Research Article

Weighted Decoding for the Competence Reliability Problem of ECOC Multiclass Classification

Lei Lei [ID][1] **and Yafei Song** [ID][2]

[1]*College of Information and Navigation, Air Force Engineering University, Xi'an 710077, China*
[2]*College of Air and Missile Defense, Air Force Engineering University, Xi'an 710051, China*

Correspondence should be addressed to Yafei Song; yafei_song@163.com

Academic Editor: Mario Versaci

Error-Correcting Output Codes has become a well-known, established technique for multiclass classification due to its simplicity and efficiency. Each binary split contains different original classes. A noncompetent classifier emerges when it classifies an instance whose real class does not belong to the metasubclasses which is used to learn the classifier. How to reduce the error caused by the noncompetent classifiers under diversity big enough is urgent for ECOC classification. The weighted decoding strategy can be used to reduce the error caused by the noncompetence contradiction through relearning the weight coefficient matrix. To this end, a new weighted decoding strategy taking the classifier competence reliability into consideration is presented in this paper, which is suitable for any coding matrix. Support Vector Data Description is applied to compute the distance from an instance to the metasubclasses. The distance reflects the competence reliability and is fused as the weight in the base classifier combination. In so doing, the effect of the competent classifiers on classification is reinforced, while the bias induced by the noncompetent ones is decreased. Reflecting the competence reliability, the weights of classifiers for each instance change dynamically, which accords with the classification practice. The statistical simulations based on benchmark datasets indicate that our proposed algorithm outperforms other methods and provides new thought for solving the noncompetence problem.

1. Introduction

Multiclass classification has been an open issue in machine learning and many solutions are possible. As a "divide-and-conquer" framework, Error-Correcting Output Codes (ECOC) [1, 2] can realize the class decomposition and dichotomize ensemble effectively through a binary or ternary matrix, which not only simplifies the complexity of pattern recognition but also enables the classic binary classifiers suitable for multiclass classification. So far, ECOC has been widely applied to biological data recognition [3, 4], disease diagnosis [5–7], military target recognition [8], and intelligent transportation systems [9] with fairly good recognition performance.

The procedures of using ECOC to solve the multiclass problems are usually divided into three steps: coding, base classifier training, and decoding.

The goal of coding is to construct a binary or ternary matrix $\mathbf{M} = (m_{ij})_{c \times l}, m_{ij} \in \{1, 0, -1\}$, where the rows hold the codewords and the columns represent the binary splits. The coding methods can be categorized into three kinds based on the existing researches: predefined code, data-dependent code, and dichotomies-based code. Independent of the specific application and instances, the predefined code ignores the potential information of the original classes and confines the improvement of classification performance. The dichotomies-based code involves finding an optimal matrix given a set of binary classifiers, which is proven to be an NP-complete problem by Crammer and Singer [10]. However, the data-dependent code can make the best of the class separability to enhance performance as a whole, which has drawn special attention. The classic data-dependent codes contain Discriminant ECOC [11], Subclass ECOC [12], and HECOC [13]. In order to construct a data-driven matrix and

enhance the diversity between dichotomizers, Zhou et al. [14] proposed a Confusion Matrix Superclass ECOC (CMECOC) based on the Fisher criterion. CMECOC used "one-versus-all" method to encode classes between the superclasses to avoid the unnecessary redundancy and "one-versus-one" method among the superclasses to reduce the complexity of decision boundaries. More research studies on data-driven ECOC are available [15–21], all of which promote the ECOC development.

The second step is base classifier training. The most common practice is to train the classifiers (decision functions) with the binary splits. The outputs of the base classifiers are merged in the final decoding step. It is generally acknowledged that there are three categories of decoding strategies according to the existing literatures. The first type is those based on the distance between the predicted codeword and the target codeword. The state-of-the-art decoding strategies of this type contain the Hamming decoding, Euclidean decoding, loss-based decoding, and so on. Zhou et al. [22] proposed a new weighted decoding by using genetic algorithm to learn the weight coefficient matrix and taking the final generalization error of the ensemble classifiers as fitness function to reduce the error caused by the consistent-diverse balance problem. However, the predicted codewords of the base classifiers are hard (either 1 or −1), which may be not suitable for the cost-sensitive classification. On this account, Lei et al. [23] introduced the reject option into coding phase to extend a binary-symbol output to a three-symbol one and used the modified Hamming decoding strategy for decoding.

The second type is those based on class posterior probability estimation, which estimates the class membership based on the decomposition framework of ECOCs. Suppose that the base classifier $h(x)$ outputs the class probability for a given test sample $\mathbf{x}$; then the probability vector can be obtained as follows:

$$\mathbf{Y} = \left(P\left(h_1\left(\mathbf{x}\right)=1\right), P\left(h_2\left(\mathbf{x}\right)=1\right), \ldots, P\left(h_L\left(\mathbf{x}\right)=1\right)\right), \tag{1}$$

where L is the number of $\mathbf{M}$ columns. Then the relationship between the coding matrix, probability vector, and the posterior probability $\mathbf{p} = \left(P\left(c_1 \mid \mathbf{x}\right), \ldots, P\left(c_k \mid \mathbf{x}\right)\right)$ can be described as

$$\mathbf{M}^T \mathbf{p} = \mathbf{Y}, \tag{2}$$

where k is the number of classes. Many research works have been done based on the equation. Zhou et al. [24] put forward a variation of the product rule and a linear rule based on posterior probabilities for ternary ECOC. The authors refined the decomposition process of the probability to get smoother estimates in the product rule and extend the linear rule in binary ECOC to ternary ones. Simeon et al. [25] designed the reject rules in decoding with an external and internal approach. The third type is those based on the analysis of the pattern space [26].

As Prior and Windeat et al. [27] pointed out, it was exactly the diversity inherent to ECOC decomposing framework that makes multiclass classification based on ECOC efficient. It is known that, due to the third symbol, ternary ECOC becomes more versatile and robust. However, when using ternary ECOC, we usually need to face the contradiction between a base classifier and the testing samples whose real class is not used to learn the classifier. We call this contradiction the noncompetence problem. On this account, Galar et al. [28] proposed a dynamic classifier selection strategy for the specific one-versus-one matrix that tried to avoid the noncompetent classifiers. The strategy considered the neighborhood of each instance to decide whether a classifier is competent or not. Furthermore, they developed a distance-based combination strategy to reduce the effect the noncompetence contradiction. In the strategy, the degree of the base classifiers' competence reliability was measured with the distance from the instance to each class [29].

