Simultaneous meta-data and meta-classifier selection in multiple classifier system.

NGUYEN, T.T., LUONG, A.V., NGUYEN, T.M.V., HA, T.S., LIEW, A.W.-C., MCCALL, J.

2019

Simultaneous Meta-Data and Meta-Classifier Selection in Multiple Classifier System

Tien Thanh Nguyen
School of Computing Science and Digital Media
Robert Gordon University, Aberdeen
United Kingdom
t.nguyen11@rgu.ac.uk

Anh Vu Luong
School of Information and Communication Technology
Griffith University, Gold Coast
Australia
vu.luong@griffithuni.edu.au

Thi Minh Van Nguyen
Department of Planning and Investment
Ba Ria Vung Tau, Vietnam
nguyenvanvt@yahoo.com

Trong Sy Ha
School of Applied Mathematics and Informatics
Hanoi University of Science and Technology, Hanoi, Vietnam
trongsyctk@gmail.com

Alan Wee-Chung Liew
School of Information and Communication Technology
Griffith University, Gold Coast
Australia
a.liew@griffith.edu.au

John McCall
School of Computing Science and Digital Media
Robert Gordon University, Aberdeen
United Kingdom
j.mccall@rgu.ac.uk

ABSTRACT

In ensemble systems, the predictions of base classifiers are aggregated by a combining algorithm (meta-classifier) to achieve better classification accuracy than using a single classifier. Experiments show that the performance of ensembles significantly depends on the choice of meta-classifier. Normally, the classifier selection method applied to an ensemble usually removes all the predictions of a classifier if this classifier is not selected in the final ensemble. Here we present an idea to only remove a subset of each classifier's prediction thereby introducing a simultaneous meta-data and meta-classifier selection method for ensemble systems. Our approach uses Cross Validation on the training set to generate meta-data as the predictions of base classifiers. We then use Ant Colony Optimization to search for the optimal subset of meta-data and meta-classifier for the data. By considering each column of meta-data, we construct the configuration including a subset of these columns and a meta-classifier. Specifically, the columns are selected according to their corresponding pheromones, and the meta-classifier is chosen at random. The classification accuracy of each configuration is computed based on Cross Validation on meta-data. Experiments on UCI datasets show the advantage of proposed method compared to several classifier and feature selection methods for ensemble systems.

CCS CONCEPTS

• **Mathematics of computing** → *Evolutionary algorithms*; • **Computing methodologies** → *Ensemble methods*;

KEYWORDS

Ensemble method, multiple classifiers, classifier fusion, combining classifiers, ensemble selection, classifier selection, feature selection, Ant Colony Optimization

1 INTRODUCTION

Learning from the Ensemble of Classifiers (EoC) to achieve higher classification accuracy than using a single classifier is one of the most popular topics in machine learning research. As each classifier is a hypothesis about the relationship between the features of an observation and its class label, by combining several base classifiers in an ensemble, we can obtain better approximation for this relationship, thereby enhance the performance of the classification system [17, 19].

There are three phases to be considered in an ensemble design, namely generation, selection, and integration. In the generation phase, the learning algorithm(s) learn on the training set(s) to obtain the base classifiers. In the selection phase, a single classifier or a subset of the best classifiers is selected to classify the test sample. In the last phase, the decisions made by the selected EoC are combined to obtain the final prediction [3, 7]. Homogeneous ensemble methods like Bagging [2], and Random Subspace [1] focus on the generation phase in which they concentrate on the generation of new training schemes from the original training set. Meanwhile, heterogeneous ensemble methods focus on combining algorithms which operate on the outputs of the base classifiers (called meta-data or Level1 data) [17–20]. The combining algorithm is also called a meta-classifier.

In this study, we focus on the selection phase in the ensemble design by proposing a simultaneous meta-data and meta-classifier selection method for heterogeneous ensemble systems. In an ensemble system, there usually exists a subset of EoC that makes the ensemble perform better than using the entire set of base classifiers. However, as pruning a classifier's output is more general then pruning the classifier itself, we are motivated to expect meta-data selection to result in better performance. Moreover, in a heterogeneous ensemble system, the performance of the ensemble system depends on the performance of the meta-classifier [26]. Therefore, by doing meta-data and meta-classifier selection simultaneously, we could obtain a higher performance ensemble than other existing selection-based ensemble systems.

This paper introduces a method to simultaneously select the subset of meta-data and the meta-classifier to obtain a high-performance ensemble system. We first generate the base classifiers and then the meta-data from the training observations. Having the meta-data and the given set of meta-classifiers, we then simultaneously search for the optimal subset of meta-data and the meta-classifier for the ensemble. To solve the optimization problem, we use Ant Colony Optimization (ACO) [21]. Although we only use the original Ant Colony Optimization, which is simple and not generally considered state-of-the-art, our model still shows better performance in comparision with other algorithms on a wide range of datasets. The contribution of our paper is in the following:

- We propose to simultaneously select the subset of meta-data and the associated meta-classifier for a heterogeneous ensemble system.
- We propose to use ACO to search for the optimal solution.
- Experiments on the 40 UCI datasets show that our approach is better than the selected benchmark algorithms.

The paper is organized as follows. In section 2, we briefly review heterogeneous ensemble systems and existing ensemble selection approaches. Our proposed method is introduced in section 3 including the model formulation and the algorithm. The experimental studies including the datasets used, the experimental settings, and the results and discussion are introduced in section 4. Finally, the conclusion and suggestions for future development are given in section 5.

2 RELATED WORK

2.1 Heterogeneous ensemble systems

In a heterogeneous ensemble system, we apply several different learning algorithms on a given training dataset to generate a set of base classifiers. The outputs of the base classifiers, called the meta-data, are then combined to obtain the final decision model. Let $\mathcal{D} = \{\mathbf{x}_i, \hat{y}_i\}, i = 1, ..., N$ be the training set of N observations, $\mathcal{Y} = \{y_m\}$ be the set of M class labels, and $\mathcal{K}$ be the set of K learning algorithms. The meta-data of the training set is given by the $N \times KM$ matrix $\mathbf{L}$:

$$\mathbf{L} = \begin{bmatrix} P_1(y_1|\mathbf{x}_1)\ \dots\ P_1(y_M|\mathbf{x}_1) \dots P_K(y_1|\mathbf{x}_1)\ \dots\ P_K(y_M|\mathbf{x}_1) \\ \vdots \qquad\quad \vdots \qquad\quad \vdots \\ \underbrace{P_1(y_1|\mathbf{x}_N) \dots P_1(y_M|\mathbf{x}_N)}_{\text{predictions of } 1^{st} \text{ classifier}} \dots \underbrace{P_K(y_1|\mathbf{x}_N) \dots P_K(y_M|\mathbf{x}_N)}_{\text{predictions of } K^{th} \text{ classifier}} \end{bmatrix} \quad (1)$$

in which $P_k(y_m|\mathbf{x}_n)$ is the prediction (posterior probability) of the k^{th} base classifier that observation $\mathbf{x}_n$ belongs to the class y_m. Each row of the matrix corresponds to an observation, and it is obtained by concatenating all the predictions of K base classifiers.

