Selective Classifier Chains Based on Max-relevance and Min-redundancy for Multi-label Classification

Ge Huang Youlong Yang and Jing Bai

***Abstract*—The classifier chains method is one of the most well-known methods for multi-label classification, which can model label correlations while keeping acceptable computational complexity. However, the drawbacks are that potential label redundancies may be overlooked and label relevances have not been specifically measured, making a rough and redundant model. In this paper, we present a new architecture of the classifier chains based on max-relevance and min-redundancy feature selection method, called mRMR-CC, and provide a dynamic learning algorithm of selection labels in the process of classifier chains. The algorithm considers concretely not only label correlations but also label redundancies, which allows us to select a compact additional attributes set for each base classifier. A series of numeric studies are performed using a broad range of multi-label data sets with a variety of evaluation metrics. Extensive experiments show that the proposed selective classifier chains model leads to promising improvement on additional attributes selection of the classifier chains method and predictive performance.**

***Index Terms*—Multi-label classification, Classifier chains, Max-relevance, Min-redundancy.**

I. Introduction

MULTI-LABEL classification [1] is a common classification problem where each instance may be assigned to multiple labels simultaneously. In recent years, the multi-label classification problems have attracted increasing attention of many research fields, such as image annotation, text categorization, semantic scene classification and bioinformatics, among others. For instance, in the scene classification [2], a scene may include sunsets and beaches at the same time.

Multi-label classification methods can be divided into three categories [3], which are problem transformation methods, algorithm adaptation methods and ensemble methods respectively. A widely used method for multi-label classification is to conduct problem transformation that our work mainly concentrates on. In problem transformation methods, a multi-label classification problem is transformed into one or more single-label classification tasks, so that traditional classifier learning algorithm can be used directly to deal with multi-label classification. Two main types of methods have been presented for problem transformation methods: binary relevance method(BR) and label power-set approach(LP). A straightforward approach for problem transformation is binary relevance method(BR)[4], which decomposes a multi-label classification problem into multiple binary classification tasks. Nevertheless, it overlooks the interdependencies between labels. Thus, BR method suffers potential information loss.

Recently, in order to overcome the existing defects in BR method, many papers have introduced the importance of label correlations, including theoretical analyses of label dependence in the context of MLC [5]. In this regard, two types of dependence have been formally distinguished, which are conditional dependence and marginal (unconditional) dependence respectively. In conditional dependence, the classifier chains method(CC) [6], [7] has been proposed for incorporating label interactions based on BR method. It includes d base classifiers linked in a chain, such that each classifier includes the labels predicted by all previous classifiers as additional attributes. The method could model label correlations while maintaining the acceptable computational complexity. However, there are three existing disadvantages for CC method: (1) label relevances have not been specifically measured;(2) label redundancies may be overlooked; (3) it becomes problematic for certain domains because the number of additional attributes increases with the number of labels. In addition to, the order of labels has a strong impact on predictive accuracy and an ensemble of classifier chains (ECC) is used with complicated computation [7], [8], [9]. Besides, a multi-dimension Bayesian network classifier [10], [11], [12] has been proposed for modeling label relevances and defending the combinatorial explosion of power-set approach. However, it suffers from the high computational complexity for learning a complete Bayesian network with numerous class variables.

In fact, the process of extending labels is corresponding to variable selection process, and feature selection is an important problem for pattern classification problem. When it comes to variable selection, it is easy to think of the max-relevance and min-redundancy feature selection algorithm based on mutual information (mRMR) [13]. In this method, mutual information is used to measure the nonlinear dependence, which is widely applied in feature selection and other methods [14], [15], [16], [17], [18]. The mRMR method not only describes the relevance between a candidate variable X_i and class C_i but also considers the redundancy between any pair of candidate variables X_i and X_j. In addition to, to overcome the limitation of previous works, an improvement for the mRMR algorithm has been proposed in [19], which introduces the max-relevance and min-

Manuscript received March 14, 2017; revised July 15, 2017

Ge Huang is with the Department of Mathematics and Statistics, Xidian University, 266 Xinglong Section of Xifeng Road, Xi'an, Shaanxi 710126, China, e-mail: hg8121019@126.com.

Youlong Yang is with the Department of Mathematics and Statistics, Xidian University, 266 Xinglong Section of Xifeng Road, Xi'an, Shaanxi 710126, China, e-mail: ylyang@mail.xidian.edu.cn.

Jing Bai is with the Department of Mathematics and Statistics, Xidian University, 266 Xinglong Section of Xifeng Road, Xi'an, Shaanxi 710126, China.

redundancy feature selection method based on normalized mutual information(NMIFS). Besides, other feature selection methods have been also proposed in some papers [20], [21], [22].

