WEIGHT OPTIMIZATION AND LAYERED CLUSTERING-BASED ECOC

Xiao-Lei Zhang and Ji Wu

Multimedia Signal and Intelligent Information Processing Laboratory,
Tsinghua National Laboratory for Information Science and Technology,
Department of Electronic Engineering, Tsinghua University, Beijing, China.
huoshan6@126.com, wuji_ee@tsinghua.edu.cn

ABSTRACT

Error correcting output code (ECOC) is a general framework of solving a multiclass classification problem via a binary-class classifier ensemble. In this paper, we propose a new heuristic coding method, named weight optimization and layered clustering-based ECOC (WOLC-ECOC). It iterates the following two steps until the training risk converges. The first step employs the layered clustering-based approach [1]. The approach can construct multiple different strong binary-class classifiers on a given binary-class problem, so that the heuristic training process will not be blocked by some difficult binary-class problems. The second step is the weight optimization technique [2]. It guarantees the non-increasing of the heuristic training process whenever we add new classifiers to the ECOC ensemble. Experimental results on several benchmark sets demonstrate that WOLC-ECOC is more effective than 15 referenced coding-decoding ECOC pairs.

Index Terms— Classifier ensemble, error correcting output codes, multiple classifier systems, multiclass classification problem.

1. INTRODUCTION

Over the last decades, *classifier ensembles* (i.e. multiple classifier systems), such as *bagging*, *boosting*, and their variations, have been proven to be effective approaches for solving learning problems like classification, regression, etc. For such tasks, the success of the classifier ensembles relies strongly on a good selection of the base learners and a strong *diversity* among the base learners. One of the well-known classifier ensembles for solving multiclass problems is the error-correcting output code (ECOC) [3]. ECOC decomposes a multiclass problem to a serial binary-class problems. Each binary-class problem is solved by some binary-class classifier, such as AdaBoost and support vector machine. Given a P class problem with a set of labeled samples $\{(\boldsymbol{\rho}_i, y_i)\}_{i=1}^n$ where $\boldsymbol{\rho}_i$ is a d dimensional sample, and $y_i \in \{1, 2, \ldots, P\}$ is the label of $\boldsymbol{\rho}_i$, the ECOC tries to use Q binary-class classifiers to address this problem. The relation between the classes and the classifiers can be expressed by a *code matrix* $\mathbf{M} \in \{-1, 0, 1\}^{P \times Q}$. An example of $\mathbf{M}$ is shown in Fig. 1 with $P = 4$ and $Q = 7$ [4].

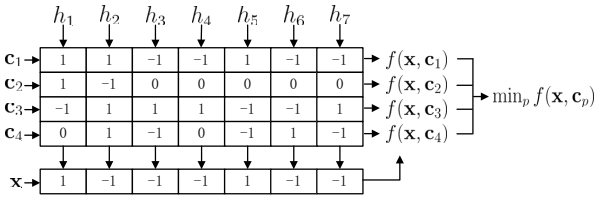

Fig. 1. Example of an ECOC code matrix $\mathbf{M}$ [4].

ECOC consists of two phases – coding and decoding. In the coding process, ECOC tries to find a code matrix $\mathbf{M}$ for the classifier training, where the p-th row of $\mathbf{M}$ expresses the codeword of the p-th class, denoted as $\mathbf{c}_p$, and the q-th column expresses the q-th classifiers, denoted as h_q. If the entry $m_{p,q} = 0$, it means that h_q does not take the data of the p-th class into classifier training [5]. In the decoding process, taking a test sample $\boldsymbol{\rho}$ into $h_1, \ldots, h_Q$ successively can get a test codeword of $\boldsymbol{\rho}$, denoted as $\mathbf{x} = [x_1, \ldots, x_Q]^T$. Given a decoding strategy $f(\mathbf{x}, \mathbf{c}_p)$, the prediction of $\boldsymbol{\rho}$ can be formulated as a minimization problem $\min_{\mathbf{c}_p \in \mathcal{M}} f(\mathbf{x}, \mathbf{c}_p)$, where $\mathcal{M} = \{\mathbf{c}_p\}_{p=1}^Q$ is the codeword set.