Table 1: Datasets used in the experimental studies

Datasets	# of observations	# of classes	# of dimension
Abalone	4174	3	8
Appendicitis	106	2	7
Artificial	700	2	10
Australian	690	2	14
Balance	625	3	4
Banana	5300	2	2
Biodeg	1055	2	41
Blood	748	2	4
Breast-cancer	683	2	9
Bupa	345	2	6
Cleveland	297	5	13
Contraceptive	1473	3	9
Fertility	100	2	9
Haberman	306	2	3
Heart	270	2	13
Hepatitis	80	2	19
Hill-valley	2424	2	100
Led7digit	500	10	7
Madelon	2000	2	500
Magic	19020	2	10
Mammographic	830	2	5
Musk1	476	2	166
Musk2	6598	2	166
Newthyroid	215	3	5
Page-blocks	5472	5	10
Phoneme	5404	2	5
Pima	768	2	8
Ring	7400	2	20
Sonar	208	2	60
Spambase	4601	2	57
Tae	151	3	5
Tic-tac-toe	958	2	9
Titanic	2201	2	3
Vehicle	846	4	18
Vertebral	310	3	6
Waveform-w-noise	5000	3	40
Waveform-wo-noise	5000	3	21
Wdbc	569	2	30
Wine-red	1599	6	11
Wine-white	4898	7	11

There are two types of meta-classifiers introduced for the heterogeneous ensemble systems: fixed combining methods and trainable combining methods. Fixed combining methods predict the label based on only the meta-data of the test sample. Kitler et al. [10] introduced six fixed combining rules (Sum, Product, Majority Vote, Min, Max, and Median) for an ensemble and pointed out that the Sum rule is the most reliable combining method for the prediction. Trainable combining methods, on the other hand, exploit the label information in the meta-data of the training set when constructing the meta-classifier. By doing this, trainable combining methods usually perform better than fixed combining methods.

In trainable combining methods, we can divide the combining methods into two categories, namely weight-based combining methods and representation-based combining methods. In the first category, the meta-classifier is formed based on the M weighted linear combinations of posterior probabilities for the M classes. The weights can be computed using, for example, the Multi-Response Linear Regression (MLR) method [25], or the MLR plus hinge loss function [22]. On the other hand, the representation-based combining methods generate a representation of the meta-data for each class label. The class label is assigned to a test sample based on the similarity between the set of representations and the meta-data of the test sample. Some examples of methods in this category are Decision Template [12], Bayesian-based combining method [19], Granular-based prototype (interval-based representation) [18], and Fuzzy IF-THEN Rule-combining method [17].

2.2 Selection Methods in Ensemble System

In this section, we briefly introduce several selection methods applied to ensemble system in which not only the base classifiers but also the features are selected to optimize the ensemble's performance. We start with the ensemble selection (ES) methods (known by two different names: selective ensemble and ensemble pruning which are methods that search for a subset of classifiers that performs better than the whole ensemble. In ES, a single classifier or an ensemble of classifiers can be obtained using a static or a dynamic approach. The static approach selects a subset of base classifiers during the training phase and uses the same subset of base classifiers to predict all unseen samples. Nguyen et al. [15] proposed a novel encoding method that encodes both the base classifiers and six fixed combining rules in a binary vector and used a Genetic Algorithm to search for the optimal EoC and the optimal fixed combining rule. Shunmugapriya and Kanmani [23] used the Artificial Bee Colony (ABC) algorithm to find the optimal set of base classifiers and the meta-classifier. Chen et al. [5] used the Ant Colony Optimization (ACO) algorithm to find the optimal set of base classifiers in the ensemble system with the Decision Tree as the meta-classifier. Zhang et al. [27] formulated the ES problem as a quadratic integer programming problem and used semi-definite programming to obtain an approximate solution.

Meanwhile, the dynamic approach selects a classifier by dynamic classifier selection (DCS) or an EoC by dynamic ensemble selection (DES) with the most competences in a defined region associated with each test sample. Some examples of DCS and DES methods are MLA [24], KNOP [4], KNORA Union and KNORA Eliminate [11], and Random Projection-based DES [7]. Comparison experiments in [3] indicated that simple dynamic selection methods like KNORA Union can sometimes perform better than the complex ones. A detailed review of methods for DCS and DES can be found in [3, 6].

Finally, we introduce some feature selection methods that were developed for ensemble systems. Kuncheva et al. [12] used a Venn diagram to encode the input features used by the learning algorithms and then search for the optimal set of input features and learning algorithms using GA. Nguyen et al. [14] developed a GA-based method to simultaneously learn the optimal EoC as well as the associated input features for the learning algorithm. The method introduced in [16] uses GA to find the optimal set of meta-data's columns from the matrix **L** for the Decision Tree meta-classifier.

3 PROPOSED METHOD

In this study, we introduce a method based on ACO to simultaneously select a subset of meta-data's columns from **L** as well as the meta-classifier for a heterogeneous ensemble system. The columns of the meta-data are selected as paths by the ants. Each ant is also assigned a certain meta-classifier. The subset of meta-data's columns and the associated meta-classifier form a configuration of an ant. In each iteration, an ant tries to select a path in its route to obtain a better configuration. At the end of ACO, we select the best configuration based on an evaluation criterion. This optimal solution will be used to classify the test samples.

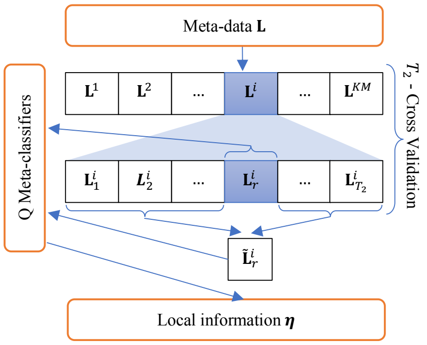

Figure 1: Module to compute the local information

Some notations that will be used in our algorithm are given below:

- $\mathbf{L}^i$: the meta-data associated with the column i of $\mathbf{L}$
- $\mathbf{L}_j^i$: the j^{th} part of $\mathbf{L}^i$
- $\mathbf{L}^{\{\boldsymbol{u}\}}$: the meta-data associated with the set of columns $\boldsymbol{u} = \{u_1, u_2, ..., u_k\}$ of $\mathbf{L}$
- $\mathbf{L}_j^{\{\boldsymbol{u}\}}$: the j^{th} part of $\mathbf{L}^{\{\boldsymbol{u}\}}$
- na: the number of artificial ants in the colony
- τ_i: the pheromone associated with the i^{th} column of $\mathbf{L}$
- $\eta_{i,q}$: the local information used to estimate the contribution of the i^{th} column of $\mathbf{L}$ to the q^{th} meta-classifier
- S_j: the configuration constructed by the j^{th} ant.
- α_{S_j}: the classification accuracy of configuration S_j
- ρ: the evaporation rate, $\rho \in [0, 1]$
- $maxT$: the maximum iteration number

In this framework, K learning algorithms, Q meta-classifiers, and the training set $\mathcal{D}$ are given. We start with the meta-data generated from the training set $\mathcal{D}$ using the T_1-fold cross validation procedure. Specifically, the training set $\mathcal{D}$ is partitioned to obtain T_1 disjoint parts $\mathcal{D} = \mathcal{D}_1 \cup ... \cup \mathcal{D}_{T_1}$, $\mathcal{D}_l \cap \mathcal{D}_r = \emptyset$ $(l \neq r)$, and $|\mathcal{D}_1| \approx ... \approx |\mathcal{D}_{T_1}|$. The meta-data of observations in $\mathcal{D}_r$ is then formed by the classifiers generated by learning the K algorithms on $\widetilde{\mathcal{D}}_r = \mathcal{D} - \mathcal{D}_r$. The meta-data of all training observations belonging to $\mathcal{D}$ is finally obtained by concatenating all meta-data from each $\mathcal{D}_r$ into the form of matrix $\mathbf{L}$ given by (1).

We then calculate the local information of each column of $\mathbf{L}$ and the meta-classifier. This is the guide for an ant to search in the local area to find the new path. To calculate $\eta_{i,q}$, a T_2-fold cross validation procedure is applied to the column $\mathbf{L}^i$ of the meta-data. We first obtain T_2 disjoint parts $\mathbf{L}^i = \mathbf{L}_1^i \cup ... \cup \mathbf{L}_{T_2}^i$, $\mathbf{L}_l^i \cap \mathbf{L}_r^i = \emptyset$ $(l \neq r)$, and $|\mathbf{L}_1^i| \approx ... \approx |\mathbf{L}_{T_2}^i|$. Predictions of observations in $\mathbf{L}_r^i$ is then formed by the q^{th} meta-classifier trained on $\widetilde{\mathbf{L}}_r^i = \mathbf{L}^i - \mathbf{L}_r^i$. Predictions of all training data belonging to $\mathbf{L}^i$ is finally obtained by gathering all predictions from each $\mathbf{L}_r^i$. The average accuracy of the q^{th} meta-classifier over all observations in $\mathbf{L}^i$ is used as the local information $\eta_{i,q}$ (see Figs 1). We also initialize the pheromone τ_i of each column $\mathbf{L}^i$ with a small positive number for the probability selection process.

