Multiobjective Fuzzy Genetics-Based Machine Learning for Multi-Label Classification

Yuichi Omozaki
Department of Computer Science and Intelligent Systems
Graduate School of Engineering
Osaka Prefecture University
Osaka, Japan
yuichi.omozaki@ci.cs.osakafu-u.ac.jp

Naoki Masuyama
Department of Computer Science and Intelligent Systems
Graduate School of Engineering
Osaka Prefecture University
Osaka, Japan
masuyama@cs.osakafu-u.ac.jp

Yusuke Nojima
Department of Computer Science and Intelligent Systems
Graduate School of Engineering
Osaka Prefecture University
Osaka, Japan
nojima@cs.osakafu-u.ac.jp

Hisao Ishibuchi*
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, China
hisao@sustech.edu.cn

***Abstract*—In multi-label classification problems, multiple class labels are assigned to each instance. Two approaches have been studied in the literature. One is a data transformation approach, which transforms a multi-label dataset into a number of single-label datasets. However, this approach often loses the correlation information among classes in the multi-class assignment. The other is a method adaptation approach where a conventional classification method is extended to multi-label classification. Recently, some explainable classification models for multi-label classification have been proposed. Their high interpretability has also been discussed with respect to the transparency of the classification process. Although the explainability is a well-known advantage of fuzzy systems, their applications to multi-label classification have not been well studied. Since multi-label classification problems often have vague class boundaries, fuzzy systems seem to be a promising approach to multi-label classification. In this paper, we propose a new multiobjective evolutionary fuzzy system, which can be categorized as a method adaptation approach. The proposed algorithm produces non-dominated classifiers with different tradeoffs between accuracy and complexity. We examine the behavior of the proposed algorithm using synthetic multi-label datasets. We also compare the proposed algorithm with five representative algorithms. Our experimental results on real-world datasets show that the obtained fuzzy classifiers with a small number of fuzzy rules have high transparency and comparable generalization ability to the other examined multi-label classification algorithms.**

Keywords—multi-label classification, multiobjective fuzzy genetics-based machine learning, fuzzy rule-based classification system, method adaptation approach.

I. INTRODUCTION

In real-world data mining applications, multiple class labels can be assigned to an instance. Recently, the handling of such a multi-label dataset (MLD) has been a hot research topic [1]. Real-world classification problems including text and music categorization [2, 3], web data mining [4], image semantic annotation [5], and medical diagnosis [6] have been studied as multi-label classification. Multi-label classifier design can be categorized into two approaches: a data transformation approach and a method adaptation approach [1]. In the data transformation approach, an MLD is transformed into one or multiple single-label datasets. Therefore, any classifier for single-label datasets can be directly applied to the transformed datasets. However, in the data transformation approach, designed classifiers do not consider the correlation information among classes in the multi-class assignment. On the other hand, in the method adaptation approach, single-label classifier design is extended to multi-label classification. Most of them consider the correlation information among classes [7-11].

Multi-label classification problems often have some degree of vagueness in classification boundaries [12]. Fuzzy rule-based classifiers can handle vague classification boundaries. In [12], a fuzzy rule-based system reported good results for this reason. However, as mentioned above, the data transformation approach does not consider the correlation among classes. Thus, we focus on the method adaptation approach in this paper and propose a new fuzzy system design algorithm for multi-label classification. We use multiobjective fuzzy genetics-based machine learning (MoFGBML) [13] because it can generate simple but accurate fuzzy rule-based classifiers. We extend MoFGBML to multi-label classification. The proposed algorithm, which is called $\text{MoFGBML}_{\text{ML}}$, generates a number of non-dominated fuzzy rule-based classifiers with different accuracy-complexity tradeoffs for multi-label classification. The contributions of this paper are as follows:

1. Two classification methods (i.e., *CF-vector* and *CF-mean*) specialized for MLDs are proposed.
2. Their characteristics are examined through computational experiments on synthetic MLDs.
3. The high transparency and comparative generalization ability of the classifiers obtained by $\text{MoFGBML}_{\text{ML}}$ are shown by computational experiments on six real-world MLDs.

The rest of this paper is organized as follows. We introduce multi-label classification in Section II. We briefly explain MLDs, performance metrics, and some data transformation algorithms

This work was partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant JP19K12159, and was supported in part by National Natural Science Foundation of China (Grant No. 61876075), Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531), the Program for University Key Laboratory of Guangdong Province (Grant No. 2017KSYS008).

*Corresponding author: Hisao Ishibuchi (hisao@sustech.edu.cn).

in Section III. We describe MoFGBML$_{ML}$ and its characteristics in Section IV. In Section V, we compare MoFGBML$_{ML}$ with two data transformation algorithms and three method adaptation algorithms. Finally, we conclude this paper in Section VI.

II. Related Works

Rule-based classifiers have been widely studied because of their high interpretability. In [12], Prati proposed a fuzzy rule-based classifier via the data transformation approach. The results showed that the fuzzy rule-based classifier was very competitive with other rule-based algorithms based on the data transformation approach. These results indicated the advantage of fuzzy classifiers for multi-label classification.

In terms of interpretability, decision tree-based algorithms have been widely studied, which is explainable by their node conditions. C4.5 [14] is one of the most well-known decision tree-based algorithms and was also extended to multi-label classification [7]. Fuzzy ID3 [15] is one of the most commonly used fuzzy decision tree-based algorithms. It was also extended into FuzzDT$_{ML}$ [8]. It achieved good performance and showed the applicability of fuzzy systems for multi-label classification.

Recently, explainable method adaptation algorithms have started to be studied. In [9], Nazmi et al. proposed an interval rule-based method adaptation algorithm MLRBC (Multi-Label Rule-Based Classifier), which is based on genetics-based machine learning. In 2019, Klein et al. proposed a rule-based plug-in approach to relax the search space pruning [10]. In [11], Panigutti et al. proposed a method adaptation algorithm MARLENA (Multi-lAbel RuLe-based ExplaNAtoins), which tries to make black-box classifiers to be explainable.