For the coding phase, there are generally two research directions for the codeword design. The first direction is the *problem-independent* coding design, such as the well-known one-versus-all (1vsALL) and one-versus-one (1vs1). The second direction is the *problem-dependent* coding design, which seems more promising and has attracted much

This work is supported in part by the National Natural Science Funds of China under Grant 61170197, in part by the China Postdoctoral Science Foundation funded project under Grant 2012M520278, in part by the Planned Science and Technology Project of Tsinghua University under Grant 20111081023, and in part by the subproject of the National High-Tech. R&D Key Project of China (863 Key Project) under Grant 2012AA011004.

attention. In [6], Pujol *et al.* proposed a problem-dependent coding method called discriminative ECOC (DECOC) . It embeds a binary decision tree to the coding design, where each node of the tree is a powerful bipartition of a multiclass problem. However, the decision tree has an intrinsic defect that if a test pattern is predicted wrongly by a father node, it will have no chance to be corrected by the child node. To overcome this drawback, in [7], Pujol *et al.* further proposed the ECOC optimizing node embedding (ECOC-ONE) algorithm. It starts with an initial ECOC classifier ensemble and iteratively adds the binary-class classifier that discriminates the most confusing pair of the classes to the ECOC ensemble until the desired performance is reached. This approach repairs the intrinsic defect of the decision tree and improves the performance directly by discriminating the most confusing pair. However, sometimes, the most confusing pair of the classes are so "stubborn" that we cannot even find a strong binary-class classifier on the pair. For this problem, in [8], Escalera *et al.* proposed to split the most confusing pair of the classes into several subclasses, such that a difficult learning problem can be decomposed to several easier subproblems. The subclass-ECOC method puts on a new scene to the ECOC study – microstructure analysis. However, the subclass-ECOC still uses a binary decision tree to construct the subclasses. Moreover, it has to control the scale of the subclasses, which might not be an easy job. Here comes the question, can we utilize the subclass technique for the most confusing pair without employing a tree structure?

For the decoding phase, recently, in [4], Escalera *et al.* introduced a weight matrix to the loss based (LB) decoding [5], which is known as the loss weighted (LW) decoding and has shown to be more powerful than traditional decoding methods. In [2], Zhang *et al.* further proposed the optimized weighted (OW) decoding. It tries to find a weight matrix that results in the minimal training risk.

In this paper, we propose a novel weight optimization and layered clustering-based ECOC (WOLC-ECOC). Specifically, we first employ a novel layered clustering-based (LC) approach to overcome the tree structure when we utilize the subclass technique. Then, we propose a wrapping-based ECOC that iterates the LC approach and the OW decoding. WOLC-ECOC makes the training risk degrade with iterations until the risk converges. Experimental results on several UCI datasets show that the new method is strongly competitive.

2. RELATED WORK

2.1. Optimized Weighted Decoding:

In [4], Escalera *et al.* presented that a good decoding strategy should make each class have the same decoding *dynamic range* and zero decoding *dynamic range bias*. Then, they proposed the LW decoding algorithm. The LW decoding introduces a predefined weight matrix $\mathbf{W} = [\mathbf{w}_1^T, \ldots, \mathbf{w}_P^T]^T = \begin{bmatrix} w_{1,1} \ldots w_{1,Q} \\ \vdots \quad \ddots \quad \vdots \\ w_{P,1} \ldots w_{P,Q} \end{bmatrix}$, which has the same size as $\mathbf{M}$ and satisfies the following constraints

$$w_{p,q} \begin{cases} = 0 & , \text{ if } m_{p,q} = 0 \\ \in [0,1], & \text{if } m_{p,q} \neq 0 \end{cases}, \forall p = 1, \ldots, P, \forall q = 1, \ldots, Q,$$

$$\sum_{q=1}^{Q} w_{p,q} = 1, \quad \forall p = 1, \ldots, P \tag{1}$$

We denote the set of all feasible weight matrices that are constrained by (1) as $\mathcal{W}$ ($\mathbf{W} \in \mathcal{W}$). The prediction function of the LW decoding algorithm is given by

$$\min_{\mathbf{c}_p \in \mathcal{M}} f_{LW}(\mathbf{x}, \mathbf{c}_p) = \min_{\mathbf{c}_p \in \mathcal{M}} \sum_{q=1}^{Q} w_{p,q} \ell(x_q c_{p,q}) \tag{2}$$

where $\ell(\cdot)$ is a user defined loss function. In this paper, we consider the linear loss function $\ell(\theta) = -\theta$. However, in [4], the authors did not mention how to get the optimal $\mathbf{W}$. They only take an empirical assignment according to the training accuracy of each dichotomizer.

