IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 3, 2020, accepted February 14, 2020, date of publication February 27, 2020, date of current version March 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2976851

A Multi-Label Classification With Hybrid Label-Based Meta-Learning Method in Internet of Things

SUNG-CHIANG LIN[1], CHIH-JOU CHEN[2], AND TSUNG-JU LEE[3]

[1]Department of Mathematics and Information Education, National Taipei University of Education, Taipei 10671, Taiwan
[2]Department of Marketing and Logistics Management, National Penghu University of Science and Technology, Penghu 88046, Taiwan
[3]Department of Finance, Feng Chia University, Taichung 40724, Taiwan

Corresponding author: Tsung-Ju Lee (freeman.tjlee@mail.fcu.edu.tw)

This work was supported by the Ministry of Science and Technology of Taiwan under Grant NSC 101-2221-E-346-010, Grant MOST 106-2221-E-152-003, and Grant MOST 107-2221-E-152-002-MY2.

ABSTRACT With the widespread adoption of Internet connected devices and the application of Internet of Things (IoT), more and more research efforts focusing on using machine learning techniques in recognizing activities from IoT sensors, especially in solving multi-label classification problems. Without considering the associations among labels, traditional approaches aim to transform the original multi-label classification problem into several single-label classification problems. The loss of information among labels will damage the classification performance. In this paper, we proposed a novel hybrid label-based meta-learning algorithm for multi-label classification based on an ensemble of a cluster algorithm and generalized linear mixed model (GLMM). In this algorithm, the clustering phase is performed to catch the association among labels and to reduce the computational complexity from vast label subsets simultaneously, and the GLMM phase is performed to solve dependence of a subject with multi-labels in training data. The numerical results show that the proposed algorithm outperforms others, especially for cases with relatively large number of labels.

INDEX TERMS Clustering, generalized linear mixed model (GLMM), meta-learning, multi-label classification.

I. INTRODUCTION

The Internet of Things (IoT) is a link and exchange network of messages formed by physical objects, such as vehicles, machines, household appliances, etc.. Moreover, IoT relies on a large number of technologies, such as the application development interface (API), that connects devices to the Internet [1]. The main goal of the IoT is to make it more intelligent by providing information to the surrounding environment, which requires automated data and historical data to be intelligently calculated to automatically make informed decisions [2]. The Internet of Things is widely used, as long as the object can collect and provide the device management platform with the monitored signals or information through sensors, and then feedback the specific decision and indicate the original object to further action. For instance, smart home, wisdom door locks, smart refrigerators, smart cars, smart cities, etc., the core of these wisdom or intelligence relies on predictive analysis based on machine learning. Therefore, how to deal with the large amount of data collected from the sensor and analyze the pattern to become information is an important research topic. In other words, the IoT platform should provide a data analysis system with machine learning techniques to help various applications analyze sensor data to find relevance and make the best response.

Therefore, in recent years, more and more researchers are investing in the development of new machine learning algorithms. In general, the main types of machine learning methods can be classified into supervised learning and unsupervised learning. Traditionally, supervised learning analyzes the data with the label Y and the feature vector X and uses its attributes to build up the classification model with the highest classification accuracy. This model is called a classifier and uses this classifier to predict labels of new observations. Supervised learning is often applied to handwriting identification, medical diagnosis, label processing of speech or text, etc.. Therefore, many important technologies and methods

The associate editor coordinating the review of this manuscript and approving it for publication was Chun-Wei Tsai.

have been developed, such as neural networks, decision trees, Bayesian learning and support vector machines.

In traditional multi-class classification problems, each subject only can be associated with single label, however, in many classification problems, many subjects may belong to multiple labels at the same time. For example, an image might belong to both the blue sky and the beach labels. This is called the multi-label classification problem. In other words, in multi-label classification problem, each subject can be associated with multiple labels. Take document classification for example, each document can be assigned to more than one class. Nowadays, many algorithms, such as neural networks, decision trees, k-nearest neighbors, kernel methods, and ensemble methods, have been proposed to investigate multi-label data including text classification [4], [5], image/video annotation [6], [7], bioinformatics/protein function classification [8], music categorization [9], and directed marketing. In IoT applications, when classifying Activities of Daily Living (ADLs) of smart home inhabitants, it is likely that multiple inhabitants are living in the same home, the multi-class or multi-label classification is needed [3].

A. RELATED WORK ON MULTI-LABEL LEARNING

The multi-label classification algorithms can be divided into two different types, algorithm adaption methods and problem transformation methods (algorithm independent methods) [10], [11]. The first one is algorithm dependent approaches which extend specific learning algorithms to solve the multi-label classification problems directly, such as decision tree [12], support vector machine [13], neural network [5] and boosting learning algorithms [14]. For instance, The decision tree method in [12] is modified from C4.5 algorithm to classify genes based on their functions. The tree nodes are defined by the entropy of C4.5 algorithm and the formula is modified in order to solve multi-label problems.

The second one is algorithm independent methods by transforming the original multi-label classification problem into one or more single-label classification problems. This transformation can be divided into label-based and instance-based transformation. In the label-based transformation, each label is predicted through an independent binary classifier $h_\lambda : X \rightarrow \{0, 1\}$ for $\lambda \in \mathcal{L}$, where X is a design matrix and $\mathcal{L}$ is a finite set of class labels; hence, there are N single-label classifiers used as the number of original classes is equal to N. Each classifier is linked to one of classes and trained by one-against-rest/one-against-all technique, which marks the subjects from i^{th} class as positive and subjects from the remaining classes as negative; hence, it is just like to solve binary classification problems. Then, for the prediction of a new subject x, each classifier independently provides a predicted value, such as a probability or a real number, to determine whether λ is associated to x ($h_\lambda(x) = 1$) or not ($h_\lambda(x) = 0$). This approach is referred to the Binary Relevance (BR) learning, which is theoretically simple and has a linear complexity with respect to the number of labels [15]. However, BR learning assumes that the labels are independent resulting in ignorance of the possible correlations among labels [16], [19] and dependence of a subject with more than one labels in training data. In contrast to the label-based transformation, the instance-based transformation produces both binary and multi-class classification problems by converting multi-label into different subset L, $L \in 2^{|\mathcal{L}|}$, as a distinct single label. For example, some techniques convert the original multi-label problem into single-label problems by elimination of multi-label instances, creation of new single-labels and decomposition of a multi-label problem into a set of single-label problems. This approach is referred to the label powerset (LP). The advantage of LP learning is considering correlations among labels because it contains all possible label subsets, but the drawback of this approach is computationally expensive because the label subsets may potentially be quite large. Therefore, random k-label-sets (RAKEL) approach is proposed in [20]. RAKEL transforms multi-label classification to many multi-class classification by iteratively constructing an ensemble of LP classifiers which is trained by different small random subsets of labels. In order to obtain better performance, the RAKEL approach needs to be tuned many parameters (such as subset size, number of models, and threshold). However, it is hard to find optimal parameters, as the number of training samples is insufficient [21].