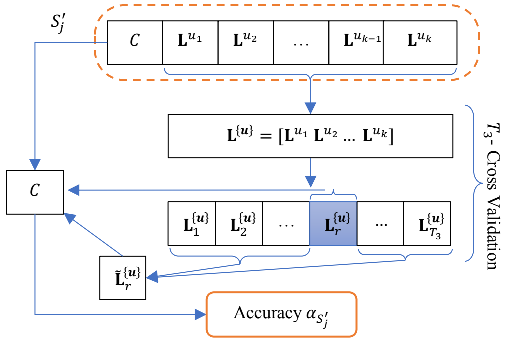

Figure 2: Module to compute the evaluation criterion of a configuration

In the first step of each iteration in the ACO algorithm, each ant is randomly given a meta-classifier (uniform distribution). In the following steps, when the j^{th} ant begins its configuration search, it tries to select a column $\mathbf{L}^i$ from $\mathbf{L}$ which does not exist in its current configuration S_j using roulette wheel selection. The probability p_i of the column $\mathbf{L}^i$ to be selected by the j^{th} ant with the associated q^{th} meta-classifier is computed based on the pheromone of each column and the local information as:

$$p_i = \begin{cases} \frac{\tau_i \times \eta_{i,q}}{\sum_{t=1,\mathbf{L}^t \notin S_j}^{m} \tau_t \times \eta_{t,q}} & \text{if } \mathbf{L}^i \notin S_j \\ 0 & otherwise \end{cases} \tag{2}$$

We define the evaluation criterion for configuration S_j as α_{S_j}. During the ACO main process, suppose that the current configuration of an ant is $S_j = \{C, \mathbf{L}^{u_1}, \mathbf{L}^{u_2}, ..., \mathbf{L}^{u_{k-1}}\}$ and a column $\mathbf{L}^i$ is selected, a new configuration $S'_j = \{C, \mathbf{L}^{u_1}, \mathbf{L}^{u_2}, ..., \mathbf{L}^{u_{k-1}}, \mathbf{L}^{u_k} = \mathbf{L}^i\} = \{C, \mathbf{L}^{\{\boldsymbol{u}\}}\}$ of this ant is generated. Then S'_j is tested by using T_3-fold cross validation on the corresponding subset of the meta-data in its configuration to calculate $\alpha_{S'_j}$. Specifically, $\mathbf{L}^{\{\boldsymbol{u}\}}$ is partitioned into T_3 disjoint parts $\mathbf{L}^{\{\boldsymbol{u}\}} = \mathbf{L}_1^{\{\boldsymbol{u}\}} \cup ... \cup \mathbf{L}_{T_3}^{\{\boldsymbol{u}\}}$, $\mathbf{L}_l^{\{\boldsymbol{u}\}} \cap \mathbf{L}_r^{\{\boldsymbol{u}\}} = \emptyset\ (l \neq r)$, and $|\mathbf{L}_1^{\{\boldsymbol{u}\}}| \approx ... \approx |\mathbf{L}_{T_3}^{\{\boldsymbol{u}\}}|$. Predictions of observations in $\mathbf{L}_r^{\{\boldsymbol{u}\}}$ is then formed by the meta-classifier C trained on $\widetilde{\mathbf{L}}_r^{\{\boldsymbol{u}\}} = \mathbf{L}^{\{\boldsymbol{u}\}} - \mathbf{L}_r^{\{\boldsymbol{u}\}}$. Based on the predictions of observations in $\mathbf{L}_r^{\{\boldsymbol{u}\}}$, we compute the loss function $\mathcal{L}_{0-1}\{\mathbf{L}_r^{\{\boldsymbol{u}\}}, S'_j\}$ by (3). The evaluation criterion $\alpha_{S'_j}$ of configuration S'_j is computed as the average classification accuracy for all $\mathbf{L}_r^{\{\boldsymbol{u}\}}$ $r = 1, ..., T_3$ (Fig. 2)

$$\mathcal{L}_{0-1}\{\mathbf{L}_r^{\{\boldsymbol{u}\}}, S'_j\} = \frac{1}{|\mathbf{L}_r^{\{\boldsymbol{u}\}}|} \sum_{\mathbf{x} \in \mathbf{L}_r^{\{u\}}} \mathbb{I}[y_{\mathbf{x}} \neq \text{predict}(S'_j, \mathbf{x})] \tag{3}$$

$$\mathcal{L}_{0-1}(S'_j) = \left\{ \frac{1}{T_3} \sum_{r=1}^{T_3} \mathcal{L}_{0-1}\{\mathbf{L}_r^{\{\boldsymbol{u}\}}, S'_j\} \right\} \tag{4}$$

$$\alpha_{S'_j} = 1 - \mathcal{L}_{0-1}(S'_j) \tag{5}$$

where predict$(S'_j, \mathbf{x})$ returns the predicted class label for observation $\mathbf{x}$ by using the configuration S'_j and $y_{\mathbf{x}}$ is the true label of $\mathbf{x}$. If the performance of S'_j is better than S_j, it will replace S_j and the ant continues to find another column using the same strategy to generate a new configuration. If S'_j cannot improve the accuracy of S_j, this ant keeps its current configuration and stops its search in the iteration. During the ants' searching process, once a column $\mathbf{L}^i$ is chosen to be added to any S_j to form a better configuration S'_j, the pheromone of $\mathbf{L}^i$ will accumulate, thus enhancing the probability of this column being selected by other ants. The improvement of accuracy from S_j to S'_j is used to update the pheromone of $\mathbf{L}^i$. The update rule for the improvement is given in (6).

$$\tau_i^{(new)} = \tau_i^{(old)} + CC \times \tau_i^{(old)} \times \frac{\alpha_{S'_j} - \alpha_{S_j}}{\alpha_{S_j}} \tag{6}$$

where CC refers to a constant number. The pheromones of all candidates will evaporate after each iteration. The evaporation rule is given in (7).

$$\tau_i^{(new)} \leftarrow \tau_i^{(old)} \times (1 - \rho) \tag{7}$$

Therefore, the pheromone of the strong candidates will accumulate and the pheromone of the poor ones will vanish by evaporation. The evaporation rate ρ and CC are introduced to adjust the emphasis of historical knowledge and the current knowledge. The greater ρ is, the less historical information will be used. The greater CC is, the more important current knowledge is considered.

When we finish looping through all iterations, the best configuration S_{best} among all na ants will be chosen as the final configuration. To speed up the training process, we mark and save all the configurations which have already been explored by artificial ants in the past. So that we do not need to recalculate the evaluation criterion for the visited configurations. Once we obtain the final configuration for our ensemble using cross-validation, all the base classifiers are trained on the entire training set $\mathcal{D}$ for the predictions on the testing set later. The pseudo code of the training process of the proposed method is presented in Algorithm 1, 2 and 3 in the Supplement Material.

The testing process uses the base classifiers and the best configuration S_{best}. For each unlabeled sample $\mathbf{x}^{test}$, we first obtain its meta-data $\mathbf{L}(\mathbf{x}^{test})$ by using the base classifers. Based on the best configuration S_{best}, we get the corresponding subset of columns $\mathbf{L}^{\boldsymbol{I}}(\mathbf{x}^{test})$ and the meta-classifier C associated with these columns. By applying C to $\mathbf{L}^{\boldsymbol{I}}(\mathbf{x}^{test})$, we get the prediction for the class label of $\mathbf{x}^{test}$. The pseudo code of the testing process is presented in Algorithm 4 in the Supplement Material.

