International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 10, ISSN No. 2455-2143, Pages 124-128
Published Online February 2023 in IJEAST (http://www.ijeast.com)

A STUDY PAPER ON ERROR CORRECTING OUTPUT CODE BUILD ON MULTICLASS CLASSIFICATION

Rinkal K. Patel, Irfan Poladi
Department of Computer Engineering
Gokul Global University, Sidhpur, Gujarat, India

Abstract— **Error-correcting output codes is a technique for using binary classification models on multi-class classification prediction tasks. Error-Correcting Output Codes (ECOC) represents a successful framework to deal with these kinds of problems. Recent works in the ECOC framework appear notable performance improvements. The ECOC framework is a high-level tool to deal with multi-class categorization problems. As the error correcting output codes have error correcting ability and improve the generalization ability to base classification. This library contains both modern coding (one-versus-one, one-versus- all, dense-random, sparse-random, DECOC, forest- ECOC, and ECOC-ONE) and decoding designs (hamming, Euclidean, inverse hamming, laplacian, β-density, density, attenuated, loss-based, probabilistic kernel-based, and loss weighted) with the framework defined by the authors, as well as the option to include your own coding, decoding, and base classifier.**

Keywords— **error-correcting output codes, coding, decoding, multi-class classification, open source, matlab, and octave.**

I. INTRODUCTION

The task of supervised machine learning can be seen as the problem of finding an unknown function C(x) given the training set of example pairs <xi,C(x)>. C(x) is usually a set of discrete tag. For example, in face detection, C(x) is a binary function C(x) ϵ {face, nonface}, in optical digit recognition C(x) ϵ {0 . . . 9}.

In order to address the binary classification task many techniques and algorithms have been proposed: neural networks, decision trees, large margin classification techniques, etc. Some of those technique can be easily extended to multiclass problems. However, some other powerful and popular classifiers, such as AdaBoost [1] and Support Vector machines [18], do not extend to multiclass easily. In those conditions, the usual way to proceed is to reduce the complexity of the multiclass problem into multiple simpler binary classification problems.

There are many different Techniques for reducing multiclass to binary classification problems. The most easy approach considers the comparison between each class against all the others. This produces Nc binary problems, where Nc is the number of classes. Other researchers suggested the comparison of all possible pairs of classes [3], resulting in an set of binary problems. In the literature, one can find several important binary classifiers. However, when one needs to deal with multiclass classification problems, many learning techniques fail to manage this information. Instead, it is build to construct the classifiers to distinguish between just two classes and to combine them in some way. In this sense, ECOCs were born as a general framework to combine binary problems to address the multiclass problem. The strategy was introduced by Dietterich and Bakiri in 1995. Based on the error correcting ideas and because of its ability to correct the bias and variance errors of the base classifiers, ECOC has been successfully applied to a wide range of applications such as face recognition, face verification, text recognition, and manuscript digit classification.

It was when All wein etal. [4] introduced a third symbol (the zero symbol) in the coding process when the coding step received special attention. This symbol increases the number of partitions of classes to be considered in a ternary Error- correcting output (ECOC) framework by allowing some classes to be ignored. Then, the ternary coding matrix becomes $M \varepsilon \{-1,+1,0\}^{N \times n}$. In this situation, the symbol zero means that a particular class is not considered by a certain binary classifier. Dietterich and Bakiri[7] presented a general framework in which the classification is performed according to a set of binary error correcting output codes(ECOC).

In this approach, the problem is transformed inn binary classification sub problems, where n is the error correcting output code length $n \varepsilon \{Nc, ... ,\infty\}$. Then, the output of all classifiers should be combined—traditionally using Hamming distance. The approach of Dietterich and Bakiri was improved by All weinetal. [6] by introducing an uncertainty value in the ECOC design and exploring alternatives for mixing the resulting outputs of the classifiers. In particular, they introduced loss-based decoding as a way of joining the classifiers. Recently, Passeriniet al. [2] proposed a new decoding function that combines the margins through an estimate of the class conditional probabilities. error correcting output codes strategies have been proven to be quite competitive with/better than other multiclass extensions of SVM and Adaboost [8], [9].

II. ERROR CORRECTING OUTPUT CODES

A. Steps for ECOC are as follows –

1) Given a set of Nc classes, the basis of the error correctingoutput codes framework consists of designing a codeword for each of the classes.
2) These code words encode the membership information ofeach class for a given binary Problem.
3) Arranging the code words as rows of a matrix, we obtain―a coding matrix Mc‖ ,where Mc {-1, 0, +1}Nc×n, being n length of the code words codifying each Classes.
4) From the point of view of learning, Mc is constructed by considering in binary problems each one corresponding to acolumn of the matrix Mc.
5) Each of these binary Problems splits the set of classes in two partitions (codedby+1or-1 in Mc according to their class set belonging or 0 if the class is not considered by the current binary problem.
6) Then at the decoding step, we applying the n trained binary classifiers, a code is obtained for each data point in the test set.
7) This code is compared to the base code words of each class defined in the matrix Mc, and the data point is assigned to the class with the closest codeword.