Although there exist some explainable method adaptation algorithms, fuzzy rule-based method adaptation algorithm has not been proposed to the best of our knowledge.

III. Multi-Label Classification Problem

In this section, we explain the formulation of MLDs, several performance metrics for multi-label classification, and some data transformation algorithms.

A. Formulation of Multi-Label Datasets

Let $L = \{\lambda_k \mid k = 1, ..., K\}$ be the set of class labels existing in the MLD, where K is the number of classes. Let us also assume that we have an n-dimensional MLD $D = \{(\mathbf{x}_1, Y_1), (\mathbf{x}_2, Y_2), ..., (\mathbf{x}_m, Y_m)\}$ including m instances, where $\mathbf{x} = (x_1, ..., x_n)$ is an input attribute vector and $Y \subset L$ is a subset of class labels. Y_p includes assigned class labels to p-th instance.

In this paper, Y is denoted by a binary vector $\mathbf{y} = (y_1, ..., y_K)$, where $y_k = 1$ if Y contains the k-th class label λ_k, or $y_k = 0$ if Y does not contain that. We denote predicted class labels for an instance by a binary vector $\boldsymbol{z} = (z_1, ..., z_K)$ or a subset $Z \subset L$.

B. Performance Metrics

In this subsection, we explain several commonly used performance metrics for multi-label classification problems [1, 7]. Since our proposed method is not a ranking-based algorithm, we use only example-based metrics in this paper.

1) Subset Accuracy (SubAcc): This is possibly the most strict performance metric [1]. This metric measures whether a predicted class-vector $\boldsymbol{z}$ for an instance is exactly equal to the true class-vector $\mathbf{y}$ for the instance. *SubAcc* is calculated by Eq. (1).

$$SubAcc = \frac{1}{m}\sum_{p=1}^{m}[[Y_p = Z_p]], \tag{1}$$

where the operator [[*condition*]] returns 1 if the *condition* is satisfied, or 0 if it is not satisfied.

2) Hamming Loss (HL): This performance metric is often used in the field of multi-label classification. This metric is calculated by Eq. (2).

$$HL = \frac{1}{m}\sum_{p=1}^{m}\frac{Y_p \,\Delta\, Z_p}{K}, \tag{2}$$

where the operator Δ returns the symmetric difference of the truly relevant class subset Y and the predicted class subset Z.

3) Precision and Recall: Precision denotes the proportion between the number of correctly predicted classes and the number of all predicted classes. *Recall* denotes the proportion between the number of correctly predicted classes and the number of the truly relevant classes to an instance. *Precision* means the percentage of predicted labels which are truly relevant for the instance. *Recall* means the percentage of labels correctly predicted among all truly relevant labels [1].

$$Precision = \frac{1}{m}\sum_{p=1}^{m}\frac{|Y_p \cap Z_p|}{|Z_p|}, \tag{3}$$

$$Recall = \frac{1}{m}\sum_{p=1}^{m}\frac{|Y_p \cap Z_p|}{|Y_p|}. \tag{4}$$

4) F-Measure (FM): This is a joint of *Precision* and *Recall*, a weighted measure of how many true relevant classes are predicted and how many predicted classes are truly relevant. This is calculated as the harmonic mean as follows:

$$FM = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}. \tag{5}$$

C. Data Transformation Approach

A data transformation approach is one of the earliest approaches to tackle multi-label classification problems [1]. In this approach, an MLD is transformed into one or multiple single-label datasets. Since the transformed single-label datasets can be regarded as conventional classification problems, any conventional classifier can be applied to them. In this subsection, we explain two widely used data transformation algorithms: Binary Relevance (BR) and Label Power Set (LP).

1) Binary Relevance (BR): This algorithm divides an MLD into K binary-classification problems. An instance in the transformed k-th problem has a label which indicates whether it is associated with the k-th class or not. K classifiers learn each single-label dataset and classify whether an instance is associated with each class. However, it is necessary to design as many classifiers as the number of classes.

2) Label Power Set (LP): This algorithm transforms multiple class labels into a single class label power set. While K datasets are produced in BR, only a single dataset is produced in LP. LP combines distinct class label power sets in the training dataset into a new class label. Thus, the class label power sets represent single-label classes. In other words, an MLD is transformed into a conventional multi-class dataset in LP. Unlike the BR, this algorithm maintains the correlation among classes.

IV. Multiobjective Fuzzy Genetics-Based Machine Learning for Multi-Label Classification

It is said that fuzzy rule-based classifiers have high interpretability. However, it is not always true because of the existence of the accuracy-complexity tradeoff in the design of fuzzy rule-based classifiers. Furthermore, classification takes a long time in fuzzy systems with many rules. To deal with these issues, multiobjective evolutionary fuzzy systems have actively been studied [16-18]. One of the most representative methods is MoFGBML proposed in [13]. MoFGBML uses an evolutionary multiobjective optimization algorithm (EMOA) to generate several non-dominated fuzzy rule-based classifiers, aiming to minimize both the classification error and the complexity of the obtained classifiers.

In this paper, we propose two classification methods specialized for MLDs and a new method adaptation algorithm named $\text{MoFGBML}_{\text{ML}}$ by extending MoFGBML to multi-label classification.

A. A Fuzzy Rule-based Classifier for Multi-Label Classification

In $\text{MoFGBML}_{\text{ML}}$, a fuzzy rule-based classifier is composed of a set of fuzzy if-then rules. We extend the conventional formulation of fuzzy if-then rules [19] as follows:

$$\begin{aligned}&\text{Rule } R_q\text{: If } x_1 \text{ is } A_{q1} \text{ and ... and } x_n \text{ is } A_{qn}\\ &\qquad \text{then Class } \boldsymbol{z}_q \text{ with } \boldsymbol{CF}_q,\ q = 1, 2, ..., N,\end{aligned} \tag{6}$$

where R_q denotes the q-th rule, $\mathbf{A}_q = (A_{q1}, ..., A_{qn})$ is the q-th antecedent fuzzy set, and N is the number of fuzzy if-then rules in a fuzzy system. $\boldsymbol{z}_q$ and $\boldsymbol{CF}_q = (CF_{q1}, ..., CF_{qK})$ are a consequent class-vector and a rule weight vector, respectively, which are extended to new consequent parts. The vectors are K-dimensional vectors, in which each element represents each class association.

