FLIP-ECOC: A Greedy Optimization of the ECOC Matrix

Cemre Zor[1 2], Berrin Yanikoglu[1], Terry Windeatt[2], Ethem Alpaydin[3]

[1]Sabanci University, Tuzla, Istanbul, Turkey, 34956
(cemre, berrin)@sabanciuniv.edu
[2]Center for Vision, Speech and Signal Processing, University of Surrey, UK, GU2 7XH
t.windeatt@surrey.ac.uk
[3]Bogazici University, Bebek, Istanbul, Turkey, 34342
alpaydin@boun.edu.tr

Abstract. Error Correcting Output Coding (ECOC) is a multiclass classification technique, in which multiple base classifiers (dichotomizers) are trained using subsets of the training data, determined by a preset code matrix. While it is one of the best solutions to multiclass problems, ECOC is suboptimal, as the code matrix and the base classifiers are not learned simultaneously. In this paper, we show an iterative update algorithm that reduces this decoupling. We compare the algorithm with the standard ECOC approach, using Neural Networks (NNs) as the base classifiers, and show that it improves the accuracy for some well-known data sets under different settings.

1 Introduction

In multiclass classification, ensembles of suboptimal classifiers are preferred over single classifiers due to the advantages they offer in terms of accuracy, complexity and flexibility. The Error Correcting Output Coding (ECOC) is one such technique [3] , where multiple base classifiers are trained according to a preset *code matrix*. Consider an ECOC matrix C, where a particular element $C_{ij} \epsilon (+1, -1)$ indicates the desired label for class i, to be used in training the base classifier j. The base classifiers are the dichotomizers which carry out the two-class classification tasks per each column of the ECOC matrix, according to the input labelling. Each row, called a *codeword*, indicates the desired output for the whole set of base classifiers for the class it is indicating.

During decoding, a given test sample is classified by computing the similarity between the output (hard or soft decisions) of each base classifier and the codeword for each class by using a distance metric, such as the Hamming (L1 Norm) or the Euclidean (L2 norm) distance. The class with the minimum distance is then chosen as the estimated class label. The method can handle incorrect base classification results up to a certain degree. Specifically, if the minimum Hamming distance (HD) between any pair of codewords is d, then up to $\lfloor (d-1)/2 \rfloor$ single bit errors can be corrected. A good practice in code matrix design is to ensure large HD between codewords of different classes in order to have large

error correction capacity and large HD between pairs of *columns,* in order to end up with uncorrelated outputs of deterministic classifiers [3].

Although the tasks of the base classifiers are significantly simplified compared to the overall classification problem, the sub-problems are still non-trivial generally. While the individual base errors may be corrected by using the ECOC approach, the encoding and the decoding of ECOC matrix are open problems. Our aim in this paper is to optimize the original matrix so as to better match the trained base classifiers. This is done by considering the performances of base classifiers over the individual classes, and changing the ECOC matrix whenever it is deemed beneficial, while taking the HD information into account.

2 Previous Work

ECOC is a powerful ensemble method for multiclass classification. For encoding the ECOC matrix, there are some commonly used data-independent techniques such as one-versus-all, one-versus-one, dense random and sparse random [4] in addition to the computationally expensive exhaustive codes, which do not guarantee the best performance. By using data dependent ECOC designs to create subproblems which can better fit the decision boundaries of the main problem, the aim is to increase the overall accuracy and overcome expensive parameter optimizations [2]. Although problem dependent coding approaches are successful, it has been theoretically and experimentally proven that the randomly generated long or deterministic equidistant code matrices are also close to optimum performance when used with strong base classifiers [9,10].

As for the decoding of the ECOC matrix, apart from the usual $L1$ decoding with the HD, weighted decoding approaches, "Centroid of Classes", "Least Squares" and "Inverse Hamming Distance" methods[11] can be used. Many static and dynamic pruning methods are also applied to the ECOC so as to increase the efficiency and accuracy.