However, Galar et al.'s work mainly concentrated the noncompetence problem on the predefined code (one-versus-one scheme) [30]. With the development of coding, most of the ternary ECOC are data-driven matrices with high base classifier diversity and low column redundancy. How to reduce the error caused by the noncompetence contradiction for the data-driven matrices under diversity big enough is the breakthrough of the ECOC classification. To this end, we propose a new weighted decoding strategy by considering the competence reliability (WCR) to decrease the error caused by the noncompetent classifiers. In the proposed algorithm, the SVDD is used to measure the closeness of an instance to each class, which can be regarded as the estimates of the competence reliability of the base classifiers for an instance. Then, we can get the weight coefficient matrix, in which the weights of the competent classifiers are higher and those of the noncompetent ones are lower. In so doing, the influence of the noncompetence contradiction on classification can be weakened or eliminated.

The outline of the paper is organized as follows: a brief introduction of the noncompetence problem and SVDD are discussed in Section 2. Section 3 focuses on the proposed WCR weighted algorithm. Experiments and results are summarized in Section 4 and Section 5 presents conclusions.

2. Noncompetence Problem

According to the pattern recognition theory, a good classification algorithm always demands for the same distribution of training samples and testing samples. Some original classes are denoted by zero when a ternary ECOC is used for classification. The testing samples and training samples for a base classifier may not belong to the same metasubclass. Therefore, a classifier is noncompetent for an instance whose real class does not belong to the metasubclass used to learn the classifier. It is a prior question to estimate whether a base classifier is competent or not for an instance, which is equivalent to the classification problem itself. However, the outputs of noncompetent classifiers are merged in the decoding step equivalently. The existing data-driven matrices possess big diversity and fewer columns. The

bias induced by the noncompetent classifiers may yield a decisive error for the final decoding results.

Taking the analysis above into consideration, to reduce the error induced by the noncompetent classifiers, it is crucial to determine how to estimate the competence reliability of a classifier for a testing instance. In order to achieve this goal, we present a new WCR decoding strategy to balance the noncompetence contradiction. As for the weighted decoding, Hüllermeie and Vanderlooy [31] provided a sound theoretical demonstration from the perspective of the optimal adaptive voting strategy. Zhou et al. [22] also acknowledged the significance of the weighted method and advocated that different weighted schemes led to different decoding results and were more important than decoding strategies themselves under some circumstances. The proposed WCR decoding can estimate the competence reliability by learning the weight coefficient matrix based on the column outputs. The ECOC framework based on WCR weighted decoding is described in Figure 1.

X is the training set with m dimensions and n samples. f denotes the base classifier. $h(\cdot)$ is the base classifier output. $(w_{ij})_{k \times L}$ is the weight coefficient matrix, in which k and L are the numbers of the classes and columns, respectively. The traditional ECOC classification is shown in the dotted box. $h(\cdot)$ can be trained with the binary splits based on the ternary matrix where the white rectangles represent 1, the black ones represent −1, and the gray ones are 0, which means the corresponding classes are not used to train the base classifier. The weight coefficient matrix can be obtained according to the decoding rule **D** and weight method. The final results can be achieved by the following formula:

$$\text{Arg} \max_j \sum_{j=1}^{L} w_j D\left(h_j(T)\right). \tag{3}$$

The key of WCR ECOC is how to calculate the weight coefficient matrix which reflects the competence reliability of the classifier for an instance. As we know, the closer an instance to a class, the more likely it belongs to, the more competent the base classifier trained with the corresponding class. So the question is how to measure the distance between an instance and each class.

First proposed by Tax and Robert [32], SVDD is an effective and commonly used single-class classification mechanism. SVDD obtains a spherically shaped boundary around a dataset $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\} \subset \mathbf{R}^m$ and can be made flexible by using kernel functions to project the original nonlinear data to high-dimensional feature space. Featured by the center **o** and radius $r > 0$, the hypersphere covers all of the target data and superfluous space as little as possible, which can be used to detect nontarget data or outliers to realize two-class classification.

The hypersphere can be obtained by minimizing r^2 with the constraint $\|\mathbf{x}_i - \mathbf{o}\|^2 \leq r^2$. In order to make the optimal zone more compact, according to the kernel projection notation, the low-dimensional input space F can be reflected into high-dimensional feature space H by nonlinear function Φ. The inner product operation in H can be replaced by the kernel function satisfied with Mercer constraints, which

means finding the kernel function $K(x, y)$ subjected to $K(x, y) = \langle \Phi(x), \Phi(y) \rangle$; after that, fix the hypersphere in H [33]. Allowing for the possibility of outliers, the distance from $\mathbf{x}_i$ to o should not be strictly smaller than r^2, but a larger distance should be penalized. Therefore, the slack variable $\xi_i \geq 0$ is introduced and the minimization problem can be rewritten as follows:

$$\min_r \ r^2 + C \sum_i \xi_i \tag{4}$$

$$\text{s.t} \quad \|\mathbf{x}_i - \mathbf{o}\|^2 \leq r^2 + \xi_i, \quad \xi_i \geq 0, \ i = 1, 2, \ldots, N.$$

Parameter C controls the trade-off between r and the number of outliers. The optimal problem can be solved by changing into a dual problem with Lagrange multipliers:

$$\max \ \sum_i \alpha_i K(\mathbf{x}_i, \mathbf{x}_i) - \sum_{i,j} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{5}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq C, \sum_i \alpha_i = 1, \quad j = 1, 2, \ldots, N.$$

Objects $\mathbf{x}_i$ with $0 < \alpha_i \leq C$ are called the support vectors of the SVDD.

For a testing sample **x**,

$$f(x) = \|\mathbf{x} - \mathbf{o}\|^2 = K(\mathbf{x}, \mathbf{x}) - 2\sum_{i=1}^{N} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + \sum_i^{N} \sum_j^{N} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j). \tag{6}$$

If $\|\mathbf{x} - \mathbf{o}\|^2 \leq r^2$, object **x** is recognized as the target class, and vice versa.

3. Weighted Decoding for the Competence Reliability Problem

3.1. Construction of Weight Coefficient Matrix. Consider a k-class classification case and the training dataset $\{X_1, \ldots, X_k\}$, in which $X_i = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_i}\}, i = 1, \ldots, k$, and N_i is the number of each class. $N = \sum_{i=1}^{k} N_i$.