1) Determination of the consequent part of fuzzy if-then rules: First, to determine the consequent part, we calculate the compatibility grade $\mu_{\mathbf{A}q}(\mathbf{x})$ between an antecedent fuzzy set $\mathbf{A}_q$ and an instance $\mathbf{x}$ by Eq. (7).

$$\mu_{\mathbf{A}_q}(\mathbf{x}) = \prod_{i=1}^{n} \mu_{A_{qi}}(x_i), \tag{7}$$

where $\mu_{Aqi}(\cdot)$ is the membership function representing the fuzzy set A_{qi}. Then, the confidence $c_k(\cdot)$ for each class is calculated, where the argument can be the linguistic association rule "$\mathbf{A}_q \Rightarrow \lambda_k \in Z_q$" or "$\mathbf{A}_q \Rightarrow \lambda_k \notin Z_q$". The confidence $c_k(\mathbf{A}_q \Rightarrow \lambda_k \in Z_q)$ is calculated as follows:

$$c_k\left(\mathbf{A}_q \Rightarrow \lambda_k \in Z_q\right) = \frac{\sum_{\left\{\mathbf{x}_p \mid y_{pk} \in Z_q\right\} \subset D} \mu_{\mathbf{A}_q}\left(\mathbf{x}_p\right)}{\sum_{p=1}^{m} \mu_{\mathbf{A}_q}\left(\mathbf{x}_p\right)}. \tag{8}$$

$c_k(\mathbf{A}_q \Rightarrow \lambda_k \notin Z_q)$ is also calculated in the same way. In this paper, "$\lambda_k \in Z_q$" and "$\lambda_k \notin Z_q$" mean "$z_{qk} = 1$" as "$z_{qk} = 0$", respectively. Finally, the consequent class-vector $\boldsymbol{z}_q$ and rule weight vector $\boldsymbol{CF}_q$ are determined with the confidence values as follows:

$$z_{qk} = \arg \max_{j \in \{0,1\}} c_k\left(\mathbf{A}_q \Rightarrow z_{qk} = j\right), \tag{9}$$

$$CF_{qk} = \left| c_k\left(\mathbf{A}_q \Rightarrow \lambda_k \in Z_q\right) - c_k\left(\mathbf{A}_q \Rightarrow \lambda_k \notin Z_q\right) \right|. \tag{10}$$

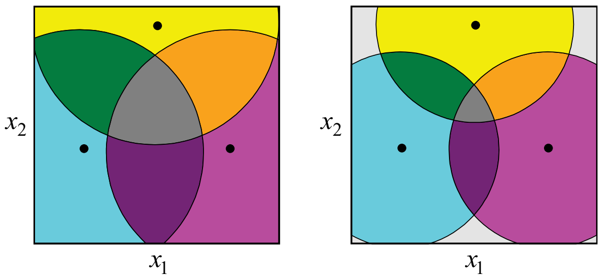
2) Classification process: $\text{MoFGBML}_{\text{ML}}$ takes a single winner-based method [19] to classify an instance $\mathbf{x}$. We propose two types of winner-rule selection methods: *CF-mean* method and *CF-vector* method. In the *CF-mean* method, a winner-rule is chosen by the maximum value calculated as follows:

$$\begin{aligned}&\textit{Winner } R_w = \max_q \left\{ \mu_{\mathbf{A}_q}(\mathbf{x}) \cdot \overline{CF}_q \mid R_q \in S \right\},\\ &\text{where } \overline{CF}_q = \frac{1}{K} \sum_{k=1}^{K} CF_{qk},\end{aligned} \tag{11}$$

and S is a set of fuzzy if-then rules in a fuzzy rule-based classifier. After the single winner-rule R_w is chosen, the consequent class-vector $\boldsymbol{z}_w$ is assigned to the instance $\mathbf{x}$. If there exist winner-rules with the same maximum value having different class-vectors, the classification is rejected. In this paper, rejected class labels are denoted as "-1".

In the *CF-vector* method, a single winner-rule is chosen for each class. That is, multiple winner-rules are often chosen for an instance. To do that, Eq. (12) is used instead of (11).

$$\textit{Winner } R_w^k = \max_q \left\{ \mu_{\mathbf{A}_q}(\mathbf{x}) \cdot CF_{qk} \mid R_q \in S \right\}. \tag{12}$$



(a) Synthesis 1 (b) Synthesis 2

Fig. 1. Examples of the created synthetic multi-label datasets.

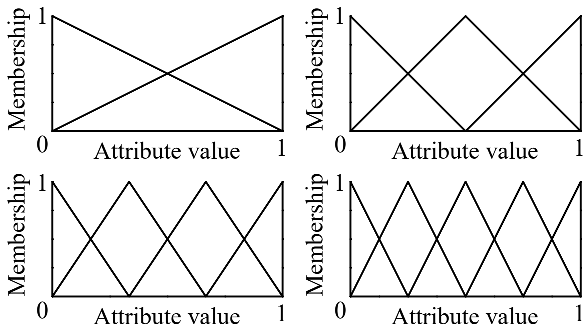
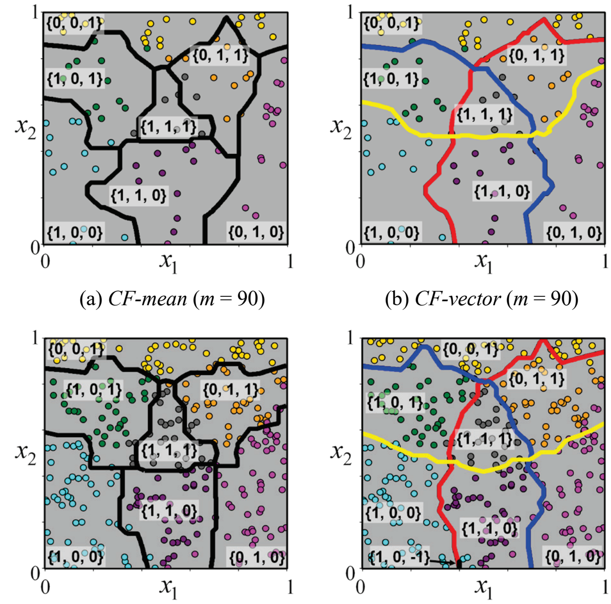


Fig. 2. Triangular fuzzy sets.