In the paper, we propose a dynamic selective classifier chains method on the basis of max-relevance and min-redundancy algorithm for additional attributes selection, called mRMR-CC method. It is well known that very little work has been reported about the redundancies between additional attributes. Our aim is to select directly a compact additional attributes set from all previous labels for each base classifier, which considers concretely not only the relevances between class labels with additional attributes but also the redundancies between the additional attributes. That is, for every class label C_i, the additional attributes selection is to find an additional attributes set S_{i-1}, which has the maximal dependency with the target class C_i and the minimal redundancy between additional attributes. The method can also effectively reduce the number of additional attributes (label variables) in the process of classifier chains. Moreover, extensive experiments are conducted to show the effectiveness of the proposed method.

Finally, the main contributions of the paper are as follows:

1) for each classifier, consider the redundancy between any pair of additional attributes and reduce the number of additional attributes.
2) measure specifically the dependency between class label with any additional attribute at each classifier.
3) propose a dynamic process of selection labels for the classifier chains model based on mRMR algorithm.
4) conduct numerous experiments and show the effectiveness of the proposed method.

The rest of this paper is organized as follows: Section 2 introduces the multi-label classification and analyzes the previous work. We discuss the max-relevance and min-redundancy feature selection work in Section 3. In section 4, we propose the dynamic selective classifier chains method based on max-relevance and min-redundancy and describe the related algorithm. The results of numerical experiments are summarized in section 5. Finally, our conclusion and future work are given in Sections 6.

II. Multi-label classification

In this section, we briefly review the multi-label classification problem at first. Then we introduce the state-of-the-art methods for multi-label classification that are used in this paper, including the binary relevance method(BR)and the classifier chains method(CC).

A. Multi-label classification

The multi-label classification task [1], [2], [12] is corresponding to searching for a function H, which assigns each instance represented by a vector of n features values $X = (x_1, x_2, ..., x_n)$ of n dimensional features variable $(X_1, X_2, ..., X_n)$ to a vector of d class values $C = (c_1, c_2, ..., c_d)$ of the d dimensional class variable $(C_1, C_2, ..., C_d)$:

$$H : \Omega_{X_1} \times ... \times \Omega_{X_n} \rightarrow \Omega_{C_1} \times ... \times \Omega_{C_d}$$

$$(X_1, X_2, ..., X_n) \rightarrow (C_1, C_2, ..., C_d)$$

here, X_i is the ith feature variable, which could be discrete or continuous, C_j represents the jth class variable that takes value is 0 or 1. And Ω_{X_i} and Ω_{C_j} represent their sample space, respectively. The goal of H function is to assign each instance X to the most likely combination of classes, that is:

$$C^* = argmax_{c_1,...,c_d} P(C_1 = c_1, ..., C_d = c_d|X) \qquad (1)$$

B. Binary relevance method-BR

Recently, numerous methods have been proposed for dealing with multi-label classification problems. The most direct and simple approach for multi-label classification task is binary relevance (BR) method [1], [4], [23]. The BR method transforms a given multi-label classification task with d labels into d binary classification tasks. More specifically, the d labels are supposed as independent with each other and are predicted separately in the testing phase.

The shortcoming of the technique is that it cannot model any label dependence, which treats multi-label classification as simple. Though the existing disadvantages of BR, it also exhibits competitive advantages: (1) each binary classifier can be built directly as a base classifier; (2) the complexity is linear with respect to the numbers of labels; (3) in spite of its simplicity, it obtains competitive results in multi-label classification problems; (4) the BR method has been proven theoretically and empirically that it exhibits quite strong performance in terms of decomposable loss functions. It can be explained from a probabilistic point of view.

C. Traditional classifier chains method-CC

On the basis of BR method, the classifier chains model(CC) has been proposed as an important improvement from some papers[6], [7], [23]. It could be seen as an alternative method for multi-label classification, which overcomes the disadvantages of BR method, achieves higher performance and maintains the computational efficiency of BR method. As its name suggests, CC selects randomly a label ordering and trains each binary classifier following this ordering.

In the training phase, a classifier chains model consists of d base binary classifiers that are linked in a chain, and the feature space of each classifier is extended with the true label information of all previous labels in the chain. For instance, if the chain follows the order of labels $C_1 \rightarrow C_2 \rightarrow ... \rightarrow C_d$, then each classifier h_i in the chain is trained to learn association of label C_i given $(X, c_1, c_2, ..., c_{i-1})$. In the prediction phase, for an unknown instance X without labels, a prediction $\hat{C} = (\hat{c}_1, ..., \hat{c}_d)$ is produced by successively querying each classifier h_i. However, the inputs of these classifiers are not well-defined, since the true attributes $c_1, c_2, ..., c_{i-1}$ are not available at prediction stage. These missing values are therefore replaced by their respective predictions, for instance, c_1 is replaced by $\hat{c_1} = h_1(X)$, $\hat{c_1}$ is used by $h_2(X)$ as an additional input, c_2 is replaced by $\hat{c_2} = h_2(X)$, $\hat{c_1}$ and $\hat{c_2}$ are used by $h_3(X)$ as additional inputs, and so forth. Finally, the class vector is informed by concatenating the outputs of all binary classifiers in the chain. And we select Naive Bayesian (NB) classifier as the base classifier [24], [25], [26].