The first step of the WCR weighted decoding is to use SVDD to compute the distance from an instance to each class. The SVDD hyperspheres $S = \{S_1, \ldots, S_k\} = \{(r_1, \mathbf{o}_1), (r_2, \mathbf{o}_2), \ldots, (r_k, \mathbf{o}_k)\}$ for each class are achieved with X_i, respectively, by using the kernel function. The Euclidean distance between an instance and the ith class is computed with $d_i = \sqrt{\|\mathbf{x} - \mathbf{o}_i\|}$. The distance vector can be obtained as $\mathbf{D} = (d_1, d_2, \ldots, d_k)$. We may make the following assumption: an instance belongs to a class if the distance is equal to or smaller than the radius, and vice versa. $\overline{D} = (d_1/r_1, d_2/r_2, \ldots, d_k/r_k)$ for belonging preference relation is identified by normalizing the distance vector. From the analysis above, we can see that if d_i/r_i is not more than 1, the instance is likely to belong to $class_i$. The smaller d_i/r_i, the larger the probability, while the larger the ratio, the smaller the probability.

Given a ternary matrix $\mathbf{M} = (m_{ij})_{c \times l}$, $m_{ij} \in \{1, 0, -1\}$, if $M_{ij} = 0$, $class_i$ is omitted in the jth base classifier training. If an instance belongs to $class_i$, then the base classifier is noncompetent for the instance, whose outputs have little
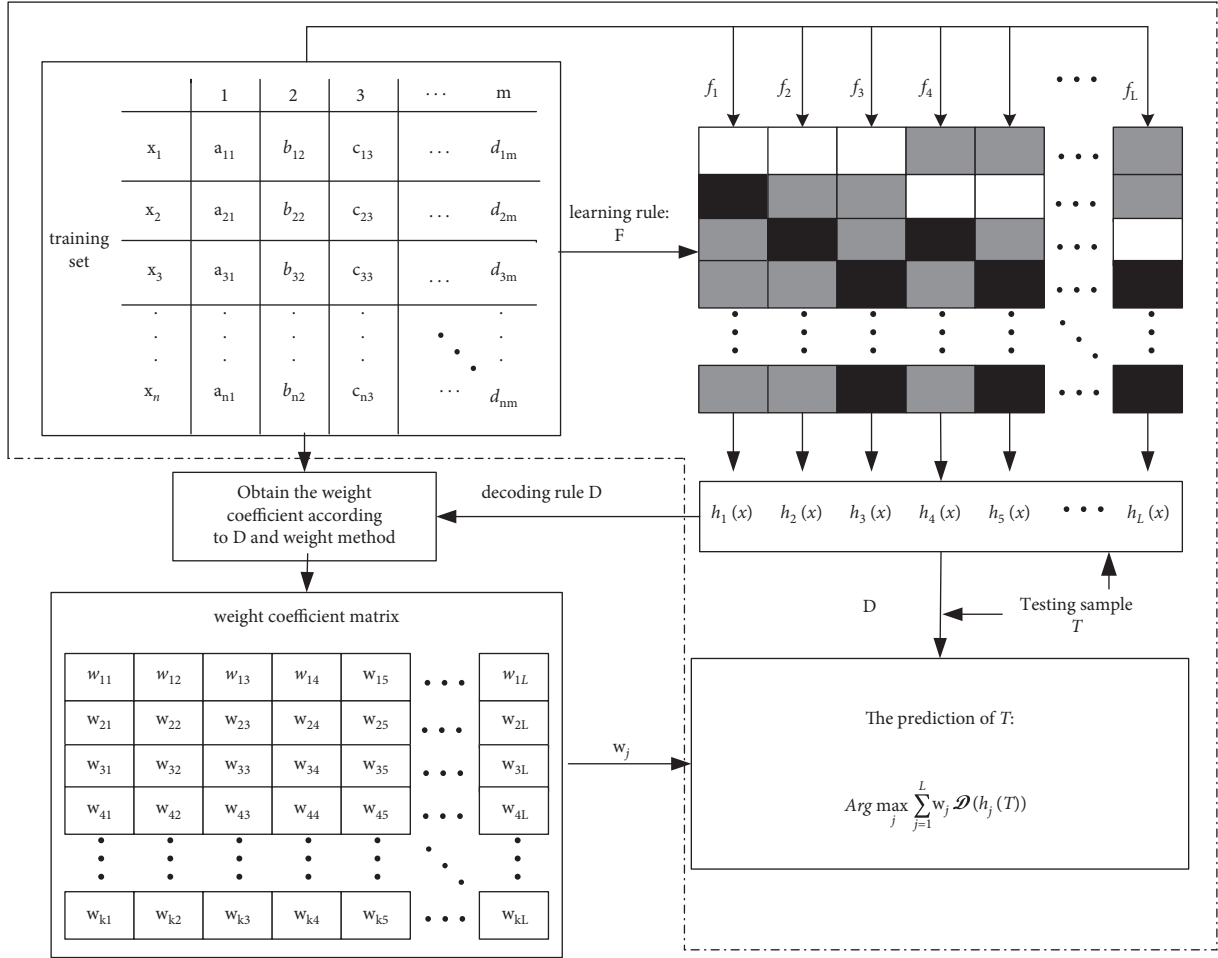
FIGURE 1: The ECOC framework based on WCR weighted decoding.

guidance for classification prediction. Therefore, the influence of the jth classifier should be weakened when the instance is classified. The WCR decoding relearns the weight of the base classifier based on competence reliability by using the following formula:

$$w_{ij} = \frac{d_i}{r_i}, \text{ if } M(i, j) = 0. \tag{7}$$

It is worth noting that more than one class will be ignored during the training, so the final weight of base classifier is obtained:

$$w_j = \sum_{i=1, M(j,i)=0}^{N_i} \frac{d_i}{r_i}. \tag{8}$$

From equation (7), we may draw the following conclusions.

Given an instance, whose real class label is class$_i$, the distance from the instance to class$_i$ is the shortest. If class$_i$ is ignored during the base classifier training, the weight of the corresponding base classifier in the decoding phase becomes smaller because $d_i/r_i \leq 1$ and it is noncompetent. The more the classes ignored, the smaller the weight and the less competent the base classifier. According to the

aforementioned considerations, the WCR weighted strategy is listed in Algorithm 1.

