diagnosis, currency recognition and biometrics. Since the error cost associated to a misclassification can be extremely high, it is convenient to reject a sample if the decision is not enough reliable.

Many methods have been proposed to enhance classification systems with a reject option [6], [7]; this is also possible

Table I  
EXAMPLE OF A CODING MATRIX FOR A 5 CLASSES PROBLEM.

classes	codewords														
$\omega_1$	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1
$\omega_2$	-1	-1	-1	-1	-1	-1	-1	-1	+1	+1	+1	+1	+1	+1	+1
$\omega_3$	-1	-1	-1	-1	+1	+1	+1	+1	-1	-1	-1	-1	+1	+1	+1
$\omega_4$	-1	-1	+1	+1	-1	-1	+1	+1	-1	-1	+1	+1	-1	-1	+1
$\omega_5$	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1

for ECOC, provided that the decoding stage is modified in order to introduce a reject threshold on the estimated reliability of the system output. To this aim, one has to take into account that the decoding stage can be accomplished in several ways, depending on the level of information available from the dichotomizers. In particular, the dichotomizers can provide their outcomes with different levels of details (crisp labels or soft values); moreover, soft outputs can be processed by the decoding stage in a suitable way if proper knowledge is available about the learning method used by the dichotomizer [8]. Accordingly, the designed reject rule has to be coherent with the output of the dichotomizers and the way they are treated.

In this paper we analyze the relation between the different reject options we can apply and the knowledge we have of the system. The aim is to prove that the more detailed the level of information accessible about the structure of the dichotomizers, the more effective the corresponding reject option. Experiments on real benchmark data sets have been performed to compare the different schemes.

## II. ERROR CORRECTING OUTPUT CODING

ECOC technique is applied by associating an  $L$ -length binary string (*codeword*) to each of the  $M$  classes of a classification problem. These codewords are arranged into an  $M \times L$  coding matrix  $\mathbf{C} = \{c_{ij}\}_{i=1,\dots,M;j=1,\dots,L}$  (e.g. see table I). During the operating phase every dichotomizer outputs a value for each unknown sample, which can be collected into an output vector  $\mathbf{o}$ .

Depending on the dichotomizer outputs and our knowledge of the system different decoding strategies may be adopted. In particular, it is possible to distinct three different cases:

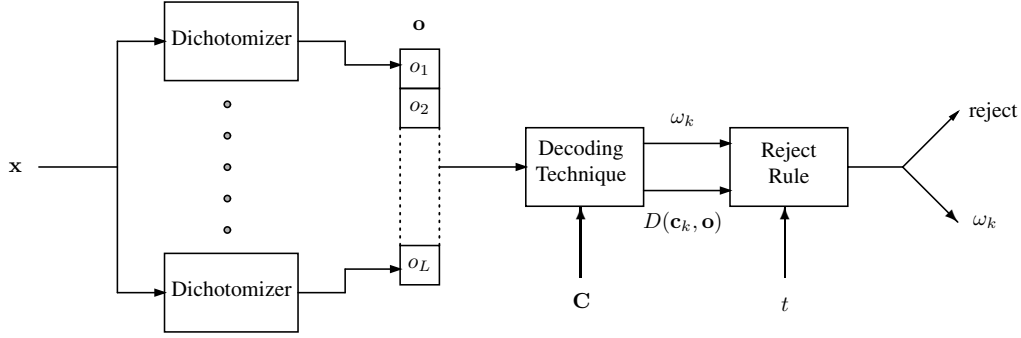


Figure 1. The block diagram for the reject rules over an ECOC system

- **Hard Decoding** - the dichotomizer output is the class label and there is no other exploitable knowledge about base classifiers.
- **Soft Decoding** - the dichotomizer output is a confidence degree about the class membership of the sample and there is no other exploitable knowledge about base classifiers.
- **Loss Decoding** - the dichotomizer output is a confidence degree about the class membership of the sample and we have exploitable knowledge about base classifiers.