Other than the research on encoding, there has been little work to update the ECOC matrix, or to analyze the performance of the base classifiers. In [8] Alpaydin et al train a multilayer perceptron to learn the new ECOC code matrix, allowing small modifications from the original. In [7], the update of the one-versus-one coding matrix has been carried out in a problem-dependent way and the generalization capability of the system is shown to increase.

Our approach is applicable to any ECOC matrix design. The experiments are carried out on random ECOC matrices of varying column sizes for systems having NNs as base classifiers. When the number of nodes and epochs used in NNs is small, increases up to 16% in the overall classification accuracy are obtained through 10-fold cross validation (CV). Since long random matrices used with strong base classifiers are proven to perform close to ideal, there is no remarkable change under this setting.

3 Proposed Method

Consider the ECOC matrix C, C_{ij} as the entry of the matrix on row i and column j, and A_{ij} as the accuracy of the base classifier j, with regard to class i. A_{ij}, measured on a validation set, is the proportion of the samples in class c_i that are correctly classified by j according to the target value specified by C_{ij}.

We propose to flip C_{ij} entries that have corresponding A_{ij} values lower than 0.5 so as to better match what is learned by the base classifiers; i.e. if the decision of the base classifier does not match the target, we consider changing the target. However, to keep the decisions of the individual base classifiers as uncorrelated from each other as possible and avoid deterioration of the row-wise and column-wise HDs, we have a certain criterion on the flipping process. Without any stopping criterion, flipping can yield a decrease in the HD between classes; which can adversely affect the small accuracy gain obtained on a single class by flipping the decision of a single base classifier.

In our method, we first list the C_{ij} entries in ascending order according to their corresponding A_{ij} values until 0.5. By using a hill climbing method, which results in a suboptimal solution due to the greedy decisions it takes in each iteration, the C_{ij} entries are sequentially proposed for flipping. In each iteration, a flip and therefore an ECOC update is accepted if the validation set accuracy does not decrease when the updated ECOC matrix is used in the decoding process instead of the current one. By considering the validation accuracy in this stage, we expect the method to take care of the row and column-wise HD information together with the error correction capacity, and therefore carry out updates without causing any degradation. In Algorithm 1, pseudo-code for the method can be found.

Algorithm 1 FLIP-ECOC

```
calculate A and C matrices
list C_ij s.t A_ij < 0.5 is in ascending order
noElements←number of elements in the list
current state←original ECOC matrix, C
for t = 1 : noElements do                                   ▷ start hill climbing
    nextState←flip t^th element of C
    Δgain ←valAccuracy[nextState]−valAccuracy[currentState]
    if Δgain ≥ 0 then
        currentState←nextState                  ▷ currentECOC←updatedECOC
    end if
end for
```

We have also studied a ternary ECOC [4] extension of the proposed method, in which the elements C_{ij} are flipped if their corresponding A_{ij} values are below a threshold (e.g. 0.4) as in the Flip-ECOC method, whereas the elements between that lower and a proposed upper threshold (e.g. 0.6) are set to zero. The use of a third label, namely zero, allows us to handle the cases where the classifier

Table 1. Summary of the 5 UCI MLR datasets

	# Training Samples	# Test Samples	# Attributes	# Classes
Glass Identification	214	-	10	6
Dermatology	358	-	33	6
Segmentation	4435	2000	36	6
Landsat Satellite Image	210	2100	19	7
Yeast	1484	-	8	10

decisions are not strong enough to justify a labelling to either class. However, our results with the extension have not shown remarkable improvement over Flip-ECOC and we only present Flip-ECOC outcomes in the experiments session. We believe that this is due to the problematic decoding of ternary ECOC matrices [6] and aim to address this problem as a future work. Finally, we also applied simulated annealing as a greedy search technique. As the results are not significantly different than those of the hill climbing, hill-climbing has been selected as the search procedure for the sake of decreased test complexity.