In order to elaborate the proposed method, the Balance benchmark dataset, 625 by 4 dataset with 3 classes from the University of California at Irvine (UCI) repository, is taken as an example. The numbers of the training and testing sets in one of the cross-validation folds are 499 [39 230 230] and 126 [58 10 58], respectively. The hypersphere configurations obtained by the proposed method during the training phase are indicated in Table 1.

In Table 1, σ is the value of radial basis function used in SVDD, α_i is the solution to equation (4), and threshold is the radius range from the support vectors to the hypersphere center.

Consider a testing instance [1 5 2 4] belonging to class$_1$. According to Algorithm 1, the distance vector **D** is (0.5417, 0.6978, 0.6519). If the ternary matrix is one-versus-one matrix, the weights of three base classifiers are $0.6519/0.4652 = 1.4013$, $0.6978/0.4482 = 1.5569$, and $0.5417/0.6003 = 0.9024$, respectively. The weight of the third base classifier is the smallest because class$_1$ does not take part in the base classifier training, so the third base classifier is noncompetent for the testing instance and the weight is smaller than the other two competent classifiers.

Input: any data-driven ternary matrix $M \in \{-1, 0, +1\}^{k \times L}$, training set $\mathfrak{I}$, testing sample x
Training phase
Step 1: use coding matrix $\mathbf{M}$ and train l base classifiers $f_j (j = 1, 2, \ldots, L)$;
Step 2: apply SVDD to get the hypersphere $(r_i, \mathbf{o}_i)$ $i = 1, \ldots, k$, for each class;
Testing phase
Step 3: for a testing instance $\mathbf{x}$, compute the distance to each class to get the $\overline{D}$ distance matrix;
Step 4: use equations (6) and (7) to calculate the weights of base classifiers if $\mathbf{M} (i, j) = 0$;
Step 5: normalize $\mathbf{w}$;
Output: base classifier weight vector $\mathbf{w}$

ALGORITHM 1: The weight coefficient matrix construction based on SVDD.

TABLE 1: Hypersphere configuration for each class.

	$Class_1$	$Class_2$	$Class_3$
σ	5	5	5
α_i	8×1	30×1	30×1
Threshold	0.6003	0.4486	0.4653
Support vectors	8×4	30×4	30×4

After constructing the weight coefficient matrix, we can get the final results based on decoding strategy.

$$\text{Arg} \max_j \sum_{j=1}^{L} w_j D\left(h_j(T)\right). \tag{9}$$

3.2. Computational Complexity of the Proposed Weighted Decoding Strategy. Compared to the traditional ECOC classification, the time of our proposal is mainly consumed in constructing SVDD hyperspheres, which needs to solve the quadratic programming problem. The testing set has a complexity of $O(m * N)$ to compute the distance from an instance to the center, where N is the number of samples and m is the dimension of attributes. Therefore, the complexity has not obviously increased. The computation time and storage are acceptable.

The approach of the weighted decoding for competence reliability comes into being. Next, we will validate the performance of its application in the classification through experiments.

4. Experiments and Comparisons

4.1. Experiment Data. In this section, we validate the proposed algorithm by using benchmark datasets. The characteristics of the UCI datasets are listed in Table 2. Certain UCI datasets are normalized by deleting the small samples. In the meanwhile, the principal component analysis is applied to reduce the dimensionality to promote classification speed.

4.2. Experimental Design. Three kinds of experiments are executed for validating the proposed method in this paper.

Firstly, the benchmark datasets are used to evaluate the classification performance of the WCR decoding by comparing the results with four classic decoding strategies: Hamming decoding (HD), inverse Hamming decoding (IHD), linear loss function-based decoding (LLB), and exponential loss function-based decoding (ELB). These strategies all belong to the first type of the decoding category.

Secondly, we compare the results of the weighted Hamming and LLB decoding based on error rate, class separability, and genetic algorithm [26], respectively. Five different ternary matrices, one-versus-one, spares random, subclass ECOC [12], HECOC based on SVDD [30], and CMECOC [22], are adopted in the experiments. When selecting the random code, we pick up the wanted matrix at random in the matrix set of 2000 sparse random matrices (whose probability of each code word is $p(-1) = 1/3$, $p(0) = 1/3$, and $p(+1) = 1/3$).

Finally, the WCR weighted method is compared to the distance-based relative competence weighting (DRCW) algorithm based on k-nearest neighbors [29] to evaluate the efficiency of estimating the competence reliability and solving the noncompetence problem. The one-versus-one coding matrix is adopted for consistency.

Decision tree and support vector machine (SVM) base classifiers are adopted in comparison, whose parameter configuration is shown in Table 3.

To evaluate the performance of different experiment results, we apply stratified tenfold cross-validation and test for the confidence interval at 95 percent with a two-tailed test if the number of the pieces of sample data is larger than 500. Otherwise, we apply stratified fivefold cross-validation [34] and the calculation formula is given by

$$\frac{|\overline{x} - \mu|}{\sigma/\sqrt{n}} \geq t_{0.025}(n - 1), \tag{10}$$

where μ and σ indicate the mean and variance, respectively, n is the number of fold validations, and $t_{0.025}(4) = 2.7764$ and $t_{0.025}(9) = 2.2622$.

4.3. Experimental Results and Analysis

4.3.1. Comparison of Classification Performance. We compare the classification results of the WCR decoding with those of the original decoding strategies based on one-versus-one matrix and CMECOC. The former belongs to the predefined-based matrix and the latter belongs to the data-driven type. Figure 2 shows the classification accuracy of

TABLE 2: The description of benchmark datasets.

Dataset	Cases	Classes	Atts	Features		
				C	B	N
Ecoli	336	8	7	7	—	0
Glass	214	6	10	9	—	−1
Iris	150	3	4	4	—	—
Satimage	6435	6	36	36	—	—
Segment	2310	7	19	19	—	—
Vehicle	846	4	18	18	—	—
Vowel	990	11	13	13	—	—
Wine	178	3	13	13	—	—
Yeast	1484	10	8	8	—	—
Zoo	101	7	16	1	15	—

Features: C, continuous; B, binary; N, nominal.

TABLE 3: The parameter configuration.

Algorithm	Parameters
SVM_{poly}	$C = 1.0$
	Tolerance parameter = 0.001
	Epsilon = $1.0E-12$
	Kernel type = polynomial
	Fit logistic models = true
SVDD [33]	Fracrej = 0.05
	Kernel function = RBF
	Sigma = 5
Treec	Maxcrit = purity
	Prune = 0 no pruning

these decoding strategies mentioned before based on decision tree classifier.