B. STRATEGIES FOR MULTI-LABEL CLASSIFICATION

In this paper, we focus on the second type, problem transformation methods, and propose a novel hybrid label-based learning algorithm for multi-label classification, which is based on an ensemble of a clustering algorithm and generalized linear mixed model (GLMM) [22]. This approach overcome some limitations of existing label-based methods, since the association among labels and dependence of a subject with more than one labels in training data are considered and solved through clustering algorithm and GLMM. In addition, the proposed hybrid method will avoids the large number of label subsets to reduce the computational complexity by clustering labels in advance. In contrast to the RAKEL approach, there is no tuning parameter in the proposed hybrid meta-learning algorithm. To understand the effectiveness of the proposed hybrid method, we compare the performance of the proposed method with the notable instance-based learning method, the multi-label k-nearest neighbor (MLKNN) method, which was proposed in [23]. Furthermore, we also compare the proposed method with a problem transformation method, binary relevance method (BR) [15], and two algorithm adaption methods, multi-label ferns (Ferns) [36] and randomForestSRC [37]. The empirical results show that the proposed hybrid method is very promising in terms of the rank loss and average precision.

In the following section, we state the proposed algorithm and the details of each component step first, and explain the related works of used statistical methods. Subsequently, empirical results based on some benchmark data sets are reported. Finally, a brief conclusion remark and discussion follows.

II. THE PROPOSED METHOD

Assume a subject i have features $\mathbf{X}_i = \{x_{i1}, \ldots, x_{ip}\}$ where $i = 1, \ldots, n$, and labels $\mathbf{Y}_i = \{y_{i1}, \ldots, y_{il}\}$, where $y_{ik} = 1$ if subject i belongs to the label λ_k; otherwise $y_{ik} = 0$, and λ_k is an element of a finite set of class labels $\mathcal{L} = \{\lambda_1, \ldots, \lambda_l\}$.

A. HYBRID META-LEARNING

Meta learning is a technique for assembling many (sub-) classifiers to improve the performance of component classifiers [24], [25] and also can be defined as a learning algorithm which is applied to the "learned meta-knowledge" [26]. In others words, the meta-learning approach is a very useful technique which computes a number of (sub-)classifiers first and integrates these (sub-)classifiers to boost overall performance [25], [27]–[29]. Some approaches have been proposed to solve the multi-label problems by applying the meta-learning techniques [20], [21], [30]. Therefore, the meta-learning algorithm includes at least two phases: (1) meta-learner construction and (2) process of ensemble.

The proposed "Hybrid label-based meta-learning with generalized linear mixed model" (HybridLBGLM) approach first cluster the multi-labels for finding implicit dependency among labels. Then, it constructs two types of meta-learner, between-group and within-group meta-learners. By utilizing the dependency information, between-group meta-learners compute rough discrimination information with whole training data. For subjects with dependent labels, Within-group meta-learners further compute precise discrimination information. Finally, discrimination information from between-group learner and within-group learner are integrated for final decisions. Thus, this HybridLBGLM algorithm includes four important steps. Step I, II and III are in meta-learner construction phase, and Step IV is in process of ensemble phase. The four steps of HybridLBGLM algorithm are as follows.

- I Cluster the multi-labels to reduce the number of labels and to detect the association among labels.
- II Construct between-group sub-classifiers via generalized linear mixed models.
- III Construct within-group sub-classifiers via generalized linear mixed models.
- IV Construct the final ensemble and the predict probabilities of original labels which are the combination of the probabilities of between-group sub-classifiers and the conditional probabilities of within-group sub-classifiers.

B. DETECT LABEL CORRELATIONS WITH CLUSTERING ALGORITHM

Some multi-label problems such as automatic image annotation may cause the challenge of large number of labels and many approaches have been proposed to overcome the difficulty [31]. In this study, in order to reduce the number of labels and detect the association among labels, we apply clustering algorithm to cluster labels and redefine the label for each cluster in advance. Cluster analysis is a fundamental technique to explore the latent pattern of data. Clustering algorithms assign a set of similar objects into groups (or called clusters) such that objects in the same cluster are more homogeneous than objects in different clusters, and clusters are heterogeneous to each other. In general, clustering algorithms can be divided into two major types, the partition and the hierarchical algorithms. Partition algorithms obtain clusters by optimizing the criteria function and work efficiently for large dataset. The most popular partition algorithm is K-means algorithm [32], which partition objects into k clusters by minimizing within-cluster sum of squares. However, the K-means algorithm only works on continuous data. For categorical data, K-modes algorithm, which uses modes to represent cluster centers rather than means, is suitable method to obtain clusters [33]. In contrast to partition algorithms, hierarchical algorithms build hierarchy of clusters as a tree structure, called dendrogram and are useful in explaining subgroup structures by visualization. In this paper, in order to deal with large dataset, we adopted K-modes algorithm to cluster labels and detect the pattern of labels.