To overcome the empirical assignment of the weight matrix, in [2], Zhang *et al.* proposed to optimize the weight matrix theoretically for the minimal training risk. The weight optimization is formulated as the following convex *linear programming* problem

$$\begin{aligned} \min_{\mathbf{W} \in \mathcal{W}, \mu, \xi_{i,p}} & -\mu + \frac{C}{n} \sum_{i=1}^{n} \sum_{p=1}^{P} \xi_{i,p} \\ \text{s.t.} & \mu \geq 0, \ \ \xi_{i,p} \geq 0, \\ & \mathbf{w}_p^T \mathbf{u}_{i,p} - \mathbf{w}_{y_i}^T \mathbf{u}_{i,y_i} \geq \mu \delta(i, y_i) - \xi_{i,p}, \forall i, \forall p \end{aligned} \tag{3}$$

where $\mathbf{u}_p = [\ell(x_1 c_{p,1}), \ldots, \ell(x_Q c_{p,Q})]^T$, μ is an unknown parameter, $\{\xi_{i,p}\}_{i,p}$ are called slack variables, C is a user defined constant, and $\delta(i, y_i)$ is defined as

$$\delta(i, y_i) = \begin{cases} 0, & \text{if } i = y_i \\ 1, & \text{otherwise.} \end{cases}$$

Problem (3) can be solved efficiently in time $\mathcal{O}(n \log n)$.

2.2. Layered Clustering-Based Approach:

The layered clustering-based (LC) approach [1] is a special classifier ensemble. It first splits the feature space into several different subspaces by clustering, where the classification problem in each subspace is further solved by a classifier. Then it repeats the above procedure several times. Each independent repeat is called a layer. The LC approach contains two complementary techniques. One is the clustering in each layer. It can identify overlapping patterns that are hard to differentiate. But it does not include any mechanism to incorporate the diversity (see the first paragraph of Section 1).

The other one is the layered approach. It uses the mechanism of the bagging and boosting to achieve the diversity between any layers for the weakness of the first property. The layered structure, as analyzed in [9], will improve the discriminability of a classifier ensemble on a binary-class problem.

3. THE PROPOSED ECOC

3.1. Layered Clustering-Based ECOC:

As analyzed in the introduction section, in [8], Escalera *et al.* proposed the subclass technique that splits a difficult classification problem to several easier sub-problems. Each subproblem is solved by an independent classifier. Finally, the difficult problem is solved by a classifier ensemble. However, they used a tree structure for the splitting. In order to inherit the advantage of the subclass technique, and meanwhile, to avoid using the decision tree for the subclass splitting, we investigate the *ensemble learning* for the solution as follows:

The key idea of the ensemble learning is to construct a strong diversity among the base classifiers. Generally, the methods of constructing the diversity can be grouped into four types [9]. They are the methods of 1) manipulating the training examples, 2) manipulating the input features, 3) manipulating the training parameters, and 4) manipulating the output targets. However, aside from manipulating the outputs of the binary classifiers which is the key idea of ECOC, the diversity has been seldom referred in the ECOC study yet. To our knowledge, only in [10], Prior and Windeatt manipulated different parameter settings of the base classifiers (multi-layer perceptrons).

In this paper, we propose the LC [1] based ECOC (LC-ECOC) for the aforementioned problem. LC-ECOC is presented as follows.

a) Take ECOC-ONE [7] as the base method. ECOC-ONE iteratively adds binary-class classifiers that discriminate the most confusing pairs of classes to the ECOC ensemble.

b) Whenever a "stubborn" pair is encountered, construct one layer classifier ensemble by LC for the pair, and continue the ECOC-ONE training. The "stubborn" pair means that the most confusing pair in the current iteration has been tried before by some classifier of the ECOC ensemble, but because the classifier is not strong enough, we encounter the problem again.

LC-ECOC has two advantages. First, unlike ECOC-ONE [7], the optimization process will not be blocked when a stubborn pair is encountered. Second, unlike the subclass-ECOC [8], if the subclass technique is utilized for the stubborn pair, the decision tree based subclass splitting can be prevented.

3.2. WOLC-ECOC:

However, the optimization process of LC-ECOC does not converge. Specifically, because adding a new classifier to the LC-ECOC does not guarantee the decrease of the training risk, we do not know when to stop adding new classifiers to the ensemble. Therefore, an empirical termination criterion has to be utilized by LC-ECOC (and also its previous ECOC-ONE).

Algorithm 1 WOLC-ECOC.

Initialization: Any initial ECOC ensemble, constant Z that controls the convergence behavior.