From the results, we can see that the WCR decoding mechanism outperforms the original decoding ones on most of the datasets, which proves that the weighted decoding for competence reliability problem indeed has a positive influence on the classification performance regardless of the coding matrix. From the other way around, the idea of considering the noncompetence contradiction for the base classifier has practical significance. Effective and feasible, the WCR decoding possesses a remarkable promotion by evaluating the distance from an instance to each class based on SVDD and shifting the distance information into weight to merge in decoding phase.

Figure 3 shows the classification accuracy of four different kinds of the weighted decoding strategies based on class separability (SW), error rate (EW), genetic algorithm (GW), and competence reliability (WCR), respectively.

In order to evaluate the performance comprehensively, Figure 4 shows the classification error for three data-driven matrices based on linear loss-based decoding and the corresponding weighted mechanism. From Figures 3 and 4, we can see that the green line representing the WCR strategy is on the top for accuracy and on the bottom for error on most of the datasets, illustrating that the decoding strategy based on the competence reliability performs better than the rest.

Different from the weighted decoding strategies based on error rate, class separability, and genetic algorithm, in which the weights for base classifier are static, the weights

in WCR mechanism dynamically change along with the distance from the testing instances to each class. In the meanwhile, the weights in classic methods are obtained in the testing step. The weights in WCR are got throughout the testing phase, which is much close to the real classification practice.

It is worth noting that some other weighted decoding algorithms perform the best sometimes. This can be explained by the fact that the corresponding dataset has a balanced distribution. Compared to the WCR decoding algorithm, there is only a little difference in the weight matrix got by error rate and class separability, which can still achieve the sound classification performance under certain situation suitable for their starting points.

Figure 5 shows the runtime of different weighted decoding strategies. The computation complexity of WCR focuses on the hypersphere construction. Once the hyperspheres are built, short testing time is required, so the computation complexity is acceptable. In the meanwhile, the weighted decoding based on GW possesses the longest runtime because the weights are obtained by solving an optimal problem. The runtime of weighted decoding based on error rate is the shortest, which can be explained by the fact that the weights based on error rate can be got directly from the training accuracy.

4.3.2. Comparison of Competence Reliability Evaluation. In order to validate the efficiency of our proposal further, the WCR mechanism based on SVDD is compared with the DRCW-OVO based on k-nearest neighbors. The coding is one-versus-one matrix, and SVM and decision tree are chosen as base classifiers, where $k = 3 *$ number of class. Table 4 lists the accuracy of two methods of evaluating competence reliability, where the bold face denotes the best classification accuracy.

From Table 4, we can see that the accuracy of the weighted decoding based on WCR wins over DRCW twelve times in total 32-time experiment.