Let $\mathcal{C} = \{C_1, \ldots, C_{l'}\}$ be new label clusters obtaining from K-modes algorithm, and each original label λ_k only belongs to one new label cluster C_j, where $l' < l$. Then, original label λ_k can be redefined as m_{jr} if $\lambda_k \in C_j$, where j is the index of new label clusters, $j = 1, \ldots, l'$, r is the index of labels within the cluster j, $r = 1, \ldots, l'_j$, l'_j is the number of labels in the label cluster C_j and $\sum_{j=1}^{l'} l'_j = l$. A subject i has a between-group data vector $(\mathbf{X}_i, \mathbf{Y^*}_i)$, where the between-group label vector $\mathbf{Y}^*_i = \{y^*_{i1}, \ldots, y^*_{il'}\}$ and $y^*_{ij} = 1$ if subject i belongs to any one label in C_j, and a within-group data vector $(\mathbf{X}_i, \mathbf{Y'}_i)$, where the within-group label vector $\mathbf{Y'}_i = \{y'_{i11}, \ldots, y'_{il'l'_{l'}}\}$, and $y'_{ijr} = 1$ if subject i belongs to the label m_{jr}; otherwise $y'_{ijr} = 0$.

C. BETWEEN-GROUP META-LEARNING WITH GLMM

We used generalized linear mixed model (GLMM) to establish sub-classifiers. Generalized linear mixed models are widely used to analyze dependent data such as repeated measurement data, longitudinal data and clustered data. GLMM can be viewed as an extension of generalized linear models (GLMs) which represent a class of fixed effects linear models for several types of response variables via different link functions. Except for fixed effect, GLMMs include random effects to consider correlations within subjects in repeated measurement data and longitudinal data, or to capture correlation structures between subjects in clustered data. The model is as follows

$$g(E(\mathbf{Y}|\gamma)) = \mathbf{X}\beta + \mathbf{Z}\gamma, \tag{1}$$

where $g(\cdot)$ is a link function, $\mathbf{X}$ are predictors corresponding to fixed effects β, and $\mathbf{Z}$ are predictors corresponding to random effects γ following some distribution.

References [34] and [22] are some textbooks with details of statistical material on GLMMs.

We trained between-group sub-classifiers via one against rest technique and new label clusters rather than original labels. To set up sub-classifiers, we chose multinomial logit link to predict probability of $\mathcal{C}$ and used random effect to construct correlation for repeated subjects who have multi-labels; that is,

$$\log \frac{Pr(Y^*_{ij}=1)}{Pr(\bigcup_{j'<j} Y^*_{ij'}=1)} = \mathbf{X_i}\beta^*_j + \mathbf{Z_i}\gamma^*_j, \qquad (2)$$

for $j = 2, \ldots, l'$. Then, we can obtain the probabilities for each new label cluster

$$s_j = Pr(Y^*_j = 1) = Pr(C_j) \qquad (3)$$

from between-group meta-learning, where $j = 1, \ldots, l'$.

D. WITHIN-GROUP META-LEARNING WITH GLMM

Subsequently, for each new label cluster C_k, we trained within-group sub-classifier based on one against rest technique and original labels belonging to the new label cluster C_k. As between-group meta-learning, we used GLMM with multinomial logit link to predict probability of λ; that is,

$$\log \frac{Pr(Y'_{ijr}=1|C_j)}{Pr(\bigcup_{r'<r} Y'_{ijr'}=1|C_j)} = \mathbf{X_i}\beta_{jr} + \mathbf{Z_i}\gamma_{jr}, \qquad (4)$$

for $j = 1, \ldots, l'$ and $r = 2, \ldots, l'_j$. Then, we can obtain the probabilities for each label given the label cluster

$$q_{jr} = Pr(Y'_{jr} = 1|C_j) = Pr(m_{jr}|C_j) \qquad (5)$$

from within-group meta-learning, where $j = 1, \ldots, l'$ and $r = 1, \ldots, l'_j$.

E. META-LEARNERS ASSEMBLING

Finally, we assembled between-group sub-classifiers and within-group sub-classifiers to establish the hybrid meta-learner. For each original label, the predicted probability is combination of the probabilities of between-group sub-classifiers and conditional probabilities of within-group sub-classifier; that is,

$$p_k = Pr(\lambda_k) = Pr(m_{jr}|C_j)Pr(C_j) = q_{jr}s_j \qquad (6)$$

Fig. 1 illustrates the basic idea of hybrid label-based meta-learning method with generalized linear mixed model. Algorithm 1 shows the details of HybbridLBGLM algorithm.

Algorithm 1 HybridLBGLM Algorithm

Input: Training Subjects $\{\mathbf{X}_i\}$, Training Labels $\{\mathbf{Y}_i\}$, Testing Subjects $\{\mathbf{X}^s_i\}$, Cluster Number l'

Output: Testing Labels $\{\mathbf{Y}^s_i\}$

1: Use K-modes algorithm to group original labels $\mathbf{Y}$ into l' new label clusters $\mathcal{C} = \{C_1, \ldots, C_{l'}\}$

2: Calculate the between-group label vector $\{\mathbf{Y}^*_i\}$ for each subject, where $\mathbf{Y}^*_i = \{y^*_{i1}, \ldots, y^*_{il'}\}$

3: Calculate the within-group label vector $\{\mathbf{Y'}_i\}$ for each subject, where $\mathbf{Y'}_i = \{y'_{i11}, \ldots, y'_{il'l'_{l'}}\}$

4: Train the between-group meta-learner by GLMM with multinomial logit link

$$\log \frac{Pr(Y^*_{ij}=1)}{Pr(\bigcup_{j'<j} Y^*_{ij'}=1)} = \mathbf{X_i}\beta^*_j + \mathbf{Z_i}\gamma^*_j,$$

for $j = 2, \ldots, l'$ and calculate the probabilities for each label cluster $s_{ij} = Pr(Y^*_{ij} = 1)$, where $j = 1, \ldots, l'$.

5: Train the within-group meta-learner by GLMM with multinomial logit link

$$\log \frac{Pr(Y'_{ijr}=1|C_j)}{Pr(\bigcup_{r'<r} Y'_{ijr'}=1|C_j)} = \mathbf{X_i}\beta_{jr} + \mathbf{Z_i}\gamma_{jr},$$

for $j = 1, \ldots, l'$, $r = 2, \ldots, l'_j$ and calculate the probabilities for each label given the label cluster $q_{ijr} = Pr(Y'_{ijr} = 1|C_j) = Pr(m_{ijr}|C_j)$, where $j = 1, \ldots, l'$ and $r = 1, \ldots, l'_j$.