Then, each z_{wk} of $R^k{}_w$ is assigned to the instance **x**. Even if multiple if-then rules are chosen, the interpretability of the fuzzy system is preserved since the antecedent fuzzy sets $\mathbf{A}^k{}_w$ of the winner $R^k{}_w$ can explain why the k-th class label is assigned (or non-assigned) to the instance **x**.

B. Characteristics of the Proposed Classification System

We show the characteristics of $\text{MoFGBML}_{\text{ML}}$ with two synthetic MLDs in this subsection. In this paper, we create two-dimensional synthetic MLDs shown in Fig. 1. These MLDs have two input variables and three classes, "*Cyan*", "*Magenta*", and "*Yellow*". The true classification boundary is defined with three center points and three radiuses i.e., a class is defined by a center point and a radius. If an instance is in an area of a class, its class label is assigned to the instance. In Fig. 1, true classification boundaries are shown with the seven colors. The areas of "*Cyan*", "*Magenta*", and "*Yellow*" assign each class label to an instance in the circle. The areas of "*Green*", "*Orange*", and "*Purple*" represent those of the subset of class labels, {"*Cyan*", "*Yellow*"}, {"*Magenta*", "*Yellow*"}, and {"*Cyan*", "*Magenta*"}, respectively. In the same way, the area of "*Gray*" assigns all class labels to an instance. We randomly generate instances in the range of [0, 1], and assign the class labels to the instances based on the above definition. For investigating the classification characteristics of our proposed methods, we use 14 antecedent fuzzy sets with four different granularities shown in Fig. 2 together with a *don't care* condition. Thus, the number of possible rules is 15^2. In this experiment, we use all the possible rules for classification. We use triangular-shaped fuzzy sets because they have been used in MoFGBML and a linguistic label can be easily assigned to each fuzzy set.



(a) *CF-mean* (m = 90) (b) *CF-vector* (m = 90)

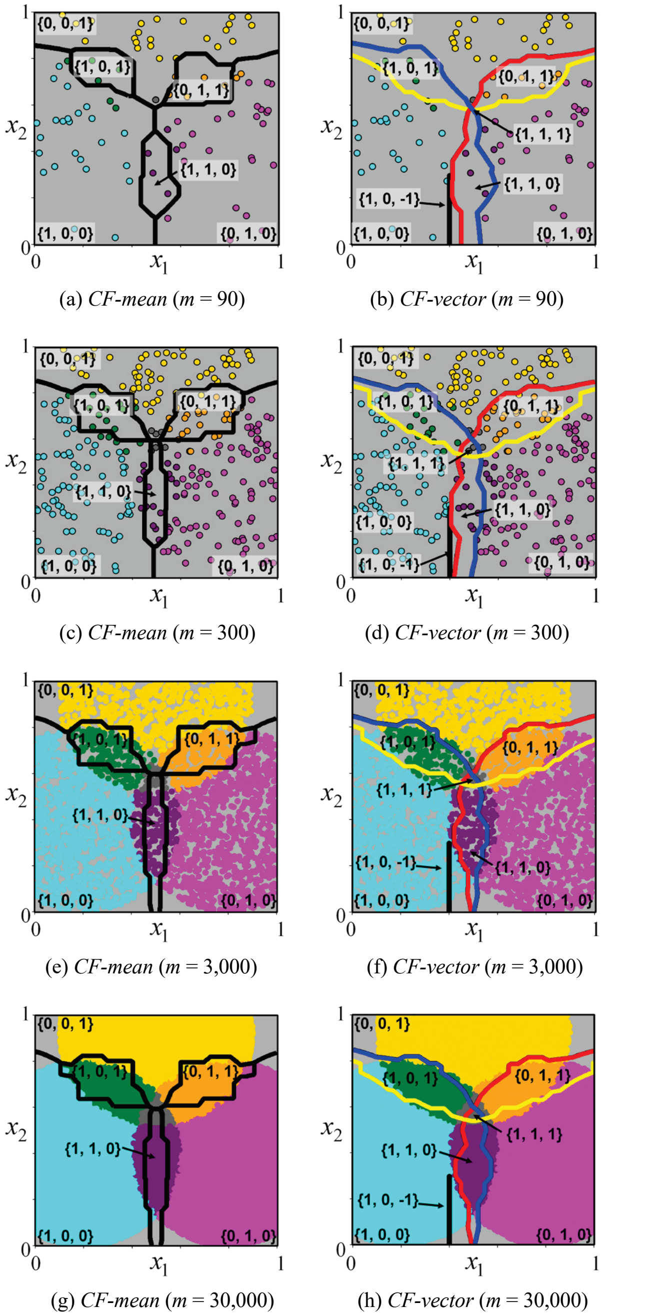
(c) *CF-mean* (m = 300) (d) *CF-vector* (m = 300)

Fig. 3. Classification boundaries by our proposed method for Fig. 1 (a).