1: **repeat**
2: Run the OW decoding algorithm to find the optimal weight matrix of the current ECOC ensemble
3: Find the most confusing pair of classes
4: **if** the pair is not "stubborn" **then**
5: Train a simple classifier as ECOC-ONE [7]
6: **else**
7: Train one layer of classifier ensemble as LC
8: **end if**
9: Add the new classifier to the ECOC ensemble
10: **until** the training risk does not decrease for continuous Z iterations

Here, we propose a new algorithm, called WOLC-ECOC, by fusing the OW decoding and LC-ECOC coding methods. A simple description of the algorithm is presented in Algorithm 1. Step 2 of Algorithm 1 is rather important for the effectiveness of adding new classifiers and for the convergence behavior of WOLC-ECOC. It guarantees the non-increase of the training risk whenever we add any new classifiers to the ECOC ensemble. Therefore, if the training risk does not decrease in several iterations, the training procedure can be stopped. Note that if we delete Step 2 of Algorithm 1, the algorithm becomes LC-ECOC.

4. EXPERIMENTS

The data used for experiments consists of 8 multiclass datasets from the UCI Machine Learning Repository database. The properties of the UCI datasets are listed in Table 1.

Table 1. Descriptions of the UCI datasets. "n" is the dataset size, "d" is the dimension, "P" is the number of the classes.

ID	Data	n	d	P	ID	Data	n	d	P
1	Ecoli	336	7	8	5	Yeast	1484	8	10
2	Thyroid	215	5	3	6	Segmentation	2310	19	7
3	Vowel	990	10	11	7	OptDigts	5620	64	10
4	Balance	625	4	3	8	Vhicle	846	18	4

WOLC-ECOC is initialized by the 1vsALL coding, and the constant Z is set to 3. Discrete AdaBoost is used as the base classifier.

To show the effectiveness of the proposed method, we compare it with 5 state-of-the-art ECOC coding designs, including 1vs1, 1vsALL, Random [5], ECOC-ONE [7], and

Table 2. Accuracy comparison (%) of different ECOC coding-decoding methods on the UCI datasets. In each grid, the second line denotes the corresponding decoding method of the accuracy.

ID	Data	1vs1	1vsALL	Random	ECOC-ONE	DECOC	WOLC-ECOC
1	Ecoli	85.00±0.00 (2) HD	81.27±0.00 (3) HD	77.52±1.36 (6) LW	80.17±1.18 (4) HD	78.47±2.37 (5) LW	**87.40±0.82** (1) OW
2	Thyroid	93.45±0.00 (6) HD	93.95±0.00 (3) HD	94.57±0.92 (2) HD	93.95±0.00 (3) HD	93.93±0.72 (5) LW	**95.45±0.00** (1) OW
3	Vowel	58.74±0.00 (2) HD	45.97±0.00 (4) LW	40.99±1.95 (6) LW	46.50±1.49 (4) LW	45.80±1.99 (5) LB	**60.61±0.82** (1) OW
4	Balance	86.56±0.00 (4) LW	87.67±0.00 (2) LW	87.55±1.53 (3) LW	77.81±0.00 (5) LW	76.70±0.00 (6) HD	**88.97±0.40** (1) OW
5	Yeast	53.99±0.00 (3) LW	54.06±0.00 (2) LW	45.50±1.51 (6) LW	50.53±0.81 (4) LW	50.53±0.99 (4) LW	**56.28±0.18** (1) OW
6	Segmentation	95.31±0.00 (2) LW	93.06±0.00 (5) LW	92.43±1.18 (6) LW	94.20±0.00 (3) LW	93.37±0.00 (4) HD	**95.60±0.39** (1) OW
7	OptDigts	95.28±0.00 (2) LW	84.09±0.00 (4) LW	74.69±0.69 (6) LW	86.03±0.00 (3) LW	75.27±0.00 (5) HD	**95.67±0.13** (1) OW
8	Vehicle	73.52±0.00 (3) LW	72.33±0.00 (6) LW	73.07±0.94 (4) LW	72.35±0.32 (5) LW	74.28±1.04 (2) LW	**75.41±0.13** (1) OW

Table 3. Code length comparison of different ECOC methods on the UCI datasets.