6: Calculate the probability of the original label λ_k for each subject i by assembling between-group learners and within-group learners

$$p_{ik} = Pr(Y_{ik} = 1) = Pr(m_{ijr}|C_j)Pr(Y^*_{ij} = 1) = q_{ijr}s_{ij}$$

7: **return** $\mathbf{P}_i = \{p_{i1}, \ldots, p_{il}\}$

III. EXPERIMENTS

In this section, we used the benchmark data from a Java Library for multi-label Learning [38] to demonstrate the performance of the proposed algorithm. We analyzed five datasets from several different application domains such as biological, music, video and image. The CAL500 and emotions are music data sets. The CAL500 collects 700 human-generated musical annotations including 500 popular western musical tracks. It includes 68 features and 174 labels. The emotions data includes 593 piece of songs created from 233 musical albums and the duration of each song was 30 seconds. It contains 72 features and six main emotional clusters, amazed-surprised, happy-pleased, relaxing-clam, quiet-still, sad-lonely and angry-aggressive, which are corresponded with the Tellegen-Watson-Clark model of mood in the emotion labeling process. The mediamill data set is multimedia data and contains pre-computed low-level multimedia features from the 85 hours of international broadcast news data belonging to the TRECVID 2005/2006 benchmark. It includes 978 instances, 1449 features and 45 labels. The image dataset scene is semantic index of still scenes. In the scene data, there are six classes including beach, sunset, foliage, field, mountain, and urban, 2047 instances and 294 features. The biological dataset yeast is used to predict the functional classes of genes. The data set contains 2417 genes grouped into 14 functional classes, and each gene is described by a 103-dimensional feature vector. These multi-label datasets

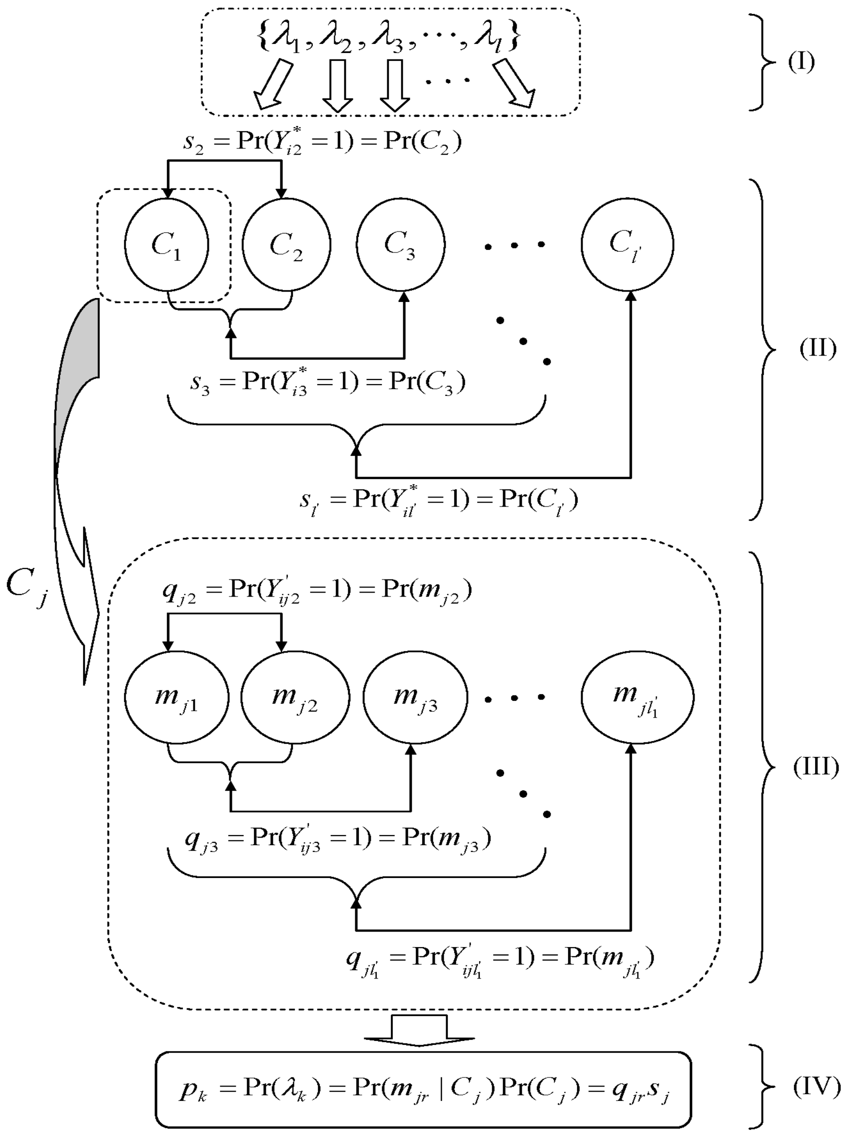

FIGURE 1. **Procedure for Constructing Hybrid label-based meta-learning method with generalized linear mixed model.**

TABLE 1. **Standard statistics for the multi-label data sets used in the experiments.**

Data set	Domain	#Instances	#Attributes	#Labels	cardinality*
CAL500	music	502	68	174	26.044
emotions	music	593	72	6	1.869
mediamill	video	43907	120	101	4.376
scene	image	2407	294	6	1.074
yeast	biology	2417	103	14	4.237

*The label cardinality denotes the average number of labels per instance

are all in Weka's ARFF format and the details are described in Table 1.

A. EVALUATION MEASUREMENTS

The performance measures of multi-label classification are different from single-label classification. In single-label classification, accuracy or F-measure are usually used to evaluate the performance of classifiers. However, in contrast to single-label classification, performance evaluation in multi-label classification is more complicated. The multi-label classification aims to build models for subjects assigned with multiple labels simultaneously. In multi-label classification, the results of classification are partially correct or partially wrong; hence, an accuracy dose not work at all and there are some measures proposed to evaluate the performance of multi-label classification algorithms [10], [16].