4 EXPERIMENTAL STUDIES

4.1 Datasets and Experimental Settings

We conducted experiments on 40 datasets to evaluate the performance of the proposed method. The datasets are selected to be diverse in the number of class labels, the number of observations, and the number of dimensions. Information about the datasets used in the experiment is given in Table 1.

We compared the proposed method to some classifier selection and feature selection methods developed for the heterogeneous ensemble systems. The benchmark algorithms we selected including: ACO-S1 [5], GA Meta-data [16], KNORA Union and KNORA Eliminate [11]. In these benchmark algorithms, we used the same learning algorithms as in the proposed method. For ACO-S1, the Decision Tree works as the meta-classifier like in the original paper.

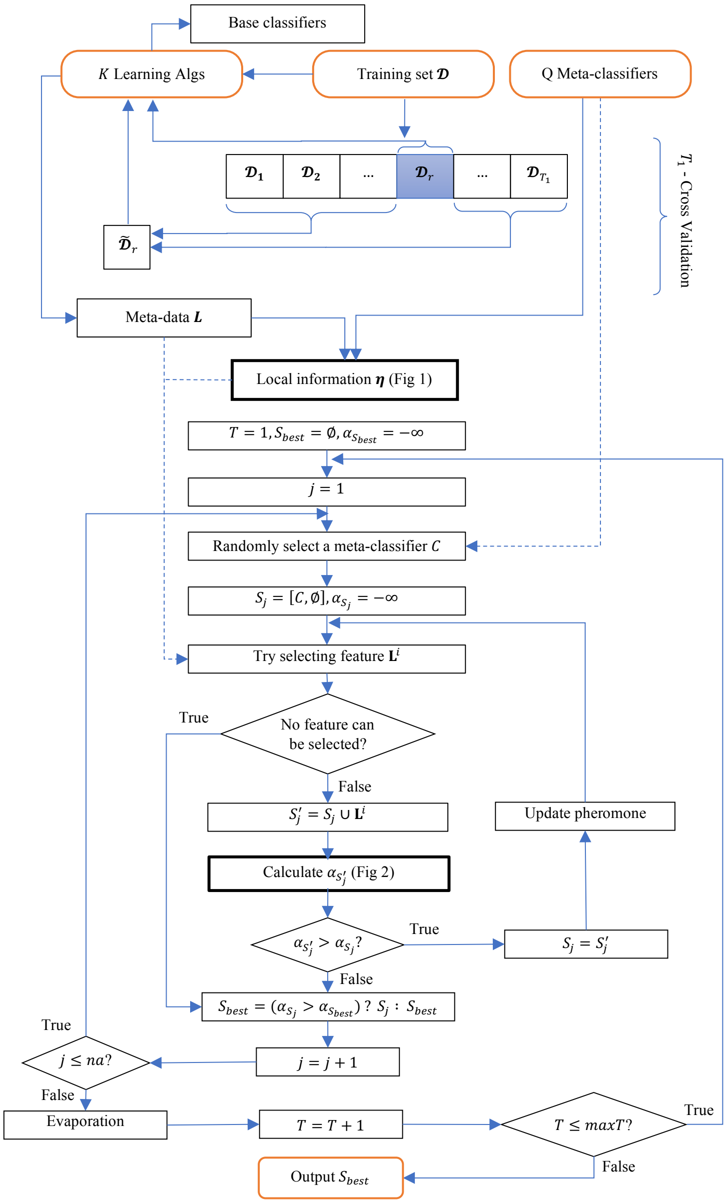
Base classifiers
K Learning Algs
Training set $\mathcal{D}$
Q Meta-classifiers
$\mathcal{D}_1$
$\mathcal{D}_2$
...
$\mathcal{D}_r$
...
$\mathcal{D}_{T_1}$
T_1 - Cross Validation
$\tilde{\mathcal{D}}_r$
Meta-data $\boldsymbol{L}$
Local information $\boldsymbol{\eta}$ (Fig 1)
$T = 1, S_{best} = \emptyset, \alpha_{S_{best}} = -\infty$
$j = 1$
Randomly select a meta-classifier C
$S_j = [C, \emptyset], \alpha_{S_j} = -\infty$
Try selecting feature $\mathbf{L}^i$
True
No feature can be selected?
False
$S_j' = S_j \cup \mathbf{L}^i$
Update pheromone
Calculate $\alpha_{S_j'}$ (Fig 2)
True
$\alpha_{S_j'} > \alpha_{S_j}$?
$S_j = S_j'$
False
$S_{best} = (\alpha_{S_j} > \alpha_{S_{best}})\ ?\ S_j : S_{best}$
True
$j \leq na$?
$j = j + 1$
False
Evaporation
$T = T + 1$
$T \leq maxT$?
True
Output S_{best}
False

Figure 3: Training process of the proposed method

The other parameters were set similar to the original paper. For GA, the number of generations and the number of individuals in each generation was set to 100 and 50, respectively. For KNORA Union and KNORA Eliminate, the number of nearest neighbors was set to 7 as it is the best value for the DES method [6]. For the ACO algorithm to search for the optimal solution in the proposed method, we set $maxT = 100, na = 50, \rho = 0.1$, and $CC = 1$. For the cross validation procedures to generate the meta-data of the training set, to calculate the local information, and the evaluation criteria, we set $T_1 = 10, T_2 = T_3 = 2$.

In this study, we performed 10-fold cross validation and ran the test 3 times to obtain 30 test results of each method on each dataset. Based on the experimental results, we used the Wilcoxon signed rank test [8] to compare the classification results of the proposed method and each benchmark algorithm on each dataset. The null hypothesis is "there is no statistically significant difference in the results produced by the two methods". The null hypothesis is rejected if the p-value of the test is smaller than a given significance level, which we set to 0.05.

4.2 Results and Discussions

We first used 3 learning algorithms, namely Linear Discriminant Analysis (denoted by LDA), Naïve Bayes, and k Nearest Neighbor

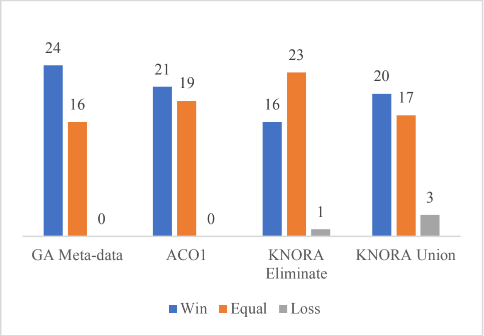

Figure 4: The Wilcoxon statistical test results (using 3 learning algorithms)

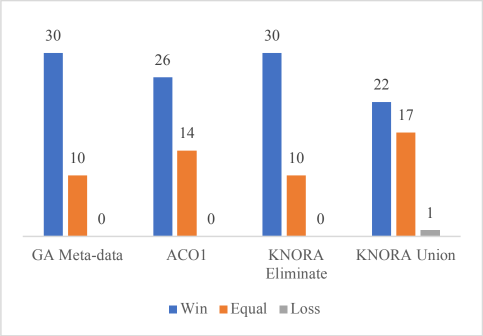

Figure 5: The Wilcoxon statistical test results (using 7 learning algorithms)

Table 2: The classification error of the proposed method and the benchmark algorithms (using 3 learning algorithms)