It is well known that a single CC model can be poorly ordered. Moreover, there is the possible effect of error propagation along the chain in classification phase. Thus,

the ensemble of classifier chains (ECC) method has been proposed in [6], [7], [8], [9], which takes random subsets of attributes and instances or random label orders. In this paper, m chain classifiers are trained by changing the order of the class variables in the chain. Finally, the label vector is obtained by using a voting scheme. Though the ECC method overcomes the instability of label prediction, it has quite higher computational complexity.

Recently, the Bayesian Chain Classifiers (BCC) has been presented in [10], [11], [12]. In the method, the parents of each class variable are only extended as additional attributes because we can represent the joint probability distribution of class variables given the features as a Bayesian network. But the main disadvantage is that learning a Bayesian network is difficult, especially with many variables. The process is:

$$C^* = argmax_{c_1,...,c_d} \prod_{i=1}^{d} P(c_i|\mathbf{pa}(C_i), X) \tag{2}$$

where $\mathbf{pa}(C_i)$ are the parents of the ith class variable.

III. Max-relevance and min-redundancy algorithm

In this section, we introduce mainly basic max-relevance and min-redundancy feature selection algorithm. There are two parts: (1) introduce the entropy and mutual information; (2) describe the max-relevance and min-redundancy feature selection algorithm.

A. Entropy and mutual information

Shannon's entropy function $H(X)$ of a random variable X measures its priori uncertainty in terms of its probability [17]. It is widely used in different domains. The conditional entropy function $H(X|Y)$ represents a posteriori uncertainty of X after Y. The MI is usually used to learn feature subset selection method [14], which measures the amount of uncertainty in X which is reduced if Y has been observed.

Definition 3.1([17]): For a continuous variable X, $H(X)$ is expressed as follows:

$$H(X) = \int_{-\infty}^{+\infty} \rho(x) \log \frac{1}{\rho(x)} dx \tag{3}$$

for discrete variable X, the $H(X)$ is defined as:

$$H(X) = \sum \rho(x) \log \frac{1}{\rho(x)} \tag{4}$$

and for discrete variables X and Y, the $H(X|Y)$ is computed as:

$$H(X|Y) = \sum \sum \rho(x, y) \log \frac{1}{\rho(x|y)} \tag{5}$$

Finally, the mutual information(MI) $I(X;Y)$ between X and Y can be defined as:

$$I(X;Y) = H(X) - H(X|Y) \tag{6}$$

B. The existing max-relevance and min-redundancy algorithm

In recent years, the max-relevance and min-redundancy feature selection approach has been proposed for the classification task [13], [19]. And the relevances and redundancies between variables are measured by mutual information(MI).

Definition3.2([17]): For a classification problem, let X_i is a candidate feature variable, X_j is a selected variable, C is the class variable, we call:

1) Relevance$(X_i;C)$: it indicates the relevance between the feature X_i and the class variable C by using mutual information as a measurement.
2) Redundancy$(X_i;X_j)$: it indicates the redundancy between any pair of candidate variables X_i and X_j measured by mutual information.

Definition3.3([13]): The purpose of feature selection is to find a feature set S with m features $X_i, i = 1, ..., m$, it satisfies:

1) Max-Relevance: it is to find features according to (7), which equals to the mean value of all mutual information values between individual feature X_i and class C:

$$\max D(S,C), \quad D = \frac{1}{|S|} \sum_{X_i \in S} I(X_i;C) \tag{7}$$

2) Min-Redundancy: it is the mean value of all mutual information values between any pair of candidate variables X_i and X_j in the feature set S. The equation is denoted as follows:

$$\min R(S), \quad R = \frac{1}{|S|^2} \sum_{X_i, X_j \in S} I(X_i, X_j) \tag{8}$$

The criterion combining the above two constraints is called "minimal-redundancy-maximal-relevance"(mRMR) [13]. The operator $\Phi(D,R)$ by combining D and R is optimized simultaneously by following equation:

$$\max \Phi(D,R), \quad \Phi = D/R \tag{9}$$

In fact, the incremental search method can be used to find the near-optimal features defined by $\Phi(.)$. Assume that we have the existing feature set S^{m-1} with $m-1$ features. The object is to select the mth feature from the set $X - S^{m-1}$. This is done by selecting the feature that maximizes $\Phi(.)$. Recently, an improved max-relevance and min-redundancy feature selection algorithm based on normalized mutual information (NMIFS) has been proposed by Vinh et al in [19]. And MI takes value in the range $[0, 1]$ for this algorithm. That is:

$$\max_{X_j \in X - S^{m-1}} [\frac{I(X_j;C)}{\min\{H(X_j);H(C)\}} / \frac{1}{|S^{m-1}|} \sum_{X_i \in S^{m-1}} \frac{I(X_j;X_i)}{\min\{H(X_j);H(X_i)\}}] \tag{10}$$

IV. Selective classifier chains based on mRMR–mRMR-CC

In this section, we propose mainly the dynamic selective classifier chains method based on max-relevance and min-redundancy algorithm, called mRMR-CC method. Our mRMR-CC model relies on two main respects: first, it uses the max-relevance and min-redundancy feature selection algorithm as a measure to extend the feature space from all previous class variables for each base classifier (inherit attributes are viewed as unchanged). We use the normalized mutual information NMIFS algorithm. Second, the most probable prediction of the whole class vector is estimated by the concatenation of the most probable individual class variables.