Many performance measures focusing on different aspects were proposed, such as Hamming loss, ranking loss, one-error, average precision, coverage [14], micro-F1 and macro-F1 [17]. Wu & Zhou (2017) show that by maximizing instance-wise margin, macro-AUC, macro-F1 and

TABLE 2. Mean (standard deviation) for testing results of MLKNN with different tuning parameters of k.

Data set	Testing Result	MLKNN				
		k=8	k=9	k=10	k=11	k=12
Emotions	Rank Loss	0.251 (0.032)	0.257 (0.034)	0.258 (0.033)	0.257 (0.034)	0.253 (0.035)
	One Error	0.380 (0.059)	0.381 (0.059)	0.385 (0.060)	0.377 (0.062)	0.369 (0.061)
	Coverage	2.219 (0.167)	2.267 (0.182)	2.244 (0.172)	2.266 (0.182)	2.247 (0.189)
	Hamming Loss	0.256 (0.021)	0.259 (0.021)	0.261 (0.019)	0.262 (0.021)	0.259 (0.020)
	Average Precision	0.715 (0.032)	0.714 (0.034)	0.713 (0.034)	0.716 (0.034)	0.719 (0.035)
Scene	Rank Loss	0.079 (0.010)	0.078 (0.009)	0.077 (0.010)	0.076 (0.010)	0.076 (0.010)
	One Error	0.226 (0.025)	0.225 (0.021)	0.223 (0.026)	0.225 (0.026)	0.226 (0.027)
	Coverage	0.482 (0.053)	0.476 (0.05)	0.470(0.053)	0.470 (0.057)	0.468 (0.056)
	Hamming Loss	0.087 (0.008)	0.087 (0.007)	0.085 (0.008)	0.086 (0.008)	0.086 (0.008)
	Average Precision	0.865 (0.015)	0.865 (0.013)	**0.867 (0.015)**	0.866 (0.015)	0.866 (0.015)
Yeast	Rank Loss	0.170 (0.011)	0.169 (0.011)	0.166 (0.012)	0.167 (0.012)	0.168 (0.011)
	One Error	0.231 (0.024)	0.232 (0.026)	0.227 (0.026)	0.230 (0.026)	0.232 (0.028)
	Coverage	6.323 (0.192)	6.302 (0.177)	6.281 (0.22)	6.259 (0.203)	6.288 (0.188)
	Hamming Loss	0.196 (0.008)	0.195 (0.008)	0.194 (0.01)	0.193 (0.009)	0.195 (0.008)
	Average Precision	0.762 (0.015)	0.762 (0.015)	**0.766 (0.016)**	0.765 (0.017)	0.763 (0.016)

Hamming loss are to be optimized. The other performance measures (ranking loss, one-error, coverage, average precision, instance-F1, micro-F1, instance-AUC, micro-AUC) except micro-AUC are to be optimized under maximizing label-wise margin [18]. Besides, the most commonly used performance measures of multi-label classification are ranking loss, one-error, coverage, hamming loss, and average precision.

For a classifier h and a subject with features x, let $h(x) \subseteq \mathcal{L}$ be multi-label prediction of x, and let L_x be the true set of relevant labels. Besides, f denotes the score function for the label $\mathcal{L}$. The measures of performance for multi-label classifications are briefly described as follows.

* **Rank loss** computes the average of label pairs that are not correctly ordered.

$$RankLoss(f) = \frac{\#\{(\lambda,\lambda')|f(x,\lambda) \leq f(x,\lambda'), (\lambda,\lambda') \in L_x \times \overline{L_x}\}}{|L_x||\overline{L_x}|} \tag{7}$$

* **One error** computes how many times the top-ranked label is not relevant.

$$OneError(f) = \begin{cases} 1 & \text{if } \mathrm{argmax}_{\lambda\in\mathcal{L}} f(x,\lambda) \notin L_x, \\ 0 & \text{otherwise} \end{cases}$$

* **Coverage** computes how far one needs to go in the list of labels to cover all the relevant labels of an instance. This measure is loosely related to the precision at the level of perfect recall.

$$Coverage(f) = \max_{\lambda\in\mathcal{L}_x} rank_f(x,\lambda) - 1 \tag{8}$$

* **Hamming loss** computes the percentage of labels whose relevance is predicted incorrectly.

$$HammingLoss(h) = \frac{1}{|\mathcal{L}|}|h(x)\Delta L_x| \tag{9}$$

where Δ is the symmetric difference between two sets.

* **Average precision** computes the percentage of relevant labels $\lambda \in L_x$ among all labels that are ranked above it and average these percentages over all relevant labels.

$$AvePrec(f) = \frac{1}{|L_x|}\sum_{\lambda\in L_x} \frac{|\{\lambda'|rank_f(x,\lambda') \leq rank_f(x,\lambda), \lambda' \in L_x\}|}{rank_f(x,\lambda)} \tag{10}$$

B. RESULTS

We compare the hybrid label-based meta-learning with generalized linear mixed model (HybridLBGLM) approach with the MLKNN [23], binary relevance method (BR) [15], [35], multi-label ferns (Ferns) [36] and randomForestSRC [37]. MLKNN is the arguably the state-of-the-art in instance-based multi-label ranking. We used the MLKNN implementation in MULAN package [38] and tried some different size of nearest neighbors, k, which is an important parameter in MLKNN approach. The nearest neighborhood size setting (k) for MLKNN is from 8 to 12. The best k, 10, is selected for comparing with other methods. Table 2 shows the results of MLKNN with different k settings. Binary relevance method, which is the simplest problem transformation method, learns a binary classifier for each label and then combined all binary classifiers to a multi-label target. We used mlr package [35] to run BR, Ferns and randomForestSRC. Table 3 and 4 summarize the testing results of our experiments. All the empirical results are based on ten-fold cross-validation and 300 times repetitions. Looking at the Table 3 and Table 4, it can be seen that the HybridLBGLM is the best one in terms of rank loss and average precision measures.

Table 3 shows the results for the multi-label data sets with relative smaller number of labels. It is obvious to find that HybridLBGLM approach performs best according to rank loss, coverage and average precision, because this method consider correlations among labels. The BR is the worst method for multi-label classification. The Fern has lowest one error, but has lower average precision. Because it may distinguish top-ranked label to obtain higher accuracy for

TABLE 3. Mean (standard deviation) for testing results of HybridLBGLM with relative smaller number of labels.