Below figure show error correcting output codes coding design for a 4 class problem. White, black and grey positions corresponds to the symbols +1, -1 and 0,.Once the four binary problems are learnt, at the decoding step a new test sample X is tested by then classifiers. Then the new code word x={x1,…,xn}is compared with the class code words {C1,…, C4}, classifying the new sample by the class Ci which codeword minimizes the decoding measure.

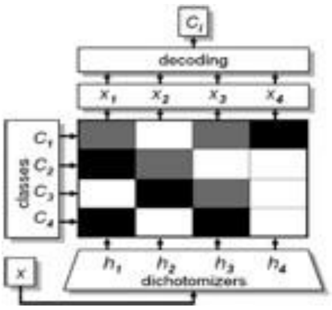

Fig. 1. ECOC Design

B. Coding Design –

Here the ECOC coding design covers the state-of-the art of coding strategies, mainly divided in two main groups: problem independent approaches ,which do not take in to account the distribution of the data to define the coding matrix, and the problem-dependent designs ,where in formation no f the particular domain issue d to guide the coding design.

1) **Problem- Independent ECOC Designs –** The methods of problem-independent are as follows:

- One-versus-all (Rifkin and Klautau, 2004):Nc dichotomizers are learnt for Nc classes ,where each one splits one class from the rest of classes.
- One-versus-one (Nilsson, 1965): n =Nc(Nc −1)/2dichotomizers are learnt for Nc classes, Splitting teach possible pair of classes.
- Dense Random (Allweinetal., 2002): n=10 · logNc dichotomizers are suggested to be learnt for Nc classes ,where P(−1)=1−P(+1), being P(−1) and P(+1) the probability of the symbols -1 and +1 to appear, respectively. Then, from a set of defined random matrices, the one which maximizes a decoding measure among all possible rows of Mcis selected.
- Sparse Random (Escaleraetal.,2009):n=15 · logNc dichotomizers are suggested to be learnt for Nc classes, where P(0)=1−P(−1)−P(+1),defining a set of random matrices Mc and selecting the one which maximizes a decoding measure among all possible rows of Mc.

2) **Problem – Dependent ECOC Designs –**

- DECOC (Pujolet al., 2006) : problem-dependent design that uses n = Nc−1 dichotomizers. The partitions of the problem are learn by means of a binary tree structure using exhaustive search or a SFFS criterion. Finally, each internal node of the tree is embed as a column in Mc.
- Forest- ECOC (Escaleraetal., 2007): problem-dependent design that uses n= (Nc−1) · T dichotomizers, where T stands for the number Of binary tree structures to be embedded. This approach extends the variableness of the classifiersof the DECOC design by including extra dichotomizers.
- ECOC- ONE (Pujolet al., 2008): problem- dependent design that uses n= 2 · Nc suggesteddichotomizers. A validation sub-set issues due to extend any initial matrix Mc and to increase its generalization by including new dichotomizers that focus on difficult to split classes.

C. Decoding Design –

- This notation corresponds to that used in (Escaleraet al., 2008):
- Hamming decoding: HD(x,yi)= ,being x a test codeword and yi a codeword from Mc corresponding to class Ci.
- Inverse Hamming decoding :IHD(x,yi)=max(−1DT

),where (i1,i2)=HD(yi1,yi2), and D is the vector of Hamming decoding values of the test codeword x for each of the base codewords yi.

- Euclidean decoding: ED(x, yi) = n 1 (x j − y j)
- Attenuated Euclidean decoding:
- Probabilistic-based decoding: PD(yi,x) = −log (j∈[1,..,n]:Mc (i,j) ≠0 P(xj =Mc (i,j)|fj)+K), Where K is a constant factor that collects the probability mass dispersed on the invalid codes, and the probability P(xj =Mc(i,j)|fj)is estimated by means of, where vectors and are obtained by solving an optimization problem (Passerinietal.,2004).
- Laplacian decoding: LAP(x, yi) = where αi is the number of matched positions between x and yi, βi is the number of miss-matches without considering the positions coded by 0, and K is An integer value that codes the number of classes considered by the classifier.