The first respect is the core of research in base classifier chains model. It is well known that the main advantage

of classifier chains method is to consider the dependencies between class variables. In each base classifier, the attributes space is extended by all previous class variables down the chain. In fact, the process corresponds to a feature selection process. Thus, in order to consider the redundancies and relevances between class variables, we use the mRMR algorithm. In second respect, since the whole abduction inference problem is an NP-hard problem, we use the concatenation of individual classes used widely in BR and CC.

A. Training phase

Generally, each classifier incorporates the attributes extended by all previous labels as additional attributes. At present, the feature space is extended with true label information of the selected labels by NMIFS method at each classifier. The following algorithm provides the selection process of the ith class label C_i $(i = 1, ..., d)$ for additional attributes. Firstly, S_{i-1} is initially empty set and represents the additional attributes set for label C_i $(S_0 = \emptyset)$. The algorithm selects additional attributes by the incremental search method (normalized mutual information NMIFS method) at every time and obtains a series of additional attributes sets $S^1 \subset S^2 \subset ... \subset S^{i-1}$ $(1, 2, ..., i-1$ represents the number of additional attributes). Then the algorithm selects compact additional attributes set S_{i-1} from $S^1, S^2, ..., S^{i-1}$ for label C_i. If a candidate attribute C makes $D/R(S_{i-1} \bigcup C)$ value increase, that is, the relevance between C_i and S_{i-1} is increasing and the redundancy between any pair of variables of S_{i-1} is reducing, then we will keep the attribute for S_{i-1}. Otherwise, we remove it. Finally, the algorithm obtains S_{i-1} with the largest D/R value from $D/R(S^1), D/R(S^2), ..., D/R(S^{i-1})$, which could keep the maximal relevance between the set with C_i and the minimal redundancy between additional attributes.

We could summarize the Algorithm 1 process as follows and initialize $S_{i-1} = \emptyset$, $D/R(S_{i-1}) = 0$, $A_i = \{C_1, C_2, ..., C_{i-1}\}$, A_i is the candidate attributes set of the ith label C_i:

1) First, for label C_i, $\forall C_j \in A_i - S_{i-1}$, compute $\{\frac{I(C_j;C_i)}{\min\{H(C_j);H(C_i)\}} / \frac{1}{|S_{i-1}|}\sum_{C_r \in S_{i-1}} \frac{I(C_j;C_r)}{\min\{H(C_j);H(C_r)\}}\}$,choose $C = \arg\max_{C_j}[\frac{I(C_j;C_i)}{\min\{H(C_j);H(C_i)\}} / \frac{1}{|S_{i-1}|}\sum_{C_r \in S_{i-1}} \frac{I(C_j;C_r)}{\min\{H(C_j);H(C_r)\}}]$, and get $S^1 = S_{i-1} \bigcup C$;
2) Then, let $S_{i-1} = S^1$, repeat the above process and get $S^2 = S^1 \bigcup C$, and so forth. This leads to $i-1$ sequential additional attributes sets $S^1 \subset S^2 \subset ... \subset S^{i-1}$;
3) Compute $D/R(S^k)$ (according to (7),(8),(9) and use normalized mutual information, label is C_i, $D/R(S^k)$ is the D/R value of selected set S^k, $1 \le k \le i-1$), k is the number of the attributes in S^k, and compare all $D/R(S^k)$ values of the $i-1$ sequential additional attributes sets, finally select the S^k with the largest D/R value. Thus $S_{i-1} = S^k$ is the compact additional attributes set of label C_i.

The method provides a dynamic process of selection labels for each base classifier in the classifier chains. The specific training process is as follows: first, we randomly select a label ordering, $C_1 \to C_2 \to ... \to C_d$, which consists of d base binary classifiers. Then, each classifier $h_i (i = 1, ..., d)$ in the chain is trained to learn the association of label C_i given (X, S_{i-1}), where S_{i-1} represents the selected additional attributes set in the ith classifier from all previous class variables along the chain. That is, for each classifier h_i, the training data consists of instance (X, S_{i-1}) labeled with C_i. The dynamic training process is defined as Algorithm 2:

Algorithm 2: mRMR-CC's training phase for the training set

```
    Training(D = {(X_1, C_1), ..., (X_N, C_N)})
1   for i = 1, ..., d
2     Do ▹ the ith binary transformation and training
3       D'_i ← {}
4       for (X, C) ∈ D
5       return Algorithm 1 get S_{i-1}
6         do X' ← [x_1, ..., x_n, S_{i-1}]
7           D'_i ← D'_i ⋃ (X', C_i)
8       ▹ train h_i to predict binary relevance of C_i
9       do D'_i → {0, 1}
```