Data set	Testing Result	HybridLBGLM	BR	Ferns	randomForestSRC	MLKNN(k=10)
Emotions	Rank Loss	**0.160(0.021)**	0.499(0.046)	0.377(0.036)	0.436(0.047)	0.258 (0.033)
	One Error	0.271(0.042)	0.169(0.049)	**0.073(0.033)**	0.116(0.040)	0.385 (0.060)
	Coverage	**1.799(0.158)**	2.501(0.233)	2.090(0.199)	2.399(0.228)	2.244 (0.172)
	Hamming Loss	0.224(0.018)	0.240(0.022)	0.248(0.024)	**0.181(0.019)**	0.261 (0.019)
	Average Precision	**0.803(0.022)**	0.705(0.034)	0.726(0.035)	0.751(0.033)	0.713 (0.034)
Scene	Rank Loss	**0.076(0.008)**	0.449(0.032)	0.264(0.016)	0.437(0.030)	0.077 (0.010)
	One Error	0.222(0.015)	0.129(0.021)	0.058(0.014)	**0.048(0.014)**	0.223 (0.026)
	Coverage	0.498(0.044)	1.130(0.104)	0.820(0.068)	1.003(0.106)	**0.470(0.053)**
	Hamming Loss	0.103(0.006)	0.121(0.008)	0.183(0.009)	0.089(0.007)	**0.085 (0.008)**
	Average Precision	**0.868(0.008)**	0.734(0.020)	0.734(0.018)	0.780(0.020)	0.867 (0.015)
Yeast	Rank Loss	**0.165(0.008)**	0.483(0.019)	0.651(0.016)	0.479(0.019)	0.166 (0.012)
	One Error	0.226(0.025)	0.110(0.020)	**0.057(0.015)**	0.139(0.022)	0.227 (0.026)
	Coverage	**6.208(0.199)**	9.619(0.239)	9.441(0.194)	9.149(0.244)	6.281 (0.22)
	Hamming Loss	0.196(0.008)	0.214(0.008)	0.458(0.013)	**0.190(0.008)**	0.194 (0.01)
	Average Precision	**0.767(0.009)**	0.653(0.017)	0.411(0.011)	0.658(0.016)	0.766 (0.016)

TABLE 4. Mean (standard deviation) for testing results of HybridLBGLM with large number of labels.

Data set	Testing Result	HybridLBGLM			MLKNN			BR	Ferns	randomForestSRC
		$l' = 15$	$l' = 20$	$l' = 25$	k=8	k=10	k=12			
CAL500	Rank Loss	0.183(0.007)	**0.178(0.006)**	0.180(0.005)	0.185(0.007)	0.186(0.006)	0.186(0.008)	0.704(0.014)	0.896(0.006)	0.772(0.009)
	One Error	0.152(0.051)	0.143(0.046)	0.146(0.059)	0.144(0.046)	0.150(0.049)	0.149(0.045)	**0.000(0.001)**	**0.000(0.000)**	**0.000(0.000)**
	Coverage	131.074(4.177)	**129.434(3.225)**	131.054(5.598)	132.915(3.985)	132.493(3.94)	132.503(3.766)	169.658(0.617)	168.830(0.803)	169.345(0.627)
	Hamming Loss	0.151(0.004)	0.149(0.004)	0.152(0.004)	0.138(0.004)	0.139(0.004)	0.139(0.004)	0.174(0.004)	0.772(0.007)	**0.137(0.004)**
	Average Precision	0.492(0.010)	**0.515(0.009)**	0.501(0.009)	0.493(0.013)	0.490(0.012)	0.489(0.014)	0.270(0.013)	0.114(0.004)	0.294(0.010)
Mediamill	Rank Loss	**0.055(0.007)**	0.056(0.009)	0.057(0.008)	0.058(0.003)	0.057(0.003)	0.056(0.004)	0.538(0.005)	0.549(0.004)	0.444(0.004)
	One Error	0.173(0.014)	0.209(0.014)	0.225(0.015)	0.158(0.015)	0.156(0.016)	0.156(0.017)	0.123(0.005)	0.123(0.005)	**0.088(0.004)**
	Coverage	16.721(1.064)	**16.084(0.794)**	17.046(1.239)	19.302(0.845)	19.081(0.892)	19.048(0.832)	56.474(0.395)	53.719(0.427)	50.698(0.505)
	Hamming Loss	0.033(0.002)	0.032(0.001)	0.032(0.001)	0.034(0.002)	0.032(0.001)	0.032(0.001)	0.032(0.000)	0.396(0.002)	**0.026(0.000)**
	Average Precision	**0.698(0.006)**	0.695(0.004)	0.694(0.005)	0.690(0.011)	0.692(0.010)	0.694(0.010)	0.484(0.004)	0.136(0.002)	0.582(0.004)

top-ranked label, but may not have sufficient discrimination for other labels.

Table 4 demonstrates the results for the multi-label data sets with large number of labels. There are more than 100 labels in CAL500 and Mediamill data sets. In the HybridLBGLM approach, l' represents the number of label clusters in the first stage of the proposed algorithm. We also compare the results with the BR, the Ferns, the randomForest-SRC and the MLKNN as k equals 8,10, and 12. The testing results show that the HybridLBGLM has lower rank loss, lower coverage and higher average precision in CAL500 and Mediamill data sets. The BR, the Ferns and the randomForest-SRC has one error almost equal to zero and worse other performance measures, the possible reason is overfitting in the training sets. In summary, for large number of labels, the HybridLBGLM approach still has competitive advantage. The HybridLBGLM approach performs better as $l' = 20$ for CAL500 data and $l' = 15$ for Mediamill data; nevertheless, the variations among different numbers of label clusters l' are small.

Overall, the methodology of this study has the advantages here. When the number of labels is not large, such as emotions, scene and yeast data sets, the MLKNN method still needs to be adjusted with some different size of nearest neighbors, k, to achieve better performance. The proposed method in this study can obtain better prediction results without tuning parameters. On the other hand, when the number of labels is relative large such as CAL500 and Mediamill datasets, the proposed method in this study will consider the relationship between labels and integrate it into the prediction model to obtain better performance results. Hence, the proposed approach, HybridLBGLM, indeed shows the efficient solution when solve the multi-label classification.