	GA Meta-data		**ACO-S1**		**KNORA Eliminate**		**KNORA Union**		**Proposed Method**	
	Mean	***Variance***	***Mean***	***Variance***	***Mean***	***Variance***	***Mean***	***Variance***	***Mean***	***Variance***
Abalone	0.4736●	5.43E-04	0.4720●	7.98E-04	0.4678	4.90E-04	0.4707●	3.88E-04	0.4576	3.99E-04
Appendicitis	0.1600	1.04E-02	0.1827	1.51E-02	0.1291	1.14E-02	0.1133□	8.38E-03	0.1415	9.27E-03
Artificial	0.2295	2.39E-03	0.2257	2.41E-03	0.2257	1.98E-03	0.2171	1.23E-03	0.2310	1.57E-03
Australian	0.1807●	1.30E-03	0.1816●	2.38E-03	0.1589●	1.52E-03	0.1357	1.06E-03	0.1333	1.01E-03
Balance	0.0852	1.10E-03	0.0960	8.14E-04	0.1184●	3.27E-04	0.1093●	3.92E-04	0.0912	8.87E-04
Banana	0.1116	1.23E-04	0.1129	2.32E-04	0.1157	1.62E-04	0.1079□	1.51E-04	0.1131	1.12E-04
Biodeg	0.1836●	1.22E-03	0.1800●	1.21E-03	0.1479	6.67E-04	0.1479	7.16E-04	0.1393	9.21E-04
Blood	0.2344	6.75E-04	0.2820●	2.86E-03	0.2286	1.32E-03	0.2205	8.87E-04	0.2348	2.32E-03
Breast-cancer	0.0420●	8.14E-04	0.0405	6.97E-04	0.0410	7.67E-04	0.0444●	6.95E-04	0.0356	6.97E-04
Bupa	0.3804●	8.55E-03	0.3548	5.43E-03	0.3469●	2.47E-03	0.3373	3.13E-03	0.3197	4.80E-03
Cleveland	0.4433●	4.06E-03	0.4643●	6.10E-03	0.4162	3.06E-03	0.4038	3.37E-03	0.4027	6.28E-03
Contraceptive	0.5237●	1.30E-03	0.5028●	1.95E-03	0.4639	2.01E-03	0.4574	1.16E-03	0.4652	1.44E-03
Fertility	0.1900●	1.29E-02	0.1467	3.16E-03	0.1367	2.99E-03	0.1333	2.89E-03	0.1300	3.43E-03
Haberman	0.2964	2.26E-03	0.2984	1.72E-03	0.2778	1.82E-03	0.2767	1.80E-03	0.2823	2.26E-03
Heart	0.2395●	6.93E-03	0.2185●	8.26E-03	0.1938	5.18E-03	0.1753	3.65E-03	0.1716	3.52E-03
Hepatitis	0.1750	1.42E-02	0.2083	9.72E-03	0.1458	7.38E-03	0.1458	9.46E-03	0.1750	8.96E-03
Hill-valley	0.2745	2.47E-03	0.2785	3.54E-03	0.3228●	1.29E-03	0.4086●	1.60E-03	0.2629	3.48E-03
Led7digit	0.2973●	4.50E-03	0.3013●	6.05E-03	0.2680□	4.82E-03	0.2653□	3.86E-03	0.2860	4.22E-03
Madelon	0.2870	6.94E-04	0.2870	6.94E-04	0.3287●	8.82E-04	0.3787●	1.14E-03	0.2873	8.98E-04
Magic	0.1920●	1.37E-04	0.1902●	4.75E-05	0.1934●	5.56E-05	0.1933●	4.80E-05	0.1887	4.13E-04
Mammographic	0.2032●	1.97E-03	0.2169●	1.76E-03	0.1855	1.90E-03	0.1851	1.58E-03	0.1827	1.17E-03
Musk1	0.1344●	1.61E-03	0.1245●	1.77E-03	0.1708●	2.28E-03	0.1695●	3.87E-03	0.1001	1.30E-03
Musk2	0.0350	3.46E-05	0.0355	3.45E-05	0.0356	4.37E-05	0.0498●	7.44E-05	0.0368	5.94E-05
Newthyroid	0.0371	1.22E-03	0.0418●	1.36E-03	0.0947●	3.74E-03	0.0900●	2.60E-03	0.0576	2.64E-03
Page-blocks	0.0420	4.35E-05	0.0462	7.31E-05	0.0424	5.44E-05	0.0503●	5.14E-05	0.0437	5.54E-05
Phoneme	0.1149	2.11E-04	0.1149	2.11E-04	0.1337●	1.77E-04	0.1796●	3.30E-04	0.1208	1.25E-03
Pima	0.3056●	2.34E-03	0.3078●	2.35E-03	0.2366	2.31E-03	0.2427	2.84E-03	0.2318	2.47E-03
Ring	0.1232●	1.83E-04	0.1211●	1.30E-04	0.2590●	1.37E-04	0.2148●	8.96E-05	0.1162	1.55E-04
Sonar	0.2583●	8.89E-03	0.2368	5.91E-03	0.2375	8.11E-03	0.2437	5.70E-03	0.2162	8.55E-03
Spambase	0.1185●	1.95E-04	0.1224●	2.96E-04	0.1072●	1.23E-04	0.0977	9.32E-05	0.0960	2.15E-04
Tae	0.5453●	1.35E-02	0.5129	1.30E-02	0.4863	1.32E-02	0.4925	1.57E-02	0.4794	1.67E-02
Tic-tac-toe	0.1166	7.12E-04	0.1166	7.12E-04	0.1754●	7.50E-04	0.2220●	9.42E-04	0.1183	6.73E-04
Titanic	0.2160	3.81E-04	0.2178	4.08E-04	0.2282	9.77E-04	0.2260	6.16E-04	0.2425	5.19E-03
Vehicle	0.2627●	1.90E-03	0.2597●	1.44E-03	0.2651●	2.13E-03	0.2569●	1.11E-03	0.2203	1.09E-03
Vertebral	0.1893●	3.38E-03	0.1527	3.46E-03	0.1753	4.21E-03	0.1968●	4.39E-03	0.1581	2.87E-03
Waveform-w-noise	0.1787●	2.04E-04	0.1770●	2.22E-04	0.1647●	2.81E-04	0.1692●	1.79E-04	0.1479	1.71E-04
Waveform-wo-noise	0.1738●	4.45E-04	0.1705●	2.75E-04	0.1569●	2.79E-04	0.1653●	2.93E-04	0.1605	1.48E-02
Wdbc	0.0352	6.19E-04	0.0457●	8.53E-04	0.0475●	9.50E-04	0.0399●	3.09E-04	0.0293	3.97E-04
Wine-red	0.4653●	2.14E-03	0.4690●	1.05E-03	0.4180	8.78E-04	0.4234●	1.19E-03	0.4084	8.95E-04
Wine-white	0.4798●	4.58E-04	0.4947●	6.21E-04	0.4502	4.67E-04	0.4682●	3.06E-04	0.4524	5.60E-04
Average ranking	**3.43**		**3.5**		**3.09**		**2.95**		**2.04**	

● and □ mean the proposed method is better or worse than the benchmark algorithm, respectively.

(k was set to 5, denoted by $k\text{NN}_5$) [17, 19] to construct the heterogeneous ensemble system. The set of meta-classifiers was set similarly to the set of learning algorithms. The experimental results of the proposed method and 4 benchmark algorithms were shown in Table 2.

Clearly, the proposed method is better than the benchmark algorithms on the datasets. Compared to ACO-S1, the proposed method wins on 21 datasets. Our method significantly outperforms GA Meta-data, winning on 24 datasets and does not lose on any datasets.