The training process is shown in Fig. 1 and Fig. 2. We assume these labels are C_1, C_2, C_3 and C_4. Fig. 1 represents the whole selection process, Fig. 2 is the base classifier for label C_i – Naive Bayesian classifier (NB). NB classifier is a simple and effective classifier, which is widely used in classification problems. And it takes advantage of the conditional independence assumption, which makes model simplification. In our method, we consider mainly the extending process of labels in CC. Thus we select uniformly NB classifier as the base classifier. And the prediction structure is the same as the training structure:

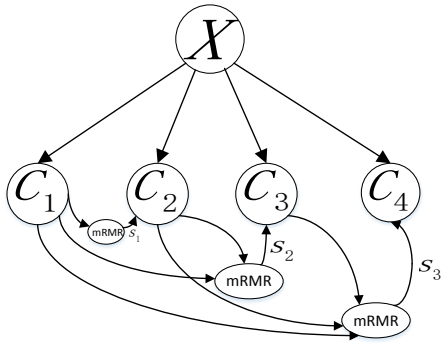


Fig. 1. Dynamic selective classifier chains method – mRMR-CC

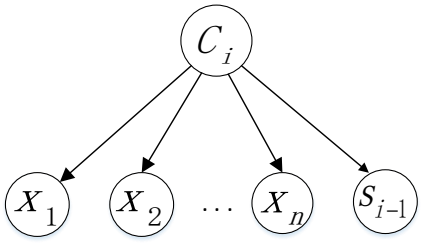


Fig. 2. The base classifier of mRMR-CC

B. Prediction phase

For the classifier chains method, the actually observed labels values $c_1, ..., c_{i-2}$ and c_{i-1} are available as additional attributes to train the binary classifier h_i only during the training stage, whereas this information is unknown for predicting the ith label C_i of a new instance. Thus, in order to make the mRMR-CC applicable, these values are replaced

by their respective predictions. In the prediction process of label C_i, the extended attributes are also the element of S_{i-1} determined by training process. For an unknown instance X, c_1 is replaced by $\hat{c_1} = h_1(X)$, and we could obtain the $\hat{S_1}$, then $\hat{S_1}$ is used by $h_2(X)$ as an additional attributes set. c_2 is replaced by $\hat{c_2} = h_2(X)$, and we could obtain the $\hat{S_2}$, $\hat{S_2}$ is used by $h_3(X)$ as an additional attributes set, and so on. Finally, we could obtain $\hat{S}_{i-1}$. The whole testing process of the label C_i is represented in Algorithm3($\hat{S_0} = \emptyset$):

Algorithm 3: mRMR-CC's prediction phase for the testing set

Classify(X)
▷ global $h = (h_1, ..., h_d)$
$C \leftarrow [\hat{c_1}, ..., \hat{c_d}]$
for $i = 1, ..., d$
predict $c_1, ..., c_{i-1}$ and get $\hat{S}_{i-1}$
  do $X' \leftarrow [x_1, ..., x_n, \hat{S}_{i-1}]$
    $\hat{c_i} \leftarrow h_i(X')$
return $\hat{C}$

The method leads to promising improvement on exploiting the relevances and redundancies between labels. It does not include other processes. Like Bayesian Chain Classifiers method(BCC), it learns the label dependence structure with a preprocessing step. But the mRMR-CC method considers the relevances and redundancies between labels by using a dynamic and direct process. Especially, it is quite effective with many label variables.

C. Computational complexity

Firstly, the BR's complexity is $O(d \times f(n, N))$, where $f(n, N)$ represents the complexity of the n attributes and N samples in each classifier. The CC's complexity is computed as $O(d \times f(n+d, N))$, and the number of additional attributes is d at most. Assume that each classifier is linear, the $f(n, N)$ becomes $nf(1, N)$. Then, the CC's complexity is $O(d \times f(n + d, N)) = O(d \times n \times f(1, N) + d \times d \times f(1, N))$. When $d \leq n$, the $O(d \times n \times f(1, N) + d \times d \times f(1, N))$ is approximately equal to $O(d \times n \times f(1, N))$, which corresponds to the complexity of BR. The $O(d \times n \times f(1, N) + d \times d \times f(1, N))$ is approximately equal to $O(d \times d \times f(1, N))$ when $n \leq d$. The mRMR-CC method mainly intends to select compact additional attributes set for CC method, which greatly reduces the number of additional attributes. That is, the d is reduced. And the complexity of CC is also reduced when the d is reduced without considering other processes, which is significative for CC method.

V. Numeric experiments

In the section, to verify the effectiveness of the proposed mRMR-CC method, we test it with four continuous data sets and five discrete data sets in MATLAB. For these data sets, we empirically evaluate the mRMR-CC and compare it against other state-of-the-art multi-label classifiers.

A. Data sets

In this experiment, nine benchmark datasets are used with media, biology, and text three different application areas [27]. These data sets are described concretely in TABLE I: N is the size of the data set, d is the number of binary classes or labels, n is the number of features, a indicates numeric attributes and b indicates binary attributes. All class variables are binary and the attributes are discrete or continuous in these data sets. These data sets are not dealt with additional process like discretization approach. Each base classifier uses Naive Bayesian classifier.