In order to get the statistical comparison, we calculate the average ranks as 1.75 and 1.25 for DRCW and WCR, respectively. The Nemenyi test [35] is used further to test the significant difference between the two methods. The critical difference value $CD = q_a \sqrt{k(k+1)/6J} = q_{0.05} \sqrt{2 \times 3/6 \times 16}$
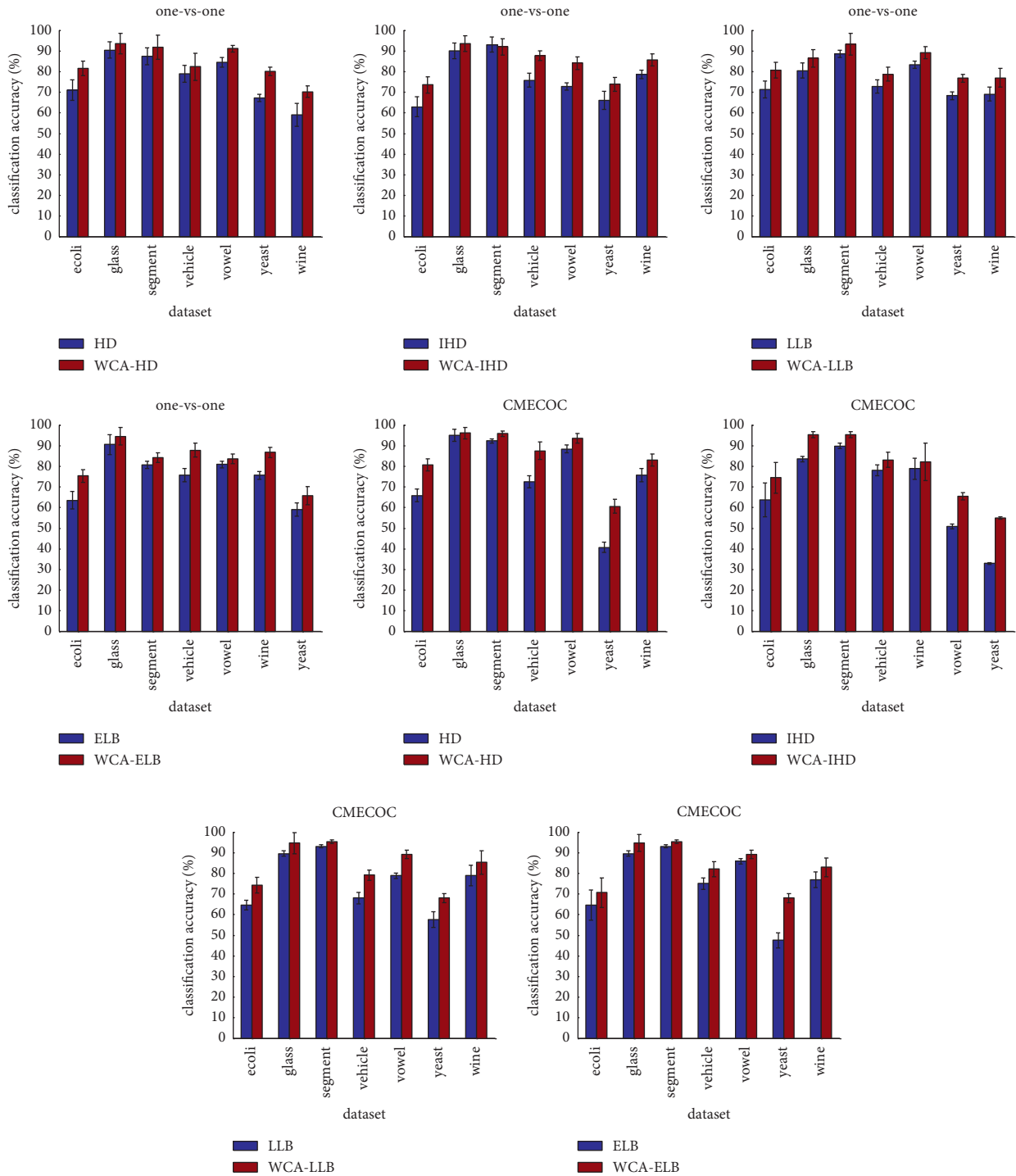
FIGURE 2: Comparison of four original decoding strategies and the corresponding WCR mechanism.
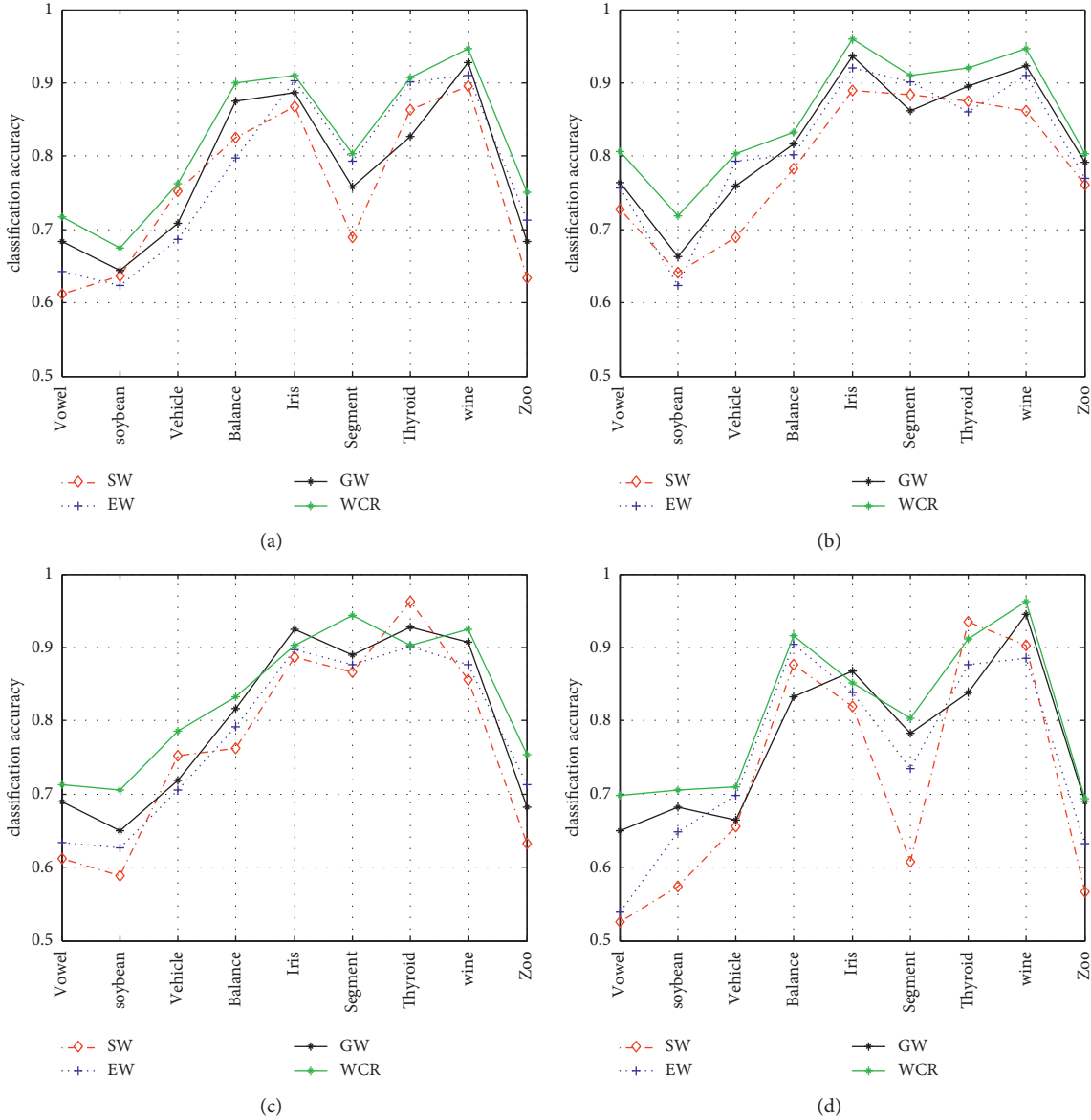
FIGURE 3: Classification accuracy of four different weighted decoding strategies. (a) Different weighted methods based on Hamming decoding and OVO. (b) Different weighted methods based on LLW decoding and OVO. (c) Different weighted methods based on Hamming decoding and sparse matrix. (d) Different weighted methods based on LLW decoding and sparse matrix.

= 0.49, where q_a is the Studentized Range Statistic and k and J are the number of methods to be compared and experiment time for each method, respectively. We can see that the average rank of WCR and the difference are both larger than CD, which means the difference between the two methods is significant and the performance of WCR with lower rank is better.