Fig. 3 shows the classification boundaries for the dataset in Fig. 1 (a) by two types of winner-rule selection methods *CF-mean* method and *CF-vector* method. We examined two specifications of the number of instances: 90 and 300. In Fig. 3, both methods showed similar performance for the dataset in Fig. 1 (a) regardless of the number of instances. However, for the dataset in Fig. 1 (b), the two methods showed different results. Fig. 4 shows the classification boundaries for the dataset in Fig. 1 (b). In Fig. 4, we examined four specifications of the number of instances: 90, 300, 3,000, and 30,000. Unlike the *CF-mean* method, a classifier with the *CF-vector* method had a classification area that can be classified as *Gray* for the dataset in Fig. 1 (b). Actually, no fuzzy if-then rule (with the 15 types of conditions) learned the consequent class-vector $\boldsymbol{z}$ = (1, 1, 1) for the dataset in Fig. 1 (b) regardless of the number of instances. It means that any classifier with the *CF-mean* method basically cannot classify any instance in the dataset as *Gray*. However, since a classifier with the *CF-vector* method classifies an instance into each class independently based on each single winner-rule, it is considered that the classifier can classify an instance even if it is assigned an unlearned subset of class labels. As a result, although no fuzzy rule included the consequent class-vector, which is a minor subset of class labels, the classifier could classify the unknown subset of class labels with the *CF-vector* method. It is important to consider the imbalance class label power sets.

C. Multiobjective Optimization Approach

The classification performance by the proposed *CF-vector* method is actually the same as that by BR in terms of the deterministic classifier design. The correlation among classes is not considered in BR even though any optimization method is applied. However, since $\text{MoFGBML}_{\text{ML}}$ can consider the correlation, optimization methods work for the classifier design



(a) *CF-mean* (m = 90) (b) *CF-vector* (m = 90)

(c) *CF-mean* (m = 300) (d) *CF-vector* (m = 300)

(e) *CF-mean* (m = 3,000) (f) *CF-vector* (m = 3,000)

(g) *CF-mean* (m = 30,000) (h) *CF-vector* (m = 30,000)

Fig. 4. Classification boundaries by our proposed method for Fig. 1 (b).

efficiently. Therefore, we apply an EMOA to obtain the best combinations of fuzzy rules in terms of the accuracy-complexity tradeoff in the classification systems.

MoFGBML [13] is the hybrid method of the Pittsburgh and the Michigan approaches. In MoFGBML$_{ML}$, the Pittsburgh-style framework is used for the rule-set-wise optimization, while the Michigan-style operation is used as a local search for the rule-wise optimization. For the Pittsburgh part, three multiobjective problem (MOP) formulations using different performance metrics are defined by formulations (13) - (15).

$$\text{MOP1: Maximize } SubAcc(S) \text{ and minimize } N(S), \tag{13}$$

$$\text{MOP2: Minimize } HL(S) \text{ and minimize } N(S), \tag{14}$$

$$\text{MOP3: Maximize } FM(S) \text{ and minimize } N(S), \tag{15}$$

where $N(S)$ is the number of rules in S, and $SubAcc(S)$, $HL(S)$, and $FM(S)$ are performance metric functions i.e., Eqs. (1), (2), and (5). The first objective in each MOP is defined to minimize the classification error. The second objective is defined to minimize the complexity. For the Michigan part, we replace the rule-fitness with Eqs. (16) and (17) for the *CF-mean* method and the *CF-vector* method, respectively.

$$f_{rule}\left(R_q\right) = \sum_{\mathbf{x}_p \in M_q} \sum_{k=1}^{K} \frac{[[y_{pk} = z_{qk}]]}{K}, \tag{16}$$

$$f_{rule}\left(R_q^k\right) = \sum_{\mathbf{x}_p \in M_q} [[y_{pk} = z_{qk}]], \tag{17}$$

where $M_q \subset D$ is the set of instances which are classified by R_q. Eq. (16) assesses how much a rule R_q correctly predicts the class label power set, while Eq. (17) assesses how much it correctly predicts any class.

V. Computational Experiments

We showed the characteristics of our proposed method with the deterministic set of fuzzy rules in Section IV. In this section, we use EMOA and compare our proposed method (based on *CF-mean* and *CF-vector*) with two data transformation algorithms (BR and LP) which applied to MoFGBML. In addition, we also compare them with three explainable method adaptation algorithms, MLC4.5, FuzzDT$_{ML}$, and MLRBC.

A. Experimental Settings

Table I shows the parameter specifications of our proposed method. In this section, we used 14 kinds of fuzzy sets in Fig. 2 and the "*don't care*" condition, which always returns 1 for any input value. Finally, we performed with each algorithm over 30 runs for each MLD i.e., 10-fold cross- validation 3 times with different random seeds. All the algorithms used the same data partitions.

TABLE I. Parameter Specifications.

Parameter	Specification
EMOA	NSGA-II [20]
Number of generations	1,000
Population size	60
Offspring size	60
Maximum number of rules	60
Minimum number of rules	1
Crossover probability	0.9
Mutation probability	1 / number of genes 1 / N (Pittsburgh-style) 1 / n (Michigan-style)
Michigan-operation probability	0.5
The ratio of replaced rules in the Michigan-operation	0.2 0.1 by crossover and mutation 0.1 by heuristic rule generation
Don't care probability	$(n - 5) / n$

MLC4.5 is implemented in Clus[a]. FuzzDT$_{ML}$[b] and MLRBC[c] are placed at the authors' GitHub repositories. We used the algorithms' specific parameters from their proposed papers in our experiments. We also placed the source-codes of MoFGBML$_{ML}$ into our GitHub repository[d].

Table II shows the details of the datasets used in this paper. We selected four datasets which have only numerical input variables and two datasets which includes a few nominal those from the Mulan repository [21]. There are some specific terms to describe MLDs, *Cardinality*, *Density*, and *Distinct* [12]. *Cardinality* is the average number of associated classes per instance. *Density* is the ratio between the cardinality and the number of associated classes. *Distinct* is the number of unique subsets of class labels in a dataset.

B. Experimental Results

Table III, Table IV, and Table V show the average results of the performance metrics, SubAcc, HL, and FM, for the test datasets, respectively. In this experiment, a rejected instance is

TABLE II. DETAILS OF MULTI-LABEL DATASETS USED IN THE EXPERIMENTS.

Dataset	**Number of Instances**	**Number of Attributes**		**Number of Classes**	***Cardinality***	***Density***	***Distinct***	**Domain**
		Nominal	**Numeric**					
CAL500	502	0	68	174	26.044	0.150	502	Music
Emotions	593	0	72	6	1.869	0.311	27	Music
Scene	2,407	0	294	6	1.074	0.179	15	Image
Yeast	2,417	0	103	14	4.237	0.303	198	Biology
Flags	194	9	10	7	3.392	0.485	54	Images
Birds	645	2	258	19	1.014	0.053	133	Audio

TABLE III. AVERAGE SUBSET ACCURACY RATES ON TEST DATA (THE HIGHER, THE BETTER).