TABLE I
The Multi-label Datasets used in the experiment

Datasets	N	d	n	LC	PU	PM	DOMAIN
Emotion	590	6	72a	1.87	0.046	0.137	media
Scene	2407	6	294a	1.07	0.006	0.168	media
Yeast	2417	14	103a	4.24	0.082	0.098	biology
Slashdot	3782	22	1079b	1.18	0.041	0.139	text
Genbase	661	27	1185b	1.25	0.048	0.257	biology
Medical	978	45	1449b	1.25	0.096	0.158	text
Enron	1702	53	1001b	3.38	0.442	0.096	text
Langlog	1460	75	1004b	1.18	0.208	0.142	text
Retures	6000	103	500a	1.46	0.147	0.064	text

Besides, there are sparse labels in some data sets, such as Genbase, Medical and so on [28], [29]. In these datasets, the distributions of labels are unbalanced, which cause some redundant information between labels. Thus we firstly reduce some sparse labels from these datesets since we mainly intend to test our experiments on dense labels for their relevances and redundancies.

B. Multi-label evaluation metrics

Recently, several evaluation metrics have been used to measure the performances of multi-label classifiers [7], [12]. These metrics are parted into two respects: (1) evaluating the performance of the multi-label classifier over each class independently of the rest. (2) measuring the performance of all the classes at the same time. Thus, to verify the effectiveness of this method, we select some evaluation metrics from two respects. These evaluation metrics are described as follows:

1) Mean accuracy of the d class variables:

$$M - Acc = \frac{1}{d}\sum_{j=1}^{d}\frac{1}{N}\sum_{i=1}^{N}\delta(c'_{ij}, c_{ij}) \tag{11}$$

where c'_{ij} denotes the C_j class value predicted by the model for instance i and c_{ij} is its true value, and $\delta(c'_{ij}, c_{ij}) = 1$ if $c'_{ij} = c_{ij}$ and 0 otherwise.

2) Hamming Loss is the simplest loss function, which is defined as:

$$Hamming\ Loss = 1 - \frac{1}{N}\sum_{i=1}^{N}\frac{1}{d}\sum_{j=1}^{d}\delta(c'_{ij}, c_{ij}) \tag{12}$$

3) F-measure is also called the harmonic mean between precision and recall, and it is calculated per label and then averaged. In order to distinguish two types of F-measure, we call it as F-measure1:

$$F - measure1 = \frac{1}{d}\sum_{j=1}^{d}\frac{1}{N}\sum_{i=1}^{N}\frac{2p_{ij}r_{ij}}{(p_{ij} + r_{ij})} \tag{13}$$

where p_{ij} and r_{ij} are the precision and recall for C_j and instance i.

4) 0/1 Loss as a loss measure:

$$0/1\ Loss = 1 - \frac{1}{N}\sum_{i=1}^{N}\delta(C_i', C_i) \tag{14}$$

where $\delta(C_i', C_i) = 1$ if $C_i' = C_i$ and 0 otherwise. We call for a total coincidence on all components of the vector of predicted classes C_i' and the vector of real classes C_i.

5) Multi-label accuracy is also called Jaccard measure, which is defined as follows:

$$ML - Acc = \frac{1}{N}\sum_{i=1}^{N}\frac{|C_i' \wedge C_i|}{|C_i' \vee C_i|} \tag{15}$$

where in the numerator we count the number of coincidences of the two vectors, and the denominator we count the number of labels covered by some of both vectors.

6) Another F-measure is based-sample set evaluation, and we call it as F-measure2: where p_i and r_i are the precision and the recall for C_j.

$$F - measure2 = \frac{1}{N}\sum_{i=1}^{N}\frac{2p_i r_i}{(p_i + r_i)} \tag{16}$$

C. Experiments and conclusion

In this section, different experiment methods are used to compare the proposed mRMR-CC method. These methods include CC, ECC, the tree Naive Bayesian chain classifier (TNBCC)and Path-Bayesian chain classifier(Path-BCC)[12]. Our method mainly intends to improve base CC method, thus we compare it with CC method at first. Then on the basis of it, we compare other multi-label classification methods with our method in pairs. Finally, we compare all methods and get the average ranking for each evaluation metric. Since the numbers of the variables and samples are different in these data sets, we use 10-cold cross-validation for the first three smaller data sets and 5-cold cross-validation for the remaining larger data sets. These comparisons are divided into the following three parts:

1) mRMR-CC against traditional classifier chains method.
2) mRMR-CC against the ensemble of classifier chains, the tree Naive Bayesian chain classifier and the Path-Bayesian chain classifier in pairs.
3) the whole comparison.