This corroborates the efficiency of WCR in dealing with the noncompetence problem. In the contrast with k-nearest neighbors method, the distance obtained by SVDD hyperspheres is more comprehensive for the reason that it takes all the class samples into consideration. In summary, the idea of the weighted decoding based on WCR provides new sparks and thought for solving noncompetence problem.
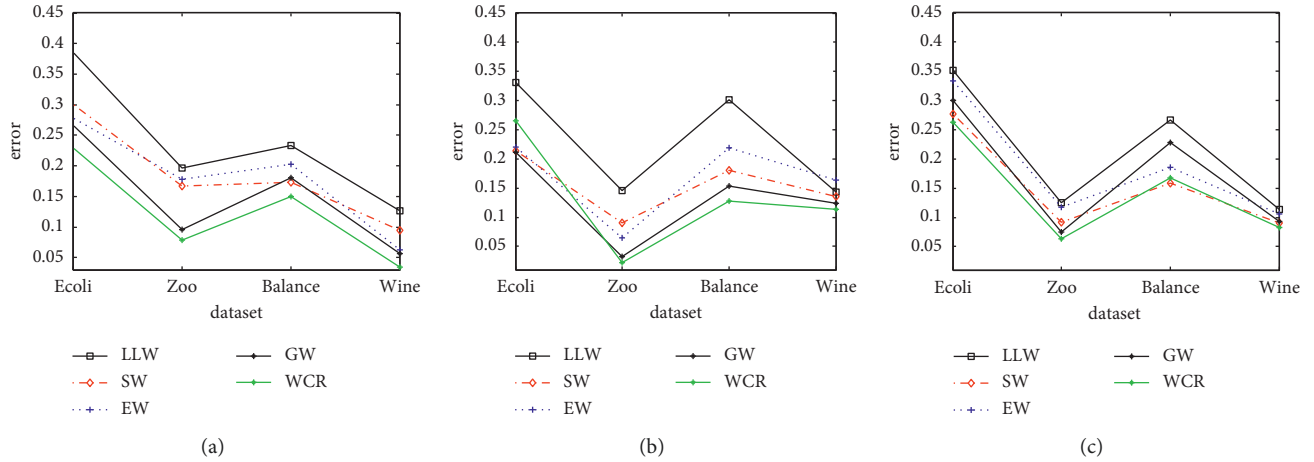
FIGURE 4: Classification error of certain datasets based on three data-driven matrices and LLW decoding strategy. (a) Different weighted decoding based on SECOC and LLW decoding. (b) Different weighted decoding based on CMECOC and LLW decoding. (c) Different weighted decoding based on HECOC and LLW decoding.
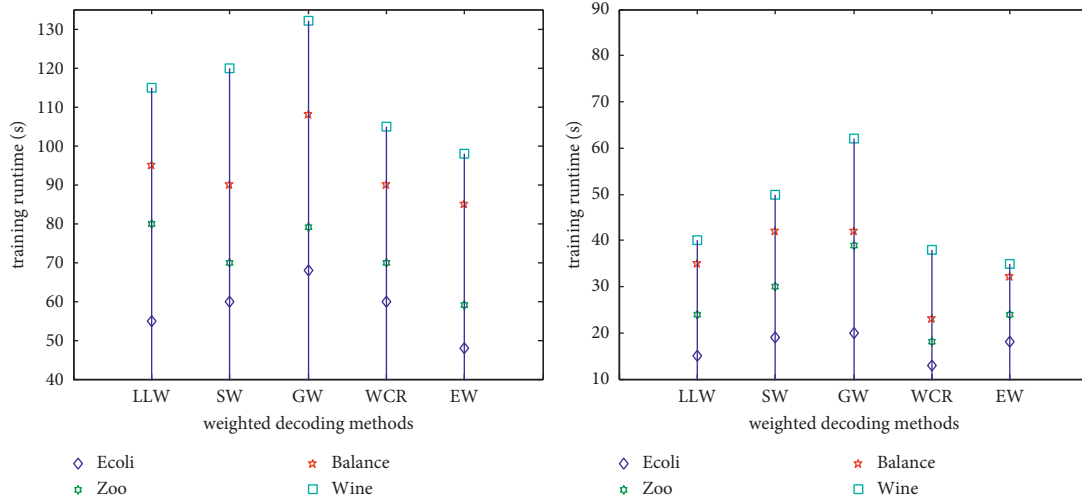

FIGURE 5: The training and testing runtime of different weighted decoding methods.

TABLE 4: Classification accuracy of DRCW-OVO and WCR.

Dataset	SVM		Treec	
	DRCW	WCR	DRCW	WCR
Ecoli	66.67 ± 2.93	**69.01 ± 2.73**	**72.91 ± 0.86**	69.64 ± 1.15
Glass	**97.26 ± 3.73**	93.47 ± 3.72	90.54 ± 7.08	**92.98 ± 2.93**
Satimage	85.84 ± 0.54	**86.14 ± 3.83**	83.93 ± 1.47	**84.01 ± 0.79**
Segment	93.25 ± 0.90	**94.87 ± 1.14**	93.94 ± 1.71	**94.68 ± 1.35**
Vehicle	79.31 ± 3.38	**79.54 ± 2.87**	65.97 ± 2.62	**67.50 ± 1.84**
Vowel	**98.69 ± 0.84**	85.66 ± 3.27	**82.73 ± 2.92**	82.02 ± 5.41
Yeast	57.34 ± 2.32	**58.48 ± 2.60**	51.15 ± 3.88	**51.87 ± 3.02**
Zoo	81.02 ± 4.12	**83.26 ± 3.89**	71.23 ± 13.91	**74.58 ± 3.26**

5. Conclusion

How to estimate the competence reliability of base classifiers is a new issue for ternary ECOC classification. In order to reduce the error caused by the noncompetent classifiers, a new WCR weighted decoding is proposed in this paper. To achieve the goal, the SVDD hyperspheres are used to measure the distance from an instance to be classified to each class. Then the distance is applied as the weight to fuse in the decoding step to make the final decision. In so doing, the

classifier competence reliability is evaluated quantitatively by the weights. In the meanwhile, the influence of the competent classifiers on classification is reinforced and that of the noncompetent ones is weakened. The experimental results prove the promotion of the WCR weighted decoding for classification. Roughly speaking, the proposed algorithm provides new possibilities for the noncompetence contradiction solutions. How to deal with the outlets for subclass separation and data uncertainty [36] is the next research direction.

Data Availability

All data used in this paper can be obtained by contacting the authors of this study.

Ethical Approval

This article does not contain any studies with human or animal subjects performed by any of the authors.

Consent

Informed consent was obtained from all individual participants included in the study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] T. G. Dieterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Artificial Intelligence Research*, vol. 2, no. 2, pp. 263–286, 1995.

[2] S. Escalera, O. Pujol, and P. Radeva, "On the decoding process in ternary error-correcting output codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 120–134, 2010.

[3] C. Ma, I. W. Tsang, F. Shen, and C. Liu, "Error correcting input and output hashing," *IEEE Transactions on Cybernetics*, vol. 49, no. 3, pp. 781–791, 2019.

[4] K. Zhao, T. Matsukawa, and E. Suzuki, "Experimental validation for N-ary error correcting output codes for ensemble learning of deep neural networks," *Journal of Intelligent Information Systems*, vol. 52, no. 2, pp. 367–392, 2019.

[5] X. D. Gu, F. Deng, X. Gao, and R. Zhou, "An improved sensor fault diagnosis scheme based on TA-LSSVM and ECOC-SVM," *Journal of Systems Science and Complexity*, vol. 31, no. 3, pp. 372–384, 2018.

[6] K.-H. Liu, Z.-H. Zeng, and V. T. Y. Ng, "A hierarchical ensemble of ECOC for cancer classification based on multiclass microarray data," *Information Sciences*, vol. 349-350, no. 1, pp. 102–118, 2016.