Dataset	**MoFGBML$_{ML}$ *CF-mean***			**MoFGBML$_{ML}$ *CF-vector***			**Data Transformation**		**Explainable Method Adaptation**		
	MOP1	MOP2	MOP3	MOP1	MOP2	MOP3	BR	LP	MLC4.5	FuzzDT$_{ML}$	MLRBC
CAL500	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Emotions	†27.31	25.41	†**27.93**	†27.44	25.63	†**27.94**	23.27	22.83	21.93	15.85	†28.15
Scene	†52.43	31.25	†**53.32**	**50.53**	32.38	48.47	35.41	42.92	†52.61	19.46	1.51
Yeast	†**17.02**	10.22	15.34	†**16.41**	9.88	14.39	8.88	1.74	12.47	7.99	12.67
Flags	†**25.05**	17.32	16.30	†**26.48**	18.36	14.04	14.91	3.62	13.92	10.82	18.25
Birds	†**48.69**	46.00	4.39	†**47.97**	46.21	4.34	43.56	45.59	41.13	†49.40	†46.88

TABLE IV. AVERAGE HAMMING LOSS RATES ON TEST DATA (THE LOWER, THE BETTER).

Dataset	**MoFGBML$_{ML}$ *CF-mean***			**MoFGBML$_{ML}$ *CF-vector***			**Data Transformation**		**Explainable Method Adaptation**		
	MOP1	MOP2	MOP3	MOP1	MOP2	MOP3	BR	LP	MLC4.5	FuzzDT$_{ML}$	MLRBC
CAL500	87.27	**14.26**	17.43	86.99	**14.17**	16.77	15.17	14.98	21.20	†13.70	14.46
Emotions	24.40	†**22.40**	23.85	26.17	†**21.67**	23.64	†21.57	27.38	26.17	24.88	†21.92
Scene	15.94	**14.23**	15.66	16.42	**14.07**	18.26	†12.87	18.55	14.76	15.22	98.17
Yeast	25.32	**21.75**	22.67	26.86	**21.45**	22.39	21.24	32.37	28.51	21.71	†20.65
Flags	†28.16	28.31	†**27.19**	†28.00	†**26.16**	†27.05	28.18	48.08	32.21	29.64	†27.01
Birds	6.74	**6.15**	20.49	6.96	**6.88**	17.39	6.11	6.03	6.30	†4.73	8.32

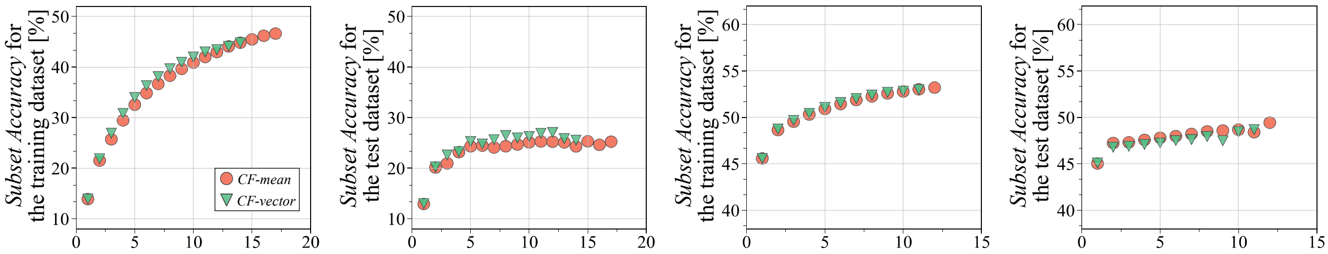
TABLE V. AVERAGE F-MEASURE RATES ON TEST DATA (THE HIGHER, THE BETTER).

Dataset	**MoFGBML$_{ML}$ *CF-mean***			**MoFGBML$_{ML}$ *CF-vector***			**Data Transformation**		**Explainable Method Adaptation**		
	MOP1	MOP2	MOP3	MOP1	MOP2	MOP3	BR	LP	MLC4.5	FuzzDT$_{ML}$	MLRBC
CAL500	5.18	33.22	†**40.10**	5.52	33.18	†**39.95**	33.33	0.00	36.90	32.67	21.18
Emotions	60.83	59.70	**63.39**	56.95	56.11	**61.98**	58.76	51.59	58.80	40.25	†66.32
Scene	56.59	34.23	**57.51**	**54.62**	35.12	53.17	41.32	46.72	†62.01	20.79	1.69
Yeast	59.18	57.11	†**61.68**	57.54	58.24	†**61.93**	59.30	31.09	55.67	54.74	60.03
Flags	69.33	70.96	†**73.83**	69.27	†72.98	†**73.25**	70.76	37.14	67.06	70.96	67.32
Birds	9.96	5.32	**19.76**	9.59	6.26	**19.12**	14.81	1.97	†24.27	12.28	19.89

TABLE VI. AVERAGE NUMBER OF RULES (THE SMALLER, THE BETTER).