4 Experimental Results

Experiments have been carried out on 5 UCI MLR [5] datasets. NNs (using the Levenberg-Marquart algorithm) are used as the base classifiers, random coding as the coding strategy and the HD as the metric in the decoding stage (i.e. the standard approach). In the experiments, the number of columns of the ECOC matrix varies between 10 and 150 (namely 10,15,25,75 and 150), that of NN nodes between 2 and 16, and the level of training between 2 and 15 epochs.

Table 2 shows the summary of the 5 datasets. For the datasets having separate test sets, the input training samples have been randomly split into a training and a validation set. The average results are recorded for 10 independent runs. For the rest, 10-fold CV has been applied together with a random split of the training samples into two as above. The size of the validation set has been selected to be equal to that of the training, as it plays an important role both as a flipping and a stopping criterion in the Flip-ECOC algorithm.

In Figure 1, the relative accuracy gain of Flip-ECOC against the standard approach is presented. The trend in the graphs show that the power of the method increases when simpler ensembles with fewer number of nodes and/or epochs and/or columns are used. Figure 2 presents the actual and the updated accuracies for some datasets. When the ECOC setup is close to optimum (i.e. when large number of columns are used with strong base classifiers under random coding scheme), the method starts to lose its capacity to increase the overall accuracy as expected; however there is no significant decrease either.

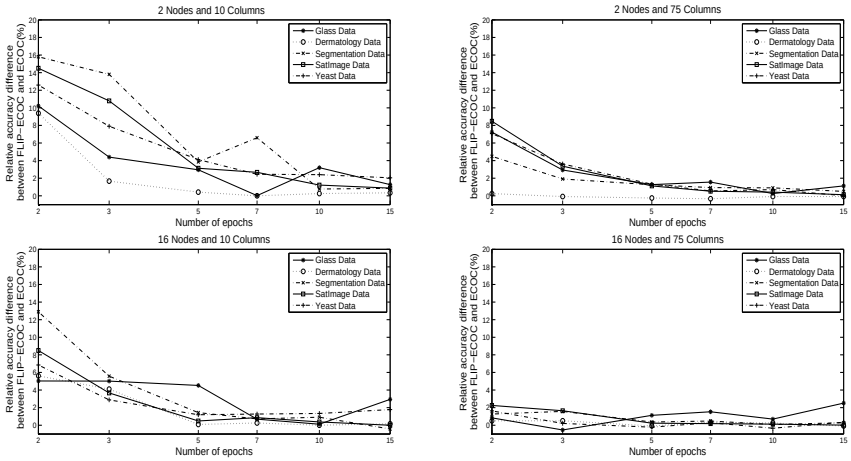

Fig. 1. Relative accucary difference between the Flip-ECOC and standard ECOC approaches vs. number of epochs. First Row: for 2 Nodes and 10 (left), 75(right) Columns Second Row: for 16 Nodes and 10 (left), 75 (right) Columns

5 Discussion

The proposed method improves the default ECOC accuracy in almost all problems and settings. The extent of the improvement varies up to 16% in certain cases. Significant improvements are observed when the base classifiers and the corresponding decision boundaries are simpler; either when fewer number of nodes are used, resulting in a less complex classifier, or when the networks are trained using fewer epochs, resulting in a less tuned decision boundary.

The improvements are larger when the number of columns is small (e.g. $<$ 75). When the number of columns is large, more flips are necessary to change the overall accuracy, due to the large HD already helping with the decoding. However, when there are too many flips the HD between certain class pairs may decrease and counter-balance the improvements to be gained from updated individual base classifier accuracies. Therefore, we may end up with smaller accuracy gains compared to the ones obtained by using fewer columns.