1) mRMR-CC vs CC: In this section, we compare mainly the base CC and the selective classifier chains mRMR-CC. These results are shown in the Table II: (a)in ML-Acc, we obtain five wins; (b)in M-Acc, F-measure 1,F-measure 2 and Hamming Loss, the mRMR-CC outperforms the CC in six datasets; (c)in 0/1 Loss, our method obtains seven wins. On the one hand, from the Table II, it can be shown that in average, the mRMR-CC method achieves better performance than traditional CC in most data sets.

On the other hand, these results are shown in the Fig. 3. These figures represent the performance of CC and mRMR-CC in M-Acc, ML-Acc, F-measure1, F-measure2, 0/1 Loss, Hamming Loss,respectively. In the Fig. 3 (a)-(d), the higher the evaluation value, the better. We could see that the proposed mRMR-CC method is better than the traditional chains classifiers. In the Fig. 3 (e)-(f), the lower the evaluation value, the better. The mRMR-CC method outperforms the CC method in most data sets. The size of the change is not very obvious since the number of labels is far less than the number of features in most multi-label data sets. However, in terms of performance, the experiment shows that considering redundant information between classes clearly benefits traditional classifier chains method.

TABLE II
THE COMPARISON OF THE CC AND THE MRMR-CC METHOD

Datasets	mRMR-CC	CC	mRMR-CC	CC
	M-Acc		F-measure1	
Emotion	**0.7417 ± 0.0280**	0.7403 ± 0.0314	**0.5112 ± 0.0401**	0.5029 ± 0.0568
Scene	0.7782 ± 0.0090	**0.8018 ± 0.0061**	**0.1725 ± 0.0574**	0.0855 ± 0.0074
Yeast	0.6703 ± 0.0090	**0.6753 ± 0.0129**	**0.3835 ± 0.0178**	0.3624 ± 0.0092
Slashdot	**0.9473 ± 0.0033**	0.9272 ± 0.0031	0.4283 ± 0.0446	**0.4309 ± 0.0446**
Genbase	**0.9816 ± 0.0077**	0.9814 ± 0.0078	**0.7447 ± 0.0631**	0.7445 ± 0.0636
Medical	**0.8807 ± 0.0068**	0.8696 ± 0.0090	**0.2812 ± 0.0043**	0.2787 ± 0.0001
Enron	**0.7029 ± 0.0182**	0.7029 ± 0.0182	**0.4230 ± 0.0139**	0.4227 ± 0.0076
Langlog	0.4916 ± 0.0338	**0.4936 ± 0.0347**	0.1728 ± 0.0146	**0.1740 ± 0.0155**
Retures	**0.9494 ± 0.0018**	0.9494 ± 0.0018	-	-
	0/1 Loss		ML-Acc	
Emotion	**0.8264 ± 0.0383**	0.8265 ± 0.0322	**0.4778 ± 0.0498**	0.4724 ± 0.0555
Scene	0.9165 ± 0.0197	**0.8638 ± 0.0133**	**0.1844 ± 0.0248**	0.1452 ± 0.0150
Yeast	**0.9905 ± 0.0047**	0.9909 ± 0.0028	0.3473 ± 0.0136	**0.3482 ± 0.0145**
Slashdot	0.2800 ± 0.0159	**0.2792 ± 0.0151**	0.1858 ± 0.0127	**0.1868 ± 0.0124**
Genbase	**0.0620 ± 0.0211**	0.0629 ± 0.0213	**0.1200 ± 0.0200**	0.1100 ± 0.0211
Medical	**0.4885 ± 0.0235**	0.5269 ± 0.0308	**0.2899 ± 0.0002**	0.2732 ± 0.0238
Enron	**0.8628 ± 0.0245**	0.8628 ± 0.0245	**0.3784 ± 0.0102**	0.3784 ± 0.0102
Langlog	**0.8582 ± 0.0125**	0.8588 ± 0.0125	0.0813 ± 0.0075	**0.0819 ± 0.0084**
Retures	**0.2750 ± 0.0068**	0.2750 ± 0.0068	-	-
	F-measure2		Hamming Loss	
Emotion	**0.6164 ± 0.0541**	0.6105 ± 0.0593	**0.2583 ± 0.0280**	0.2597 ± 0.0031
Scene	**0.2254 ± 0.0375**	0.1495 ± 0.0161	0.2218 ± 0.0090	**0.1982 ± 0.0061**
Yeast	**0.5060 ± 0.0149**	0.5027 ± 0.0143	0.3502 ± 0.0091	**0.3247 ± 0.0129**
Slashdot	0.1896 ± 0.0120	**0.1906 ± 0.0120**	**0.0526 ± 0.0032**	0.0628 ± 0.0031
Genbase	**0.1127 ± 0.0230**	0.1124 ± 0.0200	**0.0101 ± 0.0021**	0.0102 ± 0.0032
Medical	**0.3265 ± 0.0025**	0.3165 ± 0.0166	**0.1193 ± 0.0068**	0.1305 ± 0.0090
Enron	**0.4791 ± 0.0020**	0.4791 ± 0.0020	**0.2971 ± 0.0182**	0.2971 ± 0.0182
Langlog	0.1315 ± 0.0116	**0.1319 ± 0.0125**	0.5084 ± 0.0339	**0.5064 ± 0.0347**
Retures	-	-	**0.0506 ± 0.0018**	0.0506 ± 0.0018