III. OUTLINE OF ECOC ALGORITHM

***1)* Training**

1. Load training data and parameters, i.e., the length of code Land training class K.
2. Create a L-bit code for the K classes using a kinds of coding algorithm.
3. For each bit, instruct the base classifier using the binary class (0 and 1) over the total training data.

***2)* Testing**

1. Assign the test example the class with the largest votes.
2. Apply each of the L classifiers to the test example.

A. What makes a good ECOC? –

The key problem for Error correcting output Code (ECOC) approach is how to design the coding matrix M. Many studies [10, 11, 12, 13, and 14] have shown that the final classifier will have good discriminate ability if the coding matrix M has the following characteristics:

1) Characteristic 1: Row separation each codeword (a row in the coding matrix M) should be well separated in Hamming distance from each of the other code words.

2) Characteristic 2: Column separation each column should be uncorrelated with one another. This means that the binary classifiers of different columns have low relationship among them.

3) Characteristic 3: Binary classifiers have low Errors while for recognition of a large number of classes, besides classification accuracy, the efficiency is also quite important. To make a quick decision, it isexpected to evaluate as little binary classifiers as possible. This requires the code word to be efficient (i.e. contains a small number of bits). As explained in [15], for a code to be efficient, different bits should be independent of each other, and each bithasa50% chance of being one or zero. In ECOC design, independent bits can be relaxed as uncorrelated columns (i.e. property 2 mentioned above). And 50% chance offering for each bit requires:

4) Characteristic 4: Balanced column For each column i, the numbers of 1 and−1 are equal, i.e., Finding an ECOC satisfying the above characteristics is a NP- hard problem [16]. So we can say that for efficient and accurate recognition of a large number of classes, a good ECOC is expected to have the following characteristics:

- Efficient -requires a small number of bits.
- Good diversity -the coding matrix has good row and column separation.
- The resulting binary classifiers are accurate.

B. What's so good about ECOC?–

1. Improves classification accuracy.
2. Can be used with many different classifiers.
3. Commonly used in many areas.
4. Not prone to over fitting.
5. Possibly try a variant.

C. Practical Advantages of ECOC–

1. Programs can be written easily, quickly, and with little effort.
2. It is flexible and can be used with any learning algorithm.
3. Able to reduce the bias and variance produced by the learning algorithm. So it widely used to deal with multi-class categorization problems.
4. Low computational cost.
5. Out performs the direct multiclass method.
6. It can be used for textual data, numeric data, discrete data, and many other kinds of information.
7. Generally a running scheme-can be used various learning tasks.
8. Good generalization.

D. Disadvantages –

1. ECOC is not effective if each individual codeword is not separated from the other codewords by a large Hamming distance.
2. The ECOC method is only successful if the errors in the various bit positions are relatively uncorrelated, so that there are few simultaneous errors across various bit positions at the same time. If there are many simultaneous errors, the ECOC will notable to correct them (Peterson & Weldon, 1972).
3. ECOC support vector machines are not always superior to one-versus-all fuzzy support vector machines.
4. Other ECOC schemes can be more efficient than one-versus-all schemes.
5. Sometimes decomposition of multi-class.
6. Problem in to multiple binary problems we are doing in ECOC incurs considerable bias for centroid classifier,

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 10, ISSN No. 2455-2143, Pages 124-128
Published Online February 2023 in IJEAST (http://www.ijeast.com)

which results in noticeable degradation of performance for centroid classifier.

7. Finding the optimal ECOC is NP hard.

E. Comparison of Some ECOC methods –

1) One-Versus-All strategy

The most well-known binary coding strategies are the one-versus-all strategy [17], where each class is discriminated against the rest of classes. In Fig. 1a, the one-versus-all ECOC design for a four-class problem is shown. The white regions of the coding matrix M correspond to the positions coded by 1and the black region sto-1. Thus, the codeword for class C1 is{1,-1,-1,- 1}. Each column I of the coding matrix codes a binary problem learned by its corresponding dichotomizer h1. Asan example, dichotomizer h1 learns C1 based on classes C2, C3, and C4, and dichotomizer h2 evaluates C2 based on classes C1, C3, and C4, etc.

2) The Dense Random Strategy

As a result of the dense random matrix [10], the Hamming distance [7] is maximized between the rows and columns of the resulting matrix in terms of row and column separation. In Fig.1c, we show an example of a dense random matrix for a four-class problem.