Dataset	**MoFGBML$_{ML}$ *CF-mean***			**MoFGBML$_{ML}$ *CF-vector***			**Data Transformation**		**Explainable Method Adaptation**		
	MOP1	MOP2	MOP3	MOP1	MOP2	MOP3	BR	LP	MLC4.5	FuzzDT$_{ML}$	MLRBC
CAL500	26.07	12.73	**11.23**	27.50	26.43	**23.47**	492.13	4.13	193.20	312.23	11683.43
Emotions	10.23	**9.73**	11.27	**11.33**	13.03	12.90	43.47	9.90	150.40	332.03	14672.37
Scene	14.17	**9.60**	15.00	15.93	**13.83**	16.50	42.23	14.90	305.17	1190.13	14998.67
Yeast	9.30	**8.13**	9.20	**11.33**	12.67	11.80	66.90	10.63	773.40	1358.03	9943.03
Flags	**19.10**	25.10	23.27	**15.43**	28.60	33.60	68.83	16.10	69.83	135.63	13388.40
Birds	13.07	**9.53**	10.63	11.30	**11.00**	13.33	88.70	8.83	147.73	131.30	14949.30

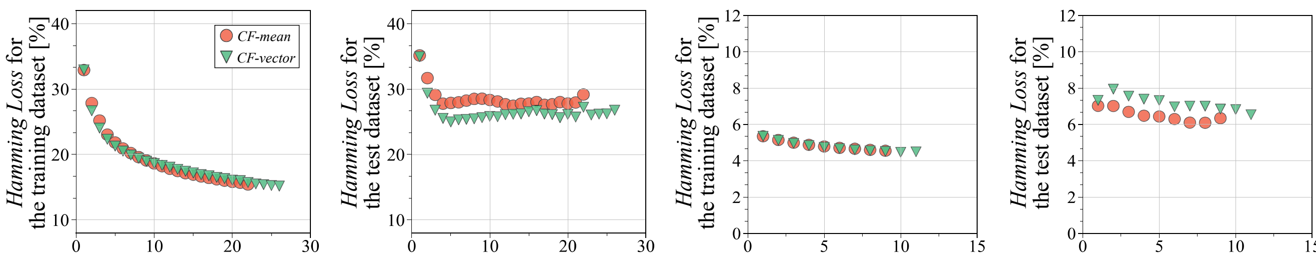
[a] http://clus.sourceforge.net
[b] https:/github.com/mhjabreel/FDTKit
[c] https://github.com/ShabnamNazmi/MLRBC
[d] https://github.com/CI-labo-OPU/MoFGBML_ML



(a) Flags

(b) Birds

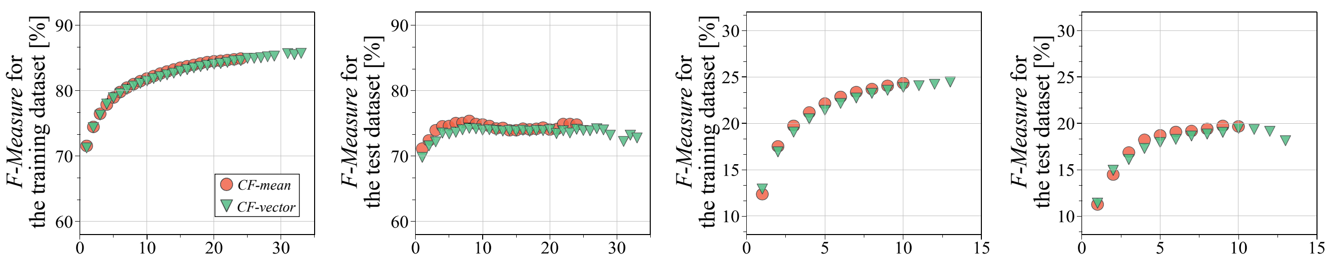
Fig. 5. The obtained non-dominated fuzzy rule-based classifiers in the objective space of MOP1 in (a) Flags and (b) Birds datasets.



(a) Flags

(b) Birds

Fig. 6. The obtained non-dominated fuzzy rule-based classifiers in the objective space of MOP2 in (a) Flags and (b) Birds datasets.



(a) Flags

(b) Birds

Fig. 7. The obtained non-dominated fuzzy rule-based classifiers in the objective space of MOP3 in (a) Flags and (b) Birds datasets.

regarded as a classification error (e.g., a none-matched instance with any rule is rejected in MLRBC). We show the best results out of three MOPs with the same classification method in Bold. The best results out of all algorithms are shown in Red. We also tested pairwise Wilcoxon signed-rank tests ($\alpha = 0.05$) between the best result out of all algorithms (namely shown in Red) and the others, respectively. Results which are not statistically significantly different from the best result are labelled with a dagger (†). From Table IV and Table V, MOP2 and MOP3 were considered as a suitable formulation for each performance metric, *HM* and *FM*, respectively. However, MOP1 did not always work well for *SubAcc*. This is because MOP1 is the most difficult problem since *SubAcc* is the strictest performance metric for multi-label classification. In Table III and Table V, almost the results of the *CF-mean* method on MOP1 and MOP3 are shown with a dagger. This means that the *CF-mean* method has the ability for *SubAcc* and *FM* abilities. On the other hand, our proposed methods were inferior for most of datasets in terms of the *HL* metric. Especially, BR and MLRBC showed better *HL* performance. This may be because they have the strength for *HL* since they optimize classifiers using the objective function similar to the *HL* metric.

Table VI shows the average number of rules. Despite a small number of rules, our proposed methods often outperform other algorithms in terms of accuracy.

In Fig. 5, Fig. 6, and Fig. 7, we show the obtained non-dominated classifiers on MOP1, MOP2, and MOP3, respectively. The vertical axis represents each performance metric and the horizontal axis represents the complexity (i.e., the number of rules). In the figures, we calculate the average performance metrics over all runs for each number of rules. We use only unique classifiers from non-dominated classifiers (i.e., the same classifiers are not used for the average) and plot the average performance metrics of only classifiers which have the number of rules obtained more than 16 runs. The tradeoff between each performance metric and its complexity can be clearly observed. Furthermore, overfitting to the training dataset is observed especially on MOP2 and MOP3. However, $MoFGBML_{ML}$ showed good generalization ability on MOP1. Since, in MoFGBML, the complexity minimization is often biased with dominance-based EMOA, the number of fuzzy rules tends to be reduced. Thus, there exists a room for searching classifiers with many fuzzy rules and improving classification performance.

In this paper, we selected the best classifier from the non-dominated ones as a representative output, which has the best performance metric for the training dataset. However, from the results, the selected classifier did not always have the best for the test dataset. This means that it is important to select an appropriate complexity to improve generalization ability.