The proposed method is also better than the two DES method, as our method win KNORA Union and KNORA Eliminate on 16 and 20 datasets, respectively. We also computed the average ranking of all methods based on their experimental results. It once again shows the outstanding performance of the proposed method as our

Table 3: The selected meta-data's columns and meta-classifier for 20 experimental datasets (using 3 classifiers)

Dataset	Selected meta-data's columns	Selected meta-classifier
Abalone	$P_2(y_1 \mid .)$, $P_3(y_2 \mid .)$, $P_3(y_3 \mid .)$	LDA
Appendicitis	$P_2(y_1 \mid .)$, $P_3(y_1 \mid .)$	kNN$_5$
Artificial	$P_3(y_1 \mid .)$, $P_3(y_2 \mid .)$	kNN$_5$
Australian	$P_2(y_2 \mid .)$	Naïve Bayes
Balance	$P_1(y_1 \mid .)$, $P_3(y_3 \mid .)$	kNN$_5$
Biodeg	$P_1(y_2 \mid .)$, $P_2(y_2 \mid .)$, $P_3(y_2 \mid .)$	LDA
Blood	$P_1(y_1 \mid .)$, $P_2(y_1 \mid .)$, $P_2(y_2 \mid .)$, $P_3(y_2 \mid .)$	kNN$_5$
Cleveland	$P_1(y_5 \mid .)$, $P_2(y_2 \mid .)$, $P_2(y_3 \mid .)$, $P_2(y_4 \mid .)$, $P_3(y_1 \mid .)$	Naïve Bayes
Contraceptive	$P_1(y_1 \mid .)$, $P_1(y_2 \mid .)$, $P_1(y_3 \mid .)$, $P_2(y_1 \mid .)$, $P_2(y_3 \mid .)$, $P_3(y_1 \mid .)$, $P_3(y_2 \mid .)$, $P_3(y_3 \mid .)$	Naïve Bayes
Haberman	$P_2(y_2 \mid .)$, $P_3(y_1 \mid .)$	Naïve Bayes
Magic	$P_2(y_1 \mid .)$, $P_3(y_1 \mid .)$	LDA
Mammographic	$P_1(y_2 \mid .)$, $P_2(y_2 \mid .)$, $P_3(y_2 \mid .)$	LDA
Page-blocks	$P_1(y_2 \mid .)$, $P_1(y_5 \mid .)$, $P_2(y_1 \mid .)$, $P_2(y_2 \mid .)$, $P_3(y_1 \mid .)$, $P_3(y_3 \mid .)$, $P_3(y_4 \mid .)$, $P_3(y_5 \mid .)$	kNN$_5$
Pima	$P_1(y_2 \mid .)$, $P_2(y_1 \mid .)$, $P_3(y_2 \mid .)$	LDA
Tae	$P_1(y_1 \mid .)$, $P_1(y_2 \mid .)$, $P_1(y_3 \mid .)$, $P_2(y_1 \mid .)$, $P_2(y_2 \mid .)$	kNN$_5$
Tic-tac-toe	$P_1(y_1 \mid .)$, $P_2(y_2 \mid .)$, $P_3(y_1 \mid .)$	kNN$_5$
Waveform-w-noise	$P_1(y_2 \mid .)$, $P_1(y_3 \mid .)$, $P_2(y_2 \mid .)$, $P_3(y_1 \mid .)$, $P_3(y_2 \mid .)$	LDA
Waveform-wo-noise	$P_1(y_1 \mid .)$, $P_2(y_2 \mid .)$	LDA
Wine-red	$P_1(y_4 \mid .)$, $P_2(y_3 \mid .)$	LDA
Wine-white	$P_1(y_4 \mid .)$, $P_1(y_5 \mid .)$, $P_3(y_3 \mid .)$, $P_3(y_5 \mid .)$	LDA

Table 4: The classification error of the proposed method and the benchmark algorithms (using 7 learning algorithms)

	GA Meta-data		ACO-S1		KNORA Eliminate		KNORA Union		Proposed Method	
	Mean	***Variance***	***Mean***	***Variance***	***Mean***	***Variance***	***Mean***	***Variance***	***Mean***	***Variance***
Abalone	0.4986●	6.84E-04	0.4888●	9.25E-04	0.4812●	6.90E-04	0.4652●	4.42E-04	0.4534	4.10E-04
Appendicitis	0.1767●	9.44E-03	0.1630	1.18E-02	0.1424	8.98E-03	0.1261	9.77E-03	0.1324	1.01E-02
Artificial	0.2667●	3.80E-03	0.2229	2.14E-03	0.2538	1.89E-03	0.2210	1.70E-03	0.2476	4.18E-03
Australian	0.1845●	1.97E-03	0.1908●	2.62E-03	0.1826●	2.30E-03	0.1541	1.63E-03	0.1401	1.32E-03
Balance	0.0491	1.08E-03	0.0581	1.13E-03	0.1157●	1.12E-03	0.1029●	4.99E-04	0.0597	1.00E-03
Banana	0.1331●	3.65E-04	0.1279●	3.51E-04	0.1184●	2.10E-04	0.1029●	8.77E-05	0.0991	9.84E-05
Biodeg	0.1773●	1.19E-03	0.1839●	1.16E-03	0.1823●	1.86E-03	0.1422●	8.31E-04	0.1273	6.66E-04
Blood	0.2861●	3.18E-03	0.2643●	1.46E-03	0.2664●	2.52E-03	0.2254	1.35E-03	0.2241	1.53E-03
Breast-cancer	0.0483●	7.88E-04	0.0449●	7.01E-04	0.0454●	7.62E-04	0.0434	8.81E-04	0.0371	5.23E-04
Bupa	0.3585●	5.38E-03	0.3606●	8.10E-03	0.3446	6.94E-03	0.2955	3.57E-03	0.3112	6.66E-03
Cleveland	0.4615●	5.21E-03	0.4631●	6.00E-03	0.4847●	7.82E-03	0.4251	3.20E-03	0.4051	5.19E-03
Contraceptive	0.5230●	1.59E-03	0.5130●	2.00E-03	0.4795●	1.69E-03	0.4415	1.64E-03	0.4410	1.34E-03
Fertility	0.1967●	1.50E-02	0.1833	1.54E-02	0.1733	9.29E-03	0.1267	3.96E-03	0.1433	5.79E-03
Haberman	0.3474●	4.33E-03	0.3312●	4.55E-03	0.2962	2.09E-03	0.2846	1.90E-03	0.2865	2.33E-03
Heart	0.2370●	6.46E-03	0.2185●	3.78E-03	0.2543●	7.02E-03	0.1889	4.60E-03	0.1778	2.78E-03
Hepatitis	0.2125	2.62E-02	0.1917	8.06E-03	0.1625	1.58E-02	0.1417	1.33E-02	0.1875	1.43E-02
Hill-valley	0.0868	1.39E-02	0.0823	1.58E-02	0.6179●	1.40E-01	0.3715●	4.97E-03	0.0679	5.59E-03
Led7digit	0.3293●	4.69E-03	0.3013●	5.20E-03	0.2947	5.09E-03	0.2673□	4.65E-03	0.2820	4.70E-03
Madelon	0.2370●	1.82E-03	0.2222	9.83E-04	0.3028●	6.19E-04	0.3060●	1.67E-03	0.2172	7.03E-04
Magic	0.2042●	1.89E-04	0.1735●	4.94E-05	0.1805●	5.87E-05	0.1726●	5.77E-05	0.1487	3.94E-05
Mammographic	0.2237●	2.55E-03	0.2092●	2.35E-03	0.2149●	1.96E-03	0.1851	1.20E-03	0.1799	1.78E-03
Musk1	0.0889●	1.73E-03	0.1002●	2.69E-03	0.1274●	3.13E-03	0.1534●	3.23E-03	0.0588	1.31E-03
Musk2	0.0062	8.10E-06	0.0076	5.82E-05	0.0168●	2.38E-05	0.0342●	3.81E-05	0.0056	1.18E-05
Newthyroid	0.0354	1.51E-03	0.0374	1.70E-03	0.0760●	4.46E-03	0.0761●	2.72E-03	0.0374	1.09E-03
Page-blocks	0.0355	5.84E-05	0.0370●	4.09E-05	0.0416●	3.68E-05	0.0478●	4.81E-05	0.0337	5.73E-05
Phoneme	0.1271●	4.04E-04	0.1172●	2.39E-04	0.1297●	1.85E-04	0.1462●	3.54E-04	0.1075	1.67E-04
Pima	0.3047●	3.03E-03	0.3077●	2.45E-03	0.2891●	1.88E-03	0.2435	2.57E-03	0.2370	2.01E-03
Ring	0.0301●	3.19E-05	0.0305●	3.70E-05	0.0985●	1.35E-04	0.1019●	5.30E-04	0.0211	4.46E-05
Sonar	0.2114●	7.56E-03	0.2391●	6.54E-03	0.1949	1.30E-02	0.2160●	7.41E-03	0.1634	6.77E-03
Spambase	0.0805●	1.43E-04	0.0826●	1.41E-04	0.1222●	3.22E-03	0.0845●	3.01E-04	0.0823	1.55E-02
Tae	0.4989	1.42E-02	0.0826	1.41E-04	0.4614	1.10E-02	0.4881	1.93E-02	0.4682	1.52E-02
Tic-tac-toe	0.0327	4.42E-04	0.0394	4.81E-04	0.0553●	4.31E-04	0.1204●	1.00E-03	0.0317	2.87E-04
Titanic	0.2267	1.79E-03	0.2161	3.26E-04	0.2359●	2.31E-03	0.2255●	5.97E-04	0.2170	3.68E-04
Vehicle	0.2514●	2.27E-03	0.2692●	1.68E-03	0.2972●	1.50E-03	0.2872●	1.23E-03	0.2242	9.85E-04
Vertebral	0.2022●	3.88E-03	0.1828	4.81E-03	0.1785	4.84E-03	0.1742	3.44E-03	0.1570	2.69E-03
Waveform-w-noise	0.1755●	2.45E-04	0.1725●	1.94E-04	0.1979●	7.42E-04	0.1641●	1.51E-04	0.1392	2.12E-04
Waveform-wo-noise	0.1773●	3.82E-04	0.1739●	2.44E-04	0.1783●	5.67E-04	0.1597●	3.92E-04	0.1346	3.12E-04
Wdbc	0.0392	5.48E-04	0.0369	6.24E-04	0.0545●	8.89E-04	0.0393	4.87E-04	0.0322	4.15E-04
Wine-red	0.4534●	1.86E-03	0.4332●	2.49E-03	0.4263●	1.84E-03	0.3900●	8.16E-04	0.3700	1.07E-03
Wine-white	0.4879●	7.54E-04	0.4578●	7.22E-04	0.4610●	8.94E-04	0.4226●	5.52E-04	0.4027	5.59E-04
Average ranking	**3.73**		**3.26**		**3.8**		**2.8**		**1.41**	