#### A. Hard Decoding

In this case dichotomizers only output the class label (+1 or -1). Hard decoding is meant to correct binary errors directly produced by the crisp dichotomizers. A common criterion to classify a sample  $\mathbf{x}$  is to choose the class with the “closest” codeword to the output vector. Typically, the Hamming distance  $D_H$  between two words can be evaluated. Such distance is given by the number of position where the bit patterns of the two words differ, i.e.,:

$$D_H(\mathbf{c}_i, \mathbf{c}_j) = \sum_{h=1}^L \frac{|c_{ih} - c_{jh}|}{2} = \text{card}(\{h : c_{ih} \neq c_{jh}\}). \quad (1)$$

The minimum Hamming distance (MHD)  $d = \min_{i,j} D_H(\mathbf{c}_i, \mathbf{c}_j)$  between any pair of codewords of  $\mathbf{C}$  is a measure of the quality of the code. In particular it is possible to correctly decode any word which contains no more than  $\lfloor (d-1)/2 \rfloor$  single bit errors. A sample  $\mathbf{x}$  is assigned to the class  $\omega_k$  corresponding to the codeword with the minimum Hamming distance from the output vector:

$$\omega_k = \arg \min_h D_H(\mathbf{c}_h, \mathbf{o}). \quad (2)$$

#### B. Soft Decoding

In the case of dichotomizers providing a confidence degree, a common strategy is to consider the real-valued output  $o_h(\mathbf{x})$  normalized in the interval  $[-1, +1]$ , and to collect the

results into a real-valued output vector  $\mathbf{o}$ . As a consequence, the Hamming distance is no longer adequate and a common solution is to replace it with  $L_1$  norm distances [2], [9]:

$$D_1(\mathbf{c}_i, \mathbf{o}) = \sum_{h=1}^L \frac{|c_{ih} - o_h(\mathbf{x})|}{2}. \quad (3)$$

A decision for the  $k$ -th class could be taken according to:

$$\omega_k = \arg \min_i D_1(\mathbf{c}_i, \mathbf{o}). \quad (4)$$

#### C. Loss Decoding

The knowledge of the learning method used by the dichotomizers allows the adoption of another possible decoding rule: the *loss-based decoding* [8]. To introduce such decoding technique, we have to refer to the concept of *margin* [8]. If we assume that  $y$  is the label of a sample  $\mathbf{x}$ , the margin associated to the prediction of a dichotomizer  $o$  on the sample  $\mathbf{x}$  is given by  $yo(\mathbf{x})$ . In this way, if  $o$  provides a wrong prediction for  $\mathbf{x}$ , the sample margin is negative. More precisely, while the sign of the margin indicates the correctness of the classifier, the magnitude estimates the confidence of the classifier in making its prediction on the sample  $\mathbf{x}$ . For this reason the learning algorithms of many dichotomizers aim at maximizing the margins on the samples of a training set. However, since margin maximization in its general form can be intractable, learning algorithms typically try to minimize a *loss function* of the margin  $L(yo(\mathbf{x}))$ , where the expression of  $L(\cdot)$  depends on the dichotomizer architecture.

If the dichotomizers used in the ECOC system are of this type and employ a loss function  $L(\cdot)$  in their learning algorithm, it is possible to consider a *loss-based distance* between the output vector  $\mathbf{o}$  and a codeword  $\mathbf{c}_i$  defined as:

$$D_L(\mathbf{c}_i, \mathbf{o}) = \sum_{h=1}^L L(c_{ih}o_h(\mathbf{x})) \quad (5)$$

In other words, the distance is computed as the total loss produced by all the dichotomizers taking into account, for each prediction  $o_h(\mathbf{x})$ , the corresponding label  $c_{ih}$  coming

from the codeword  $\mathbf{c}_i$ . The underpinning idea for such approach is that the loss  $L(c_{ih}o_h(\mathbf{x}))$  produced by a particular prediction  $o_h(\mathbf{x})$  is minimum for those  $c_{ih}$  with  $i = 1, \dots, n$  whose value equals the true label of the sample  $\mathbf{x}$  in the  $h$ -th dichotomy. The corresponding decoding rule chooses the class  $\omega_k$  corresponding to the minimum loss-based distance, that is to choose the codeword made by the labels most consistent with the predictions of the dichotomizers:

$$\omega_k = \arg \min_i D_L(\mathbf{c}_i, \mathbf{o}). \quad (6)$$

### III. DEFINITION OF REJECT RULES

A common approach to decrease the costs of a classification system consists in turning as many errors as possible into rejects. In a real application the cost of an error is typically higher than a reject cost, thus an effective reject option is a general benefit for the original multi class classification problem[10]. Typically a reject option is accomplished by evaluating the reliability of the decision taken by the classifier and rejecting such decision if the reliability is lower than some given threshold.

The reliability of the decision can be estimated by looking at the distance between the output vector and the nearest codeword both for hard decoding and soft decoding. As well the loss distance estimates the total loss function of the ensemble of classifiers which properly evaluate a confidence level for the output. The simplest approach is then to consider the entire ECOC system as a monolithic classification system and the main idea is to apply a reject option which modifies only the decoding stage, where a threshold can be easily set by fixing any of the distance values.