[7] D. Sánchez, M. A. Bautista, and S. Escalera, "HuPBA8k+: Dataset and ECOC-graph-cut based segmentation of human limbs," *Neurocomputing*, vol. 150, pp. 173–188, 2015.

[8] G. Wei, Z. Ping, and Z. Liang, "Research on synthetic aperture radar image target recognition based on AdaBoostECOC," *Journal of Harbin Engineering University*, vol. 31, no. 2, pp. 232–236, 2012.

[9] S. Escalera, D. Masip, E. Puertas, P. Radeva, and O. Pujol, "Online error correcting output codes," *Pattern Recognition Letters*, vol. 32, no. 3, pp. 458–467, 2011.

[10] K. Crammer and Y. Singer, "On the learnability and design of output codes for multi-class problems," *Machine Learning*, vol. 47, no. 2, pp. 896–909, 2000.

[11] O. Pujol, P. Radeva, and J. Vitria, "Discriminate ECOC: A heuristic method for application dependent design of error correcting output codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 1001–1007, 2006.

[12] 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 Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 1041–1054, 2008.

[13] L. Lei, W. Xiao-Dan, L. Xi, and S. Ya-Fei, "Hierarchical error-correcting output codes based on SVDD," *Pattern Analysis & Applications*, vol. 19, no. 1, pp. 163–171, 2016.

[14] J. Zhou, Y. Yang, M. Zhang, and H. Xing, "Constructing ECOC based on confusion matrix for multiclass learning problems," *Science China Information Sciences*, vol. 59, no. 1, pp. 1–14, 2016.

[15] A. B. Miguel, S. Escalera, and B. Xavier, "On the design of an ECOC-compliant genetic algorithm," *Pattern Recognition*, vol. 47, no. 8, pp. 865–884, 2014.

[16] L. Marie, S. L. Hegarat-Mascle, and E. Aldea, "Evidential framework for error correcting output codes classification," *Engineering Applications of Artificial Intelligence*, vol. 73, no. 5, pp. 10–21, 2018.

[17] Y. Song, X. Wang, L. Lei, and A. Xue, "A novel similarity measure on intuitionistic fuzzy sets with its applications," *Applied Intelligence*, vol. 42, no. 2, pp. 252–261, 2015.

[18] X. Wang and Y. Song, "Uncertainty measure in evidence theory with its applications," *Applied Intelligence*, vol. 48, no. 7, pp. 1672–1688, 2018.

[19] Y. Song, Q. Fu, Y.-F. Wang, and X. Wang, "Divergence-based cross entropy and uncertainty measures of Atanassov's intuitionistic fuzzy sets with their application in decision making," *Applied Soft Computing*, vol. 84, Article ID 105703, 2019.

[20] Y. Song, X. Wang, W. Quan, and W. Huang, "A new approach to construct similarity measure for intuitionistic fuzzy sets," *Soft Computing*, vol. 23, no. 6, pp. 1985–1998, 2019.

[21] Y. Song, J. Zhu, L. Lei, and X. Wang, "Self-adaptive combination method for temporal evidence based on negotiation strategy," *Science China Information Sciences*, vol. 63, no. 11, pp. 56–68, 2020.

[22] J. D. Zhou, X. D. Wang, W. Quan, and X. Yan, "Application of weighted decoding for the consistent-diverse balance problem of error correcting output codes," *Acta Electronica Ginica*, vol. 39, no. 7, pp. 1514–1522, 2011.

[23] L. Lei, Y. Song, and X. Luo, "A new Re-encoding ECOC using a reject option," *Applied Intelligence*, vol. 50, pp. 3090–3100, 2020.

[24] J. D. Zhou, X. D. Wang, H. J. Zhou, J. M. Zhang, and N. Jia, "Decoding design based on posterior probabilites in ternary error-correcting output codes," *Pattern Recognition*, vol. 45, no. 6, pp. 1802–1818, 2012.

[25] P. Simeon, C. Marrocco, and F. Tortorella, "Design of reject rules for ECOC classification systems," *Pattern Recognition*, vol. 45, no. 2, pp. 863–875, 2012.

[26] R. S. Smith and T. Windeat, "Decoding rules for error correcting output code ensembles multiple classifier systems," in *Proceedings of the 6th Intenational Workshop on Multiple Classifier Systems*, pp. 53–63, Springer, Seaside, CA, USA, January 2005.

[27] M. Prior and T. Windeat, "Over-fitting in ensembles of neural network classifiers within ECOC frameworks," *Lecture Notes in Computer Science*, vol. 3541, no. 4, pp. 286–295, 2005.

[28] M. Galar, A. Fernandez, and E. Barrenechea, "Dynamic classifier selection for one-vs-one strategy: avoiding non-competent classifiers," *Pattern Recognition*, vol. 46, no. 12, pp. 3412–3424, 2013.

[29] M. Galar, A. Fernandez, and E. Barrenechea, "DRCW-O VO: distance-based relative competence weighting combination for one-vs-one strategy in multi-class problems," *Pattern Recognition*, vol. 48, no. 1, pp. 28–42, 2015.

[30] N. Hatami, "Thinned-ECOC ensemble based on sequential code shrinking," *Expert Systems with Applications*, vol. 39, no. 1, pp. 936–947, 2012.

[31] E. Hüllermeier and S. Vanderlooy, "Combining predictions in pairwise classification: an optimal adaptive voting strategy and its relation to weighted voting," *Pattern Recognition*, vol. 43, no. 1, pp. 128–142, 2010.

[32] D. Tax and P. W. Robert, "Support vector domain description," *Pattern Recognition Letters*, vol. 20, no. 11-13, pp. 1191–1199, 1999.

[33] D. Tax and P. W. Robert, "Support vector data description," *Machine Learning*, vol. 54, no. 1, pp. 45–66, 2004.

[34] S. Özögür-Akyüz, T. Windeatt, and R. Smith, "Pruning of error correcting output codes by optimization of accuracy-diversity trade-off," *Machine Learning*, vol. 101, no. 1-3, pp. 253–269, 2015.

[35] S. Garcia and F. Herrera, "An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons," *Journal of Machine Learning Research*, vol. 12, no. 9, pp. 2677–2694, 2008.

[36] D. Pellicanó, I. Palamara, M. Cacciola, S. Calcagno, M. Versaci, and F. C. Morabito, "Fuzzy similarity measures for detection and classification of defects in CFRP," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 60, no. 9, pp. 1917–1927, 2013.