It is also observed that the *CF-mean* method often outperforms the *CF-vector* method for the test dataset in Section V. This is a counterintuitive result of the characteristics of $MoFGBML_{ML}$ in Section IV. In the *CF-vector* method, a fuzzy rule-based classifier classifies an instance by several fuzzy rules for each class. It must be a better strategy for reasoning an unknown class label power set. However, since it can be regarded as classifying each class independently, this method may lack the correlation among classes. Therefore, a fuzzy rule-based classifier based on the *CF-mean* method showed good performance for the real-world datasets.

VI. Conclusion

We proposed a new method adaptation algorithm for multi-label classification, which is named $MoFGBML_{ML}$. It can obtain a number of non-dominated fuzzy rule-based classifiers with different tradeoffs between accuracy and complexity. We also proposed two fuzzy rule-based classification methods, the *CF-mean* method and the *CF-vector* method. We investigated the classification characteristics of our proposed method using two synthetic multi-label datasets. A fuzzy classifier with the *CF-vector* method has the generalization ability for unlearned subsets of class labels even if it does not include the subsets in its consequent class-vectors.

In the computational experiments, we compared our proposed method with two data transformation algorithms applied to MoFGBML and three explainable method adaptation algorithms. We used six real-world multi-label datasets from the Mulan repository and showed the advantages of our proposed method. As a result, $MoFGBML_{ML}$ with the *CF-mean* method showed its high generalization ability. Although other algorithms obtained more complex classifiers, our proposed methods tended to reduce the number of fuzzy rules because it considers the accuracy-complexity tradeoff.

As future research topics, we will examine the effects of other evolutionary multiobjective optimization algorithms in $MoFGBML_{ML}$ because NSGA-II tends to bias to minimizing the complexity in our FGBML algorithm. In addition, since the ratio of the class label power set is often imbalanced in the multi-label dataset, we will also consider a better algorithm dealing with the imbalance of the class label power set.

References

[1] F. Herrera, F. Charte, A. J. Rivera, and M. J. del Jesus, *Multilabel Classification*, Springer, 2016.

[2] N. Spolaôr and G. Tsoumakas, "Evaluating feature selection methods for multi-label text classification," in *BioASQ Workshop*, vol. 1094, 2013.

[3] A. Wieczorkowska, P. Synak, and Z. Raś, "Multi-label classification of emotions in music," in *Intelligent Information Processing and Web Mining*, Springer, vol. 35, pp. 307-315, 2006.

[4] V. F. López, F. de La Prieta, M. Ogihara, and D. D. Wong, "A model for multi-label classification and ranking of learning objects," *Expert Systems with Applications*, vol. 39, no. 10, pp. 8878-8884, 2012.

[5] Y. Xia *et al*., "Weakly supervised multilabel clustering and its applications in computer vision," *IEEE Trans. on Cybernetics*, vol. 46, no. 12, pp. 3220-3232, 2016.

[6] G. P. Mohana and S. Chitra, "Design and development of an efficient hierarchical approach for multi-label protein function prediction," *Biomedical Research Health Science and Bio Convergence Technology: Edition-II*, pp. 370-379, 2017.

[7] A. Clare and R. D. King, "Knowledge discovery in multi-label phenotype data," in *Lecture Notes in Artificial Intelligence 3056*, pp. 1757-1771, 2004.

[8] R. C. Prati, F. Charte, and F. Herrera, "A first approach towards a fuzzy decision tree for multilabel classification," *Proc. of 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1-6, 2017.

[9] S. Nazmi, X. Yan, and A. Homaifar, "Multi-label classification using genetic-based machine learning," *Proc. of 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 675-680, 2018.

[10] Y. Klein, M. Rapp, and E. L. Mencía, "Efficient discovery of expressive multi-label rules using relaxed pruning," *Discovery Science*, Springer, Cham, pp. 367-382, 2019.

[11] C. Panigutti, R. Guidotti, A. Monreale, and D. Pedreschi, "Explaining multi-label black-box classifiers for health applications," *Precision Health and Medicine*, Springer, Cham, pp. 97-110, 2019.

[12] R. C. Prati, "Fuzzy rule classifiers for multi-label classification," *Proc. of 2015 IEEE International Conference on Fuzzy Systems*, pp. 1-8, 2015.

[13] H. Ishibuchi and Y. Nojima, "Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning," *International Journal of Approximate Reasoning*, vol. 44, no. 1, pp. 4-31, January 2007.

[14] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, California, 1993.

[15] M. Umano, H. Okamoto, I. Hatono, H. Tamura, F. Kawachi, S. Umedzu, and J. Kinoshita, "Fuzzy decision trees by fuzzy ID3 algorithm and its application to diagnosis systems," *Proc. of 1994 IEEE third International Conference on Fuzzy Systems*, vol. 3, pp. 2113-2118, 1994.

[16] M. Fazzolari, R. Alcala, Y. Nojima, H. Ishibuchi, and F. Herrera, "A review of the application of multiobjective evolutionary fuzzy systems: current status and future directions," *IEEE Trans. on Fuzzy Systems*, vol. 21, no. 1, pp. 45-65, February 2013.

[17] H. Ishibuchi and Y. Nojima "Multiobjective genetic fuzzy systems," In J. Kacprzyk and W. Pedrycz (eds.) *Springer Handbook of Computational Intelligence*, Springer-Verlag Berlin Heidelberg, pp. 1479-1498 (Chapter 77), May 2015.

[18] A. Fernandez, F. Herrera, O. Cordon, M. J. del Jesus and F. Marcelloni, "Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to?," in *IEEE Computational Intelligence Magazine*, vol. 14, no. 1, pp. 69-81, Feb. 2019.

[19] H. Ishibuchi, T. Nakashima, and M. Nii, *Classification and Modeling with Linguistic Information Granules*, Springer-Verlag Berlin, Heidelberg, 2004.

[20] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, 2002.

[21] G. Tsoumakas, E. Spyromitos-Xioufis, J. Vilcek, and I. Vlahavas, "Mulan: A java library for multi-label learning," *Journal of Machine Learning Research*, vol. 12, pp. 2411-2414, 2011.