Finally, the proposed method is less applicable when highly accurate base classifiers and long random ECOC matrices are employed, where it is already proven to yield results close to optimal. While theoretically interesting, the use of ECOC approach with large number of accurate base classifiers is not practical, due to prohibitive training time. Therefore, we believe that the reliable improvements gained with very small effort in simpler ECOC ensembles are significant.

Techniques on updating the matrix can be further examined by concentrating on the settings in which improvements were small. Future work is also aimed at using ECOC matrices other than random ones together with different types of base classifiers and different decoding techniques.

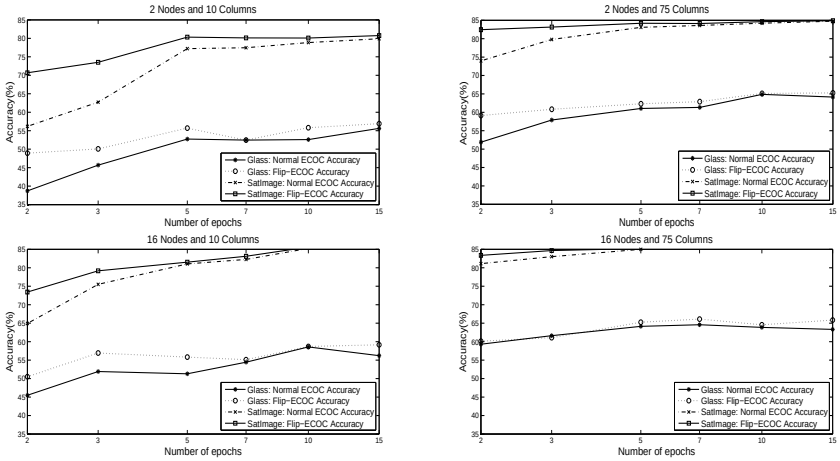

Fig. 2. Accuracies of the Flip-ECOC and standard ECOC approaches vs. number of epochs. First Row: for 2 Nodes and 10 (left), 75(right) Columns Second Row: for 16 Nodes and 10 (left), 75(right) Columns

References

1. Tumer K., Ghosh, J.: Error Correlation and Error Reduction in Ensemble Classifiers. Connection Science, Special Issue on Combining Artificial Neural Networks: Ensemble Approaches, 8(3) 385–404 (1996)
2. Escalera, S., Tax, D. M. J., Pujol, O., Radeva, P., Duin, R. P. W.: Subclass Problem-Dependent Design for Error-Correcting Output Codes. In: IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 30, no. 6, pp. 1041-1054 (2008)
3. Dietterich, T.G., Bakiri, G.: Solving Multi-class Learning Problems via Error-Correcting Output Codes. J. Artificial Intelligence Research 2. 263–286 (1995)
4. Allwein, E., Schapire, R., Singer, Y.: Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers. JMLR 1. 113–141 (2002)
5. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository, http://www.ics.uci.edu/~mlearn/MLRepository.html. School of Information and Computer Science, University of California, Irvine, CA (2007)
6. Escalera, S., Pujol, O., Radeva, P.: On the Decoding Process in Ternary Error-Correcting Output Codes. In: CIARP, vol. 4225, pp. 753–763 (2006)
7. Escalera, S., Pujol, O., Radeva, P.: Recoding Error-Correcting Output Codes. Proceedings of the 8th International Workshop on MCS, vol. 5519, pp. 11–21 (2009)
8. Alpaydin, E., Mayoraz, E.: Learning error-correcting output codes from data. In: Proc. Int. Conf. Neural Networks (ICANN) (1999)
9. James, G. M.: Majority Vote Classifiers: Theory and Applications. PhD Thesis, Department of Statistics, University of Standford (1998)
10. James, G. M., Hastie, T.: The Error Coding Method and PICT's, Computational and Graphical Statistics, vol. 7, no. 3, pp. 377-387 (1998)
11. Windeatt, T., Ghaderi R.: Coding and Decoding Strategies for Multi-class Learning Problems. Information Fusion, 4(1), pp. 11-21 (2003)