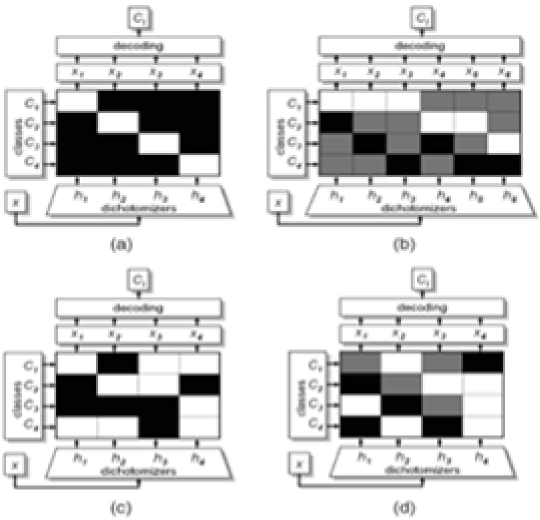

Fig. 2. (a) One-versus-all (b) one-versus-one (c) Dense randomand (d) sparse random ECOC Designs

3) One-Versus-One and Random Sparse Strategy

It was when All wein etal.[10] introduced a third Symbol (the zero symbol) in the coding process when the coding step received special attention. In a ternary ECOC framework, this symbol increases the number of partitionsof classes that can be considered. Then the ternary coding matrix becomes. In this case, the symbol zero means that a particular class is not considered by a certain binary classifier. 1b shows the one-versus-one ECOC configuration for a four-class problem. In this case, the gray positions correspond to the zero symbols. A possible sparse random matrix for a four-class problem is shown inFig. 1d.

IV. CONCLUSION

In this paper the different coding and decoding methods for Error Correcting Output Code have been studied. Advantages and disadvantages of some ECOC coding methods are discussed. As a result of this study on ECOC, we can conclude that Error Correcting Output Code delivers better performance than other methods.

V. REFERENCE

[1]. K.Crammer and Y.Singer. (2002) – On the Learn ability and Design of Output Codes for Multiclass Problems,

[2]. ‖ Machine Learning, vol. 47, no. 2-3, (pp.201-233).

[3]. A.Passerini, M.Pontil, and P.Frasconi. (2004) – New Results on Error Correcting Codes of Kerne Machines,

[4]. ‖ IEEETrans.NeuralNetworks, vol.15, no.1, (pp.45-54).

[5]. V.N.Vapnik. (Springer1995) – The Nature of Statistical Learning Theory.

[6]. Y.Freund and R.E.Shapire. (1997) – A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting,‖ J.Computer and System Sciences, vol.55,no.1, (pp.119-139).

[7]. T.Hastie and R.Tibshirani. (1998) – Classification by Pairwise Coupling-‖ Annals of Statistics, vol.26, no.2, (pp.451-471).

[8]. E.L Allwein, R.E Shapire, and Y. Singer. (2000) – Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers, ‖ J. Machine Learning Research, vol.1,(pp.113-141).

[9]. T.G.Dietterich and G.Bakiri. (1995) – Solving Multiclass Learning Problems via Error-Correcting Output Codes‖ J. Artificial Intelligence Research, vol. 2, (pp.263-286).

[10]. R.E. Schapire. (1997) – Using Output Codes to Boost Multiclass Learning Problems,‖ Machine Learning: Proc. 14th Int'lConf., (pp. 313-321).

[11]. C.Hsu and C.Lin. (Mar.2002) – A Comparison of Methods For Multi-Class Support Vector Machines,

[12]. ‖ IEEE Trans. Neural Networks, vol.13, no.2, (pp.415- 425).

[13]. E.L.Allwein and R.E. Schapire. (2000). Reducing Multiclass to binary: A unifying approach for Margin classifiers .Journal of Machine Learning Research, (1:113–141).

[14]. N. Garc´ı a-Pedrajas and C. Fyfe. (2008). Evolving output codes for multiclass problems. IEEE Trans. Evolutionary Computation, (12(1):93–106).

[15]. R. Ghaderi and T.Windeatt. (2000). Circular ecoc: A The oretical and experimental analysis .In ICPR, (pages 2203–2206).
[16]. O.Pujol and P.Radeva. (2006). Discriminate ecoc: A heuristic method for application dependent design of error correcting output codes. (PAMI,28(6):1007–1012).
[17]. R. Schapir and Y. Singer. (1995). Solving multiclass learning problems via error-correcting output codes. Journal of Artificial Intelligence Research, (2:263–286).
[18]. Y.Weiss ,A.Torralba, and R.Fergus. (2008). Spectral hashing. In NIPS.
[19]. K.Crammer and Y.Singer. (2002). On the learn ability And design of output codes for multiclass problems. Machine Learning, (47(2-3):201–233).
[20]. N.J.Nilsson. (1965). Learning Machines. McGraw-Hill.
[21]. T.Hastie and R.Tibshirani (1998) – Classification by Pair wise Grouping-‖ Proc. Neural Information Processing Systems Conf.,vol.26, (pp. 451-471).