IV. DISCUSSION AND CONCLUSION

We proposed a hybrid label-based learning algorithm which assembles a clustering algorithm and a generalized linear mixed model to deal with multi-label classification problems. It carries out the correlations among labels and reduce the size of labels subset as applying one against rest technique and cluster analysis. Additionally, it uses GLMM to construct the association among labels and dependence of a subject with multi-labels in training data, and finally provide the probabilities of each label through the hybrid meta-learning ensemble to obtain better predictions. In other words, the advantage of combining cluster algorithm and generalized linear mixed model is that reducing the size of label subsets and considering correlation among labels can be tackled simultaneously.

The first step in HybridLBGLM method is the clustering stage that can reduce the number of labels and detect the possible correlations among labels. The clustering step in HybridLBGLM method plays an important role to solve the problem of label powerset which is computationally expensive when the label subsets are potentially quite large. However, the labels are categorical data and common cluster algorithms such as $k-$means and hierarchical cluster methods based on Euclidean or Mahalanobis distance do not work. Hence, how to choose a suitable cluster algorithm and a suitable similarity/distance measure are important. Moreover, the HybridLBGLM approach deals with the correlations among labels by assembling between-group sub-classifiers and within-group sub-classifiers to establish the hybrid meta-learner. For each original label, the prediction is based on the combination of the probabilities from between-group sub-classifiers and conditional probabilities from within-group sub-classifier.

In addition, the generalized linear mixed model is used to establish sub-classifiers and to deal with the dependence of a subject with multi-labels in training data. The GLMM can be viewed as an extension of generalized linear models which consider correlations within subjects in repeated measurement data and longitudinal data, or to capture correlation structures between subjects in clustered data. By applying generalized linear mixed model, the HybridLBGLM method can capture the correlation among sub-groups and the correlation within a sub-group. This is another key point in HybridLBGLM method to improve performance when applying binary relevance learning.

In the last step, meta-learning assembling, the final predicted probability will be assembled by the probabilities of between-group sub-classifiers and conditional probabilities of within-group sub-classifier. The correlation among labels will be handled again. Thus, this HybridLBGLM approach can overcome the disadvantage of traditional BR learning which ignore the correlation among labels and dependence of subject with multi-labels; as well as, HybridLBGLM also improves the drawback of LP learning which contains potentially large number of label subsets and computational complexity.

ACKNOWLEDGMENT

The authors would like to thank Dr. C. Wang for her valuable contribution in statistical consultation.

REFERENCES

[1] C. Cecchinel, M. Jimenez, S. Mosser, and M. Riveill, "An architecture to support the collection of big data in the Internet of Things," in *Proc. IEEE World Congr. Services*, Jun. 2014, pp. 442–449.

[2] F. Alam, R. Mehmood, I. Katib, and A. Albeshri, "Analysis of eight data mining algorithms for smarter Internet of Things (IoT)," *Procedia Comput. Sci.*, vol. 98, pp. 437–442, Jan. 2016.

[3] T. Alshammari, N. Alshammari, M. Sedky, and C. Howard, "Evaluating machine learning techniques for activity classification in smart home environments," *Int. J. Comput. Elect. Automat. Control Inf. Eng*, vol. 12, pp. 48–54, Feb. 2018.

[4] S. Godbole and S. Sarawagi, "Discriminative methods for multi-labeled classification," in *Proc. PAKDD*, in Lecture Notes in Computer Science, vol. 3056, H. Dai, R. Srikant, and C. Zhang, Eds. 2004, pp. 22–30.

[5] M.-L. Zhang and Z.-H. Zhou, "Multilabel neural networks with applications to functional genomics and text categorization," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 10, pp. 1338–1351, Oct. 2006.

[6] M. A. Tahir, J. Kittler, F. Yan, and K. Mikolajczyk, "Kernel discriminant analysis using triangular kernel for semantic scene classification," in *Proc. 7th Int. Workshop Content-Based Multimedia Indexing*, Jun. 2009, pp. 1–6.

[7] A. Dimou, G. Tsoumakas, V. Mezaris, I. Kompatsiaris, and I. Vlahavas, "An empirical study of multi-label learning methods for video annotation," in *Proc. 7th Int. Workshop Content-Based Multimedia Indexing*, Jun. 2009, pp. 19–24.

[8] R. Cerri, R. R. Silva, and C. A. Carvalho, "Comparing methods for multilabel classification of proteins using machine learning techniques," in *Proc. 4th Brazilian Symp. Bioinf. (BSB)*, 2009, pp. 109–120.

[9] T. Li and M. Ogihara, "Toward intelligent music information retrieval," *IEEE Trans. Multimedia*, vol. 8, no. 3, pp. 564–574, Jun. 2006.

[10] A. C. P. L. F. de Carvalho and A. A. Freitas, "A tutorial on multilabel classification techniques," in *Foundations of Computational Intelligence: Function Approximation and Classification*, vol. 5, A. Abraham, A.-E. Hassanien, and V. Snášel, Eds. Berlin, Germany: Springer, 2009, pp. 177–195.

[11] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int. J. Data Warehousing Mining*, vol. 3, pp. 1–13, Jul. 2007.

[12] A. Clare and R. King, "Knowledge discovery in multi-label phenotype data," in *Proc. PKDD*, in Lecture Notes in Computer Science, vol. 2168, A. Siebes and L. De Raedt, Eds. 2001, pp. 42–53.

[13] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 14, 2001, pp. 681–687.

[14] R. E. Schapire and Y. Singer, "BoosTexter: A boosting-based system for text categorization," *Mach. Learn.*, vol. 39, nos. 2–3, pp. 135–168, May 2000.

[15] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Mach. Learn.*, vol. 85, no. 3, pp. 333–359, Jun. 2011.

[16] W. Cheng and E. Hüllermeier, "Combining instance-based learning and logistic regression for multilabel classification," *Mach. Learn.*, vol. 76, nos. 2–3, pp. 211–225, Jul. 2009.