● and □ mean the proposed method is better or worse than the benchmark algorithm, respectively.

method rank first (with rank value of 2.04), followed by KNORA Union and KNORA Eliminate (with rank value of 2.95 and 3.09, respectively).

Compared to the benchmark algorithms, our approach has several advantages that explain the superior performance. First, GA Meta-data uses GA to select the meta-data's columns while fixes the meta-classifier, making it less flexible than our method. ACO-S1, meanwhile, selects the base classifiers and also fixes the meta-classifier.

Table 3 shows some examples of the selected meta-data's columns and meta-classifier of the proposed method. As mentioned before, instead of using classifier selection that removes all predictions of a classifier if it is not selected, we selected a subset of its prediction and a suitable meta-classifier. This makes our model more general and better than ACO-S1. Finally, the two DES methods perform poorer than our method because their performance depends on the choice of techniques that define the region associated with each test sample [6]. The average training time of proposed method computed on 30 test rounds on Abalone dataset is 3 seconds compared to 0.5 and 0.3 seconds of ACO-S1 and GA Meta-data, respectively. Although proposed method generally has longer running time than ACO-S1 and GA Meta-data, the differences are within practical limit.

4.3 Different number of learning algorithms

To evaluate the influence of using different number of learning algorithms on the ensemble performance, we added four learning algorithms to the previous set of learning algorithms introduced in Section 4.2. The newly added learning algorithms are Decision Tree, LibLinear [9], Nearest Mean Classifier, and Discriminative Restricted Boltzmann Machines [13]. The set of meta-classifiers were selected to be the same as the set of learning algorithms. The experimental results of the proposed method and the benchmark algorithms with the new ensemble system are shown in Table 3. The statistical test results in Fig. 5 once again show the superior performance of the proposed method compared to the benchmark algorithms: we win KNORA Eliminate and GA Meta-data on 30 datasets, wins ACO-S1 on 26 datasets and win KNORA Union on 22 datasets. The proposed method only loses KNORA Union on 1 dataset.

5 CONCLUSIONS

In summary, we have introduced a method to simultaneously select a subset of meta-data and a meta-classifier for the heterogeneous ensemble system to obtain higher classification accuracy than using the entire meta-data with one fixed meta-classifier. Our method first uses the cross validation procedure on the training dataset with the given learning algorithms to obtain the base classifiers and the meta-data. Having obtained the meta-data and the given set of meta-classifiers, we applied ACO to search for the optimal subset of meta-data and the associated meta-classifier. An ant will search around the local area based on the local information. In this study, we defined the local information as the classification accuracy associated with each meta-data's column and meta-classifier. Each ant's configuration including the candidate solution is evaluated by using another cross validation procedure on the selected meta-data. After ACO, we obtain the best configuration consisting of the subset of meta-data's columns and the associated meta-classifier for the ensemble. The classification process works in a straightforward manner by employing the best configuration on the test samples. Experiments conducted on 40 UCI datasets show that the proposed method is better than the benchmark algorithms we compared concerning the classification accuracy.

REFERENCES

[1] Iñigo Barandiaran. 1998. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence* 20, 8 (1998).

[2] Leo Breiman. 1996. Bagging predictors. *Machine learning* 24, 2 (1996), 123–140.

[3] Alceu S Britto Jr, Robert Sabourin, and Luiz ES Oliveira. 2014. Dynamic selection of classifiers - a comprehensive review. *Pattern Recognition* 47, 11 (2014), 3665–3680.

[4] Paulo R Cavalin, Robert Sabourin, and Ching Y Suen. 2013. Dynamic selection approaches for multiple classifier systems. *Neural Computing and Applications* 22, 3-4 (2013), 673–688.

[5] Yijun Chen, Man-Leung Wong, and Haibing Li. 2014. Applying Ant Colony Optimization to configuring stacking ensembles for data mining. *Expert systems with applications* 41, 6 (2014), 2688–2702.

[6] Rafael MO Cruz, Robert Sabourin, and George DC Cavalcanti. 2018. Dynamic classifier selection: Recent advances and perspectives. *Information Fusion* 41 (2018), 195–216.

[7] Manh Truong Dang, Anh Vu Luong, Tuyet-Trinh Vu, Quoc Viet Hung Nguyen, Tien Thanh Nguyen, and Bela Stantic. 2018. An Ensemble System with Random Projection and Dynamic Ensemble Selection. In *Asian Conference on Intelligent Information and Database Systems*. Springer, 576–586.

[8] Janez Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research* 7, Jan (2006), 1–30.

[9] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of machine learning research* 9, Aug (2008), 1871–1874.

[10] Josef Kittler, Mohamad Hatef, Robert PW Duin, and Jiri Matas. 1998. On combining classifiers. *IEEE transactions on pattern analysis and machine intelligence* 20, 3 (1998), 226–239.

[11] Albert HR Ko, Robert Sabourin, and Alceu Souza Britto Jr. 2008. From dynamic classifier selection to dynamic ensemble selection. *Pattern Recognition* 41, 5 (2008), 1718–1731.

[12] Ludmila I Kuncheva, James C Bezdek, and Robert PW Duin. 2001. Decision templates for multiple classifier fusion: an experimental comparison. *Pattern recognition* 34, 2 (2001), 299–314.

[13] Hugo Larochelle and Yoshua Bengio. 2008. Classification using discriminative restricted Boltzmann machines. In *Proceedings of the 25th international conference on Machine learning*. ACM, 536–543.

[14] Tien Thanh Nguyen, Alan Wee-Chung Liew, Xuan Cuong Pham, and Mai Phuong Nguyen. 2014. A novel 2-stage combining classifier model with stacking and genetic algorithm based feature selection. In *International Conference on Intelligent Computing*. Springer, 33–43.