In our case, it is possible to define a common framework of a reject rule for each of the decoding techniques of the previous section. As illustrated in fig. 1, once evaluated one of the distances in the decoding block, the value is fed to a block which applies the proper threshold. In particular, we can define a generic reject rule as:

$$r(\mathbf{o}, t) = \begin{cases} \omega_k & \text{if } D(\mathbf{c}_k, \mathbf{o}) < t, \\ reject & \text{if } D(\mathbf{c}_k, \mathbf{o}) \geq t. \end{cases} \quad (7)$$

where  $D$  is one of the distances presented in the previous section and  $t$  is chosen according to the employed distance.

It is worth remarking that this scheme does not require any assumption neither on the dichotomizers nor on the coding matrix, thus it can be applied for each decoding technique previously described without modifying the internal organization of the ECOC system. On each method of the external reject option it will be very simple to build the error-reject curve by varying the threshold (whether the distance adopted) and observing the errors and rejects obtained.

Table II  
THE DATA SETS USED.

Data Set	Classes	Features	Length ( $L$ )	Samples
Glass	6	9	31	214
Letter	26	16	63	5003
Pendigits	10	12	31	10992

### IV. EXPERIMENTS

Three publicly available multi class data sets were chosen from the UCI machine learning repository [11] to evaluate the performance of the reject rules. To avoid any bias in the comparison, 12 runs of a multiple hold out procedure have been performed on all the data sets. In each run, the data set has been split into three subsets: a training set (containing the 70% of the samples of each class) to train the base classifiers, a validation set and a test set (each containing the 15% of the samples of each class) used, respectively, to normalize the outputs into the range  $[-1, 1]$  and to evaluate the performance of the classification system. A short description of the data sets is given in table II together with the number of columns of the coding matrix chosen for each data set according to [1].

Modest Adaboost (MA)[12] has been used as base classifier. It has been built using a decision tree with maximum depth equal to 10 as weak learner and 50 boosting steps. For this classifier the loss function is  $L(z) = e^{-z}$ .

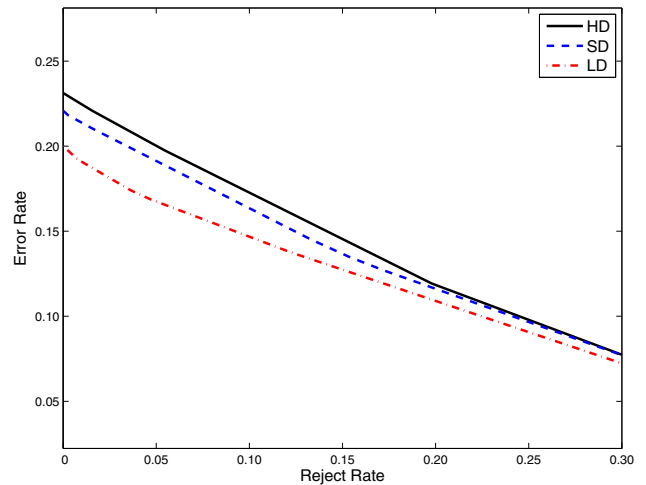


Figure 2. Reject techniques comparison for Glass data set

To have a comparison of the reject rules and to understand how the knowledge of the classification system details is a valuable instrument to improve the performance of the system, we decided to evaluate each data set in terms of error-reject curves obtained by varying the threshold  $t$  in the interval  $[-1, +1]$  with step 0.01. Such results are shown in fig. 2, 3 and 4 in the interval  $[0, 0.30]$  on the x-axis which