[17] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Random k-labelsets for multilabel classification," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 7, pp. 1079–1089, Jul. 2011.

[18] X.-Z. Wu and Z.-H. Zhou, "A unified view of multi-label performance measures," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 3780–3788.

[19] J. Fürnkranz, E. Hüllermeier, E. L. Mencía, and K. Brinker, "Multilabel classification via calibrated label ranking," *Mach. Learn.*, vol. 73, no. 2, pp. 133–153, Aug. 2008.

[20] G. Tsoumakas and I. Vlahavas, "Random k-labelsets: An ensemble method for multilabel classification," in *Proc. ECML*, vol. 4701, 2007, pp. 406–417.

[21] M. A. Tahir, J. Kittler, K. Mikolajczyk, and F. Yan, "Improving multilabel classification performance by using ensemble of multi-label classifiers," in *Proc. MCS*, 2010, pp. 11–21.

[22] C. E. McCulloch and S. R. Searle, *Generalized, Linear, and Mixed Models*. New York, NY, USA: Wiley, 2001.

[23] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007.

[24] C. Giraud-Carrier, R. Vilalta, and P. Brazdil, "Introduction to the special issue on meta-learning," *Mach. Learn.*, vol. 54, no. 3, pp. 187–193, Mar. 2004.

[25] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*. Berlin, Germany, 2000, pp. 1–15, doi: 10.1007/3-540-45014-9_1.

[26] S. Ali and K. A. Smith-Miles, "A meta-learning approach to automatic kernel selection for support vector machines," *Neurocomputing*, vol. 70, nos. 1–3, pp. 173–186, Dec. 2006.

[27] A. Prodromidis, P. Chan, and S. Stolfo, "Meta-learning in distributed data mining systems: Issues and approaches," *Adv. Distrib. Parallel Knowl. Discovery*, vol. 3, pp. 81–114, 2000.

[28] Y.-C. I. Chang and S. C. Lin, "Synergy of logistic regression and support vector machine in multiple-class classification," in *Proc. IDEAL*, 2004, pp. 132–141.

[29] S.-C. Lin, Y.-C.-I. Chang, and W.-N. Yang, "Meta-learning for imbalanced data and classification ensemble in binary classification," *Neurocomputing*, vol. 73, nos. 1–3, pp. 484–494, Dec. 2009.

[30] N. Cesa-Bianchi, M. Re, and G. Valentini, "Synergy of multi-label hierarchical ensembles, data fusion, and cost-sensitive methods for gene functional inference," *Mach. Learn.*, vol. 88, nos. 1–2, pp. 209–241, Dec. 2011.

[31] G. Nasierding, G. Tsoumakas, and A. Z. Kouzani, "Clustering based multi-label classification for image annotation and retrieval," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2009, pp. 4514–4519.

[32] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.* Berkeley, CA, USA: Univ. of California Press, 1967, pp. 281–297.

[33] Z. Huang, "Extensions to the K-means algorithm for clustering large data sets with categorical values," *Data Mining Knowl. Discovery*, vol. 2, no. 3, pp. 283–304, Sep. 1998.
[34] P. Diggle, P. Heagerty, K.-Y. Liang, and S. L. Zeger, *Analysis of Longitudinal Data*, 2nd ed. New York, NY, USA: Oxford Univ. Press, 2002.
[35] P. Probst, Q. Au, G. Casalicchio, C. Stachl, and B. Bischl, "Multilabel classification with R package MLR," 2017, *arXiv:1703.08991*. [Online]. Available: http://arxiv.org/abs/1703.08991
[36] M. B. Kursa and A. A. Wieczorkowska, "Multi-label ferns for efficient recognition of musical instruments in recordings," in *Proc. Int. Symp. Methodol. Intell. Syst.* Cham, Switzerland: Springer, 2014, pp. 214–223.
[37] H. Ishwaran and U. B. Kogalur. (2016). *RandomForestSRC:Random Forests for Survival, Regression Classification (RF-SRC)*. [Online]. Available: http://cran.r-project.org/package=randomForestSRC
[38] G. Tsoumakas, E. Spyromitros-Xioufis, J. Vilcek, and I. Vlahavas, "Mulan: A java library for multi-label learning," *J. Mach. Learn. Res.*, vol. 12, pp. 2411–2414, Jun. 2011.

SUNG-CHIANG LIN received the B.S. degree in applied mathematics from National Chung Hsing University, Taichung, Taiwan, in 1998, and the M.B.A. and Ph.D. degrees in information management from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 2010. From 2001 to 2008, he was a Research Assistant with the Institute of Statistical Science, Academia Sinica, Taipei. Since 2010, he has been an Assistant Professor with the Information Management Department, National Penghu University of Science and Technology, Penghu, Taiwan. He is currently an Assistant Professor with the Department of Mathematics and Information Education, National Taipei University of Education, Taipei. His research interests include machine learning, data mining, technology-assisted instruction/learning, information technology integration, and customer relationship management (CRM).

CHIH-JOU CHEN received the bachelor's degree from National Chiao Tung University, Taiwan, the master's degree from National Central University, Taiwan, and the Ph.D. degree from the Department of Business Administration, National Central University. He is currently a Professor with the Department of Marketing and Logistics Management, National Penghu University of Science and Technology, Penghu, Taiwan. His current research interests focus on the development of virtual communities, knowledge management, new product innovation, and performance management. His researches have been published in *Computers in Human Behavior*, the *International Journal of Organizational Analysis*, and other leading refereed journals and conference proceedings.

TSUNG-JU LEE received the B.S. degree in mathematics from Tunghai University, Taiwan, in 2000, and the M.S. degree in applied mathematics and the Ph.D. degree in computer science and engineering from National Chiao Tung University, Taiwan, in 2002 and 2013, respectively. He was an Assistant Professor with the Department of Information Management, National Penghu University of Science and Technology. He is currently an Assistant Professor with the Department of Finance, Feng Chia University. His research interests include human computation, machine learning, data mining, and related applications, especially in network security, e-learning, and FinTech. He was a Guest Editor of Special Collection on Resource Recycling and Renewable Energy Development in Energy Exploration and Exploitation, in December 2019.