ID	Data	1vs1	1vsALL	Random	ECOC-ONE	DECOC	WOLC-ECOC
1	Ecoli	28	8	10	9.48±0.13	7	14.75±2.01
2	Thyroid	3	3	10	6.63±0.00	2	3.00±0.00
3	Vowel	55	11	10	12.10±0.11	10	26.64±0.58
4	Balance	3	3	10	8.00±0.00	2	15.16±1.96
5	Yeast	45	10	10	12.73±0.29	9	13.30±0.63
6	Segmen.	21	7	10	8.25±0.00	6	13.18±2.05
7	OptDigts	45	10	10	11.00±0.00	9	22.45±5.62
8	Vehicle	6	4	10	5.42±0.43	2	10.96±0.68

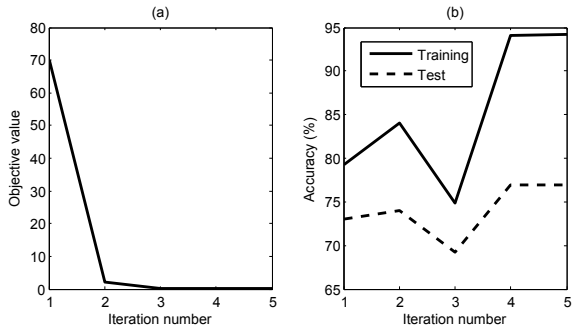

Fig. 2. Convergence behavior of WOLC-ECOC on the Vehicle dataset. (a) Convergence behavior of the training risk (objective value). (b) Curves of the training and test accuracies.

DECOC [6]. Each of the competitive coding methods combines with 3 decoding methods, including Hamming distance (HD) decoding, LB decoding [5], and LW decoding [4]. Therefore, we totally compare WOLC-ECOC with 15 different coding-decoding method pairs. We run each pair of the coding-decoding methods 10 times and record the average experimental results. For each time, we apply a stratified sampling and ten-fold cross-validation, and test for confidence interval at 95 with a two-tailed t test.

Tables 2 and 3 list the accuracy and code length comparisons. From Table 2, it's clear that the proposed algorithm is more effective than 15 referenced methods. From Table 3, we can see that although the code length of the proposed method is longer than 1vsALL, DECOC, and ECOC-ONE, it is much shorter than 1vs1. Generally, it's worth of using a little longer code length for a much higher accuracy.

Fig. 2 gives an example of the convergence behavior of the training risk of WOLC-ECOC. From Fig. 2 (a), we can see that the training risk decreases with respect to the iteration numbers. Note that, from Fig. 2 (b), we can observe that the accuracy is not always improved. This is because that the training risk we optimize here is not rigourously equivalent to the accuracy. Anyhow, we can still see that the accuracy is generally improved.

5. CONCLUSIONS

In this paper, we have presented a new ECOC method, called WOLC-ECOC. It iterates two novel parts until the training risk converges. The first part is the LC approach. It inherits the advantage of the subclass splitting technique for the "stubborn" problem without utilizing the decision tree for the subclass splitting. The second part is the OW decoding. it guarantees the decreasing of the training risk with respect to the iterations. Finally, the experimental results on the UCI datasets have shown that the WOLC-ECOC algorithm is more effective than 15 referenced coding-decoding pairs.

6. REFERENCES

[1] A. Rahman and B. Verma, "Novel layered clustering-based approach for generating ensemble of classifiers," *IEEE Trans. Neural Netw.*, vol. 22, no. 5, pp. 781–792, 2011.

[2] X. L. Zhang, J. Wu, Z. P. Chen, and P. Lv, "Optimized weighted decoding for error correcting output codes," in *Proc. Int. Conf. Acoustic, Speech, Signal Process.*, 2012, pp. 2101–2104.

[3] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *J. Artif. Intell. Res.*, vol. 2, pp. 263–286, 1995.

[4] S. Escalera, O. Pujol, and P. Radeva, "On the decoding process in ternary error-correcting output codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 120–134, 2010.

[5] E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," *J. Mach. Learn. Res.*, vol. 1, pp. 113–141, 2001.

[6] Oriol Pujol, Petia Radeva, and Jordi Vitria, "Discriminant ecoc: A heuristic method for application dependent design of error correcting output codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 6, pp. 1007–1012, 2006.

[7] O. Pujol, S. Escalera, and P. Radeva, "An incremental node embedding technique for error correcting output codes," *Pattern Recogn.*, vol. 41, no. 2, pp. 713–725, 2008.

[8] S. Escalera, D. M. J. Tax, O. Pujol, P. Radeva, and R. P. W. Duin, "Subclass problem-dependent design for error-correcting output codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 6, pp. 1041–1054, 2008.

[9] T. G. Dietterich, "Ensemble methods in machine learning," in *Proc. Multiple Classifier Syst.*, 2000, pp. 1–15.

[10] M. Prior and T. Windeatt, "Over-fitting in ensembles of neural network classifiers within ecoc frameworks," in *Proc. Multiple Classifier Syst.*, 2005, pp. 834–844.