[15] Tien Thanh Nguyen, Alan Wee-Chung Liew, Xuan Cuong Pham, and Mai Phuong Nguyen. 2014. Optimization of ensemble classifier system based on multiple objectives genetic algorithm. (2014), 46–51 pages.

[16] Tien Thanh Nguyen, Alan Wee-Chung Liew, Minh Toan Tran, Xuan Cuong Pham, and Mai Phuong Nguyen. 2014. A novel genetic algorithm approach for simultaneous feature and classifier selection in multi classifier system. In *Evolutionary Computation (CEC), 2014 IEEE Congress on*. IEEE, 1698–1705.

[17] Tien Thanh Nguyen, Mai Phuong Nguyen, Xuan Cuong Pham, and Alan Wee-Chung Liew. 2018. Heterogeneous classifier ensemble with fuzzy rule-based meta learner. *Information Sciences* 422 (2018), 144–160.

[18] Tien Thanh Nguyen, Mai Phuong Nguyen, Xuan Cuong Pham, Alan Wee-Chung Liew, and Witold Pedrycz. 2018. Combining heterogeneous classifiers via granular prototypes. *Applied Soft Computing* 73 (2018), 795–815.

[19] Tien Thanh Nguyen, Thi Thu Thuy Nguyen, Xuan Cuong Pham, and Alan Wee-Chung Liew. 2016. A novel combining classifier method based on Variational Inference. *Pattern Recognition* 49 (2016), 198–212.

[20] Tien Thanh Nguyen, Xuan Cuong Pham, Alan Wee-Chung Liew, and Witold Pedrycz. 2018. Aggregation of classifiers: a justifiable information granularity approach. *IEEE Transactions on Cybernetics* (2018).

[21] Simon Parsons. 2005. Ant Colony Optimization by Marco Dorigo and Thomas Stützle, MIT Press, 305 pp., 40.00, ISBN 0-262-04219-3. *The Knowledge Engineering Review* 20, 1 (2005), 92–93.

[22] Mehmet Umut ŞEn and Hakan Erdogan. 2013. Linear classifier combination and selection using group sparse regularization and hinge loss. *Pattern Recognition Letters* 34, 3 (2013), 265–274.

[23] Palanisamy Shunmugapriya and S Kanmani. 2013. Optimization of stacking ensemble configurations through artificial bee colony algorithm. *Swarm and Evolutionary Computation* 12 (2013), 24–32.

[24] Paul C Smits. 2002. Multiple classifier systems for supervised remote sensing image classification based on dynamic classifier selection. *IEEE Transactions on Geoscience and Remote Sensing* 40, 4 (2002), 801–813.

[25] Kai Ming Ting and Ian H Witten. 1999. Issues in stacked generalization. *Journal of artificial intelligence research* 10 (1999), 271–289.

[26] Chun-Xia Zhang and Robert PW Duin. 2011. An experimental study of one-and two-level classifier fusion for different sample sizes. *Pattern Recognition Letters* 32, 14 (2011), 1756–1767.

[27] Yi Zhang, Samuel Burer, and W Nick Street. 2006. Ensemble pruning via semi-definite programming. *Journal of Machine Learning Research* 7, Jul (2006), 1315–1338.

SUPPLEMENT MATERIAL

Paper: Simultaneous Meta-Data and Meta-Classifier Selection in Multiple Classifier System

Algorithm 1: Training process

Input: K learning algorithms, Q meta-classifiers, and the training set $\boldsymbol{\mathcal{D}}$

Output: The best configuration S_{best} and the base classifers

```
Generate the base classifiers and meta-data $\mathbf{L}$
Initialize settings: $na, \rho, \tau, CC, maxT$
Calculate local information $\eta$ using Algorithm 2
While $T < maxT$
  (a) For $j$ from 1 to $na$, the $j^{th}$ ant begins its searching
    Initialize the $S_j = \{C, \emptyset\}$ by selecting a meta-classifier $C$ at random
    Initialize $\alpha_{S_j} = -\infty$
    Set the flag $search\_next = true$
    While $search\_next = true$
      Select a column $\mathbf{L}^i$ from $\mathbf{L}$ using roulette wheel selection with probabilities given in equation (2)
      If no feature can be selected
        Set $search\_next = false$
      Else
        Add column $\mathbf{L}^i$ to generate new configuration $S_j' = S_j \cup \mathbf{L}^i$
        Calculate $\alpha_{S_j'}$ using Algorithm 3
        If $\alpha_{S_j'} > \alpha_{S_j}$
          $S_j = S_j'$
          Update the pheromone of $\mathbf{L}^i$ by the rule given in (6)
        Else
          Set $search\_next = false$
  (b) Evaporation works by the rule given in (7) after an iteration ends
```

```
    (c) $T = T + 1$
Get the best configuration $S_{best} = arg\max_{S} \alpha_S$ in the final iteration
```

Algorithm 2: Calculate local information

Input: Meta-data $\mathbf{L}$, Q Meta-classifiers $\{C_1, C_2, \dots, C_Q\}$

Output: Local information $\boldsymbol{\eta}$

```
For $q$ from 1 to $Q$
    For $i$ from 1 to $K \times M$
        $\{\mathbf{L}_1^i, \mathbf{L}_2^{\mathrm{i}}, \dots, \mathbf{L}_{T_2}^i\}$ = crossvalid($\mathbf{L}^i$)
        $sum\_error = 0$
        For $r$ from 1 to $T_2$
            Train $C_q$ on $\tilde{\mathbf{L}}_r^i = \mathbf{L}^i - \mathbf{L}_r^i$
            $sum\_error = sum\_error$ + error when using $C_q$to predict on $\mathbf{L}_r^i$
        $\eta_{i,q} = 1 - sum\_error/T_2$
Output local information $\boldsymbol{\eta} = \{\, \eta_{i,q}\}$
```

Algorithm 3: Calculate the accuracy of a configuration

Input: Configuration $S_j' = \{C, \mathbf{L}^{\{u\}}\}; \mathbf{L}^{\{u\}} = [\mathbf{L}^{u_1}\, \mathbf{L}^{u_2} \dots \mathbf{L}^{u_k}]$

Output: Evaluation criterion of S_j': $\alpha_{S_j'}$

```
$\{\mathbf{L}_1^{\boldsymbol{u}}, \mathbf{L}_2^{\boldsymbol{u}}, \dots, \mathbf{L}_{T_3}^{\boldsymbol{u}}\} = crossvalid(\mathbf{L}^{\boldsymbol{u}})$
$sum\_error = 0$
For $r$ from 1 to $T_3$
    Train $C$ on $\tilde{\mathbf{L}}_r^{\boldsymbol{u}} = \{\mathbf{L}_1^{\boldsymbol{u}}, \mathbf{L}_2^{\boldsymbol{u}}, \dots, \mathbf{L}_{T_3}^{\boldsymbol{u}}\} - \mathbf{L}_r^{\boldsymbol{u}}$
    Compute $\boldsymbol{\mathcal{L}}_{0-1}\left\{\mathbf{L}_r^{\{u\}}, S_j'\right\}$ by (3)
End
Compute $\alpha_{S_j'}$ by (5)
```

Algorithm 4: Testing process

Input: K base classifiers, best configuration S_{best}, test sample $\mathbf{x}^{test}$

Output: Predicted label for $\mathbf{x}^{test}$

```
1. Generate the $\mathbf{L}(\mathbf{x}^{test})$ using $K$ base classifiers
2. Using $S_{best}$ to determine the corresponding subset $\mathbf{L}^{\boldsymbol{I}}(\mathbf{x}^{test}) \subset \mathbf{L}(\mathbf{x}^{test})$ and the meta-classifier $C$
3. Using the meta-classifier $C$ to make predictions on $\mathbf{L}^{\boldsymbol{I}}(\mathbf{x}^{test})$
```