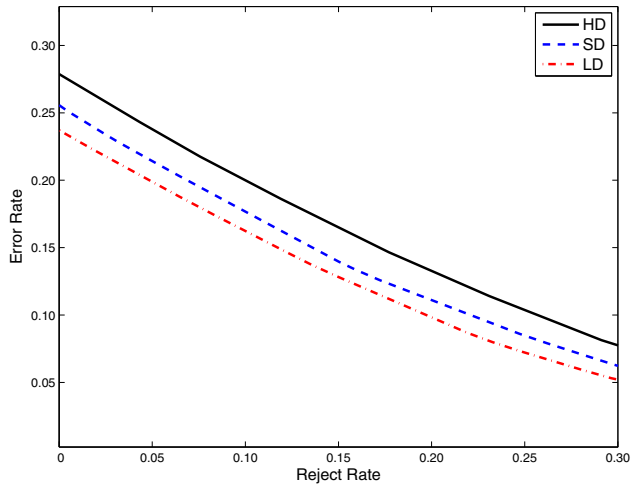


Figure 3. Reject techniques comparison for Letter data set

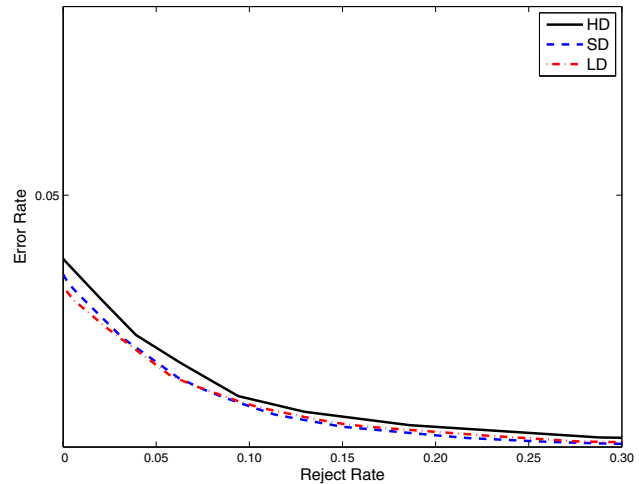


Figure 4. Reject techniques comparison for Pendigits data set

is a significant range in terms of reduction of errors for real applications. Each method has been plotted: Hard Decoding (HD), Soft Decoding (SD) and Loss Decoding (LD).

Figs. 2 and 3 show a constant improvement of the performance as the level of information about the dichotomizers increases. SD gives more reliable results than HD while the best performance are obtained through LD, i.e. if we know the loss function of the base classifier we have the highest classification quality and a faster decreasing of error rate. Fig. 4 confirms this tendency even if in such a case the three methods have really similar performance probably due to the good performance obtained with AdaBoost on this data set (the initial error rate is very low).

## V. CONCLUSIONS AND FUTURE WORKS

In this paper we have studied how the knowledge of base classifiers can influence the performance of a reject rule of an ECOC classification system. Three different rules have been analyzed according to the level of information provided: from the nature of the outputs, as they represents or not a confidence degree, to the model itself. Experiments on benchmark data sets have shown an improvement of the performance in terms of error-reject curve when we have a deeper comprehension of the system details.

Future developments will focus on an extended analysis of the ECOC framework (e.g. the coding matrix and other decoding techniques) and a detailed study of a new reject option to be directly applied to the dichotomizers.

## REFERENCES

- [1] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Artificial Intelligence Research*, vol. 2, pp. 263–286, 1995.
- [2] E. Kong and T. Dietterich, "Error-correcting output coding corrects bias and variance," in *Proceedings of the International Conference on Machine Learning*, 1995.
- [3] J. Zhou and C. Y. Suen, "Unconstrained numeral pair recognition using enhanced error correcting output coding: A holistic approach," in *ICDAR*. IEEE Computer Society, 2005, pp. 484–488.
- [4] J. Kittler, R. Ghaderi, T. Winderatt, and J. Matas, "Face verification via error correcting output codes," *Image Vision Comput.*, vol. 21, no. 13-14, pp. 1163–1169, 2003.
- [5] S. Singh, A. Kodali, K. Choi, K. R. Pattipati, S. M. Namururu, S. C. Sean, D. V. Prokhorov, and L. Qiao, "Dynamic multiple fault diagnosis: Mathematical formulations and solution techniques," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 39, no. 1, pp. 160–176, 2009.
- [6] P. L. Bartlett and M. H. Wegkamp, "Classification with a reject option using a hinge loss," *J. Mach. Learn. Res.*, vol. 9, pp. 1823–1840, 2008.
- [7] G. Fumera and F. Roli, "Analysis of error-reject trade-off in linearly combined multiple classifiers," *Pattern Recognition*, vol. 37, no. 6, pp. 1245–1265, 2004.
- [8] E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," *Journal of Machine Learning Research*, vol. 1, pp. 113–141, 2000.
- [9] F. Masulli and G. Valentini, "An experimental analysis of the dependence among codeword bit errors in ECOC learning machines," *Neurocomputing*, vol. 57, pp. 189–214, 2004.
- [10] C. Chow, "On optimum recognition error and reject tradeoff," *Information Theory, IEEE Transactions on*, vol. 16, no. 1, pp. 41–46, 1970. [Online]. Available: [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=1054406](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1054406)
- [11] A. Asuncion and D. Newman, "UCI machine learning repository," 2007. [Online]. Available: <http://www.ics.uci.edu/~mllearn/MLRepository.html>
- [12] A. Vezhnevets and V. Vezhnevets, "Modest adaboost - teaching adaboost to generalize better," in *Graphicon-2005*, 2005.