TABLE III
THE COMPARISON OF THE ECC AND THE MRMR-CC METHOD

Datasets	mRMR-CC	ECC	mRMR-CC	ECC
	M-Acc		F-measure1	
Emotion	**0.7417 ± 0.0280**	0.7298 ± 0.0569	0.5112 ± 0.0401	**0.5134 ± 0.0530**
Scene	0.7782 ± 0.0090	**0.7966 ± 0.0047**	**0.1725 ± 0.0574**	0.1138 ± 0.0164
Yeast	0.6703 ± 0.0090	**0.6708 ± 0.0115**	0.3835 ± 0.0178	**0.3886 ± 0.0141**
Slashdot	**0.9473 ± 0.0033**	0.9270 ± 0.0031	**0.4283 ± 0.0446**	0.4268 ± 0.0461
Genbase	**0.9816 ± 0.0077**	0.9815 ± 0.0078	**0.7447 ± 0.0631**	0.7440 ± 0.0567
Medical	**0.8807 ± 0.0068**	0.8696 ± 0.0090	**0.2812 ± 0.0043**	0.2787 ± 0.0001
Enron	**0.7029 ± 0.0182**	0.7029 ± 0.0182	**0.4230 ± 0.0139**	0.4230 ± 0.0139
Langlog	0.4916 ± 0.0338	**0.4936 ± 0.0347**	0.1728 ± 0.0146	**0.1740 ± 0.0155**
Retures	**0.9494 ± 0.0018**	0.9494 ± 0.0018	-	-
	0/1 Loss		ML-Acc	
Emotion	**0.8264 ± 0.0383**	0.8281 ± 0.0338	0.4778 ± 0.0498	**0.4813 ± 0.0504**
Scene	0.9165 ± 0.0197	**0.8714 ± 0.0188**	**0.1844 ± 0.0248**	0.1330 ± 0.0191
Yeast	**0.9905 ± 0.0047**	0.9921 ± 0.0027	0.3473 ± 0.0136	**0.3500 ± 0.0121**
Slashdot	0.2800 ± 0.0159	**0.2787 ± 0.0138**	**0.1858 ± 0.0127**	0.1856 ± 0.0120
Genbase	**0.0620 ± 0.0211**	0.0628 ± 0.0213	**0.1200 ± 0.0200**	0.1105 ± 0.0220
Medical	**0.4885 ± 0.0235**	0.5264 ± 0.0302	**0.2899 ± 0.0002**	0.2732 ± 0.0238
Enron	**0.8628 ± 0.0245**	0.8628 ± 0.0245	**0.3784 ± 0.0102**	0.3784 ± 0.0102
Langlog	**0.8582 ± 0.0125**	0.8599 ± 0.0125	0.0813 ± 0.0075	**0.0817 ± 0.0082**
Retures	**0.2750 ± 0.0068**	0.2750 ± 0.0068	-	-
	F-measure2		Hamming Loss	
Emotion	0.6164 ± 0.0541	**0.6194 ± 0.0558**	**0.2583 ± 0.0280**	0.2683 ± 0.0280
Scene	**0.2254 ± 0.0375**	0.1364 ± 0.0196	0.2218 ± 0.0090	**0.2034 ± 0.0470**
Yeast	0.5060 ± 0.0149	**0.5076 ± 0.0145**	0.3502 ± 0.0091	**0.3292 ± 0.0115**
Slashdot	**0.1896 ± 0.0120**	0.1893 ± 0.0116	**0.0526 ± 0.0032**	0.0527 ± 0.0029
Genbase	**0.1127 ± 0.0230**	0.1125 ± 0.0231	**0.0101 ± 0.0021**	0.0102 ± 0.0031
Medical	**0.3265 ± 0.0025**	0.3165 ± 0.0166	**0.1193 ± 0.0068**	0.1305 ± 0.0090
Enron	**0.4791 ± 0.0020**	0.4791 ± 0.0020	**0.2971 ± 0.0182**	0.2971 ± 0.0182
Langlog	0.1315 ± 0.0116	**0.1319 ± 0.0125**	0.5084 ± 0.0339	**0.5064 ± 0.0347**
Retures	-	-	**0.0506 ± 0.0018**	0.0506 ± 0.0018

2) mRMR-CC vs ECC, TNBCC and Path-BCC: (a) There are some comparisons between the mRMR-CC and ECC. In order to conduct the ECC method, we adopt the voting method by selecting different classifier chains. Obviously, the complexity of ECC is higher than CC method. From

the Table III, in F-measure1, ML-Acc and F-measure2, we obtain five wins; in M-Acc and Hamming Loss, the mRMR-CC wins ECC in six data sets; and in 0/1 Loss, our method outperforms ECC in seven datasets. In addition to, there is high computational complexity in the ensemble of classifier chains. Finally, the results show that the mRMR-CC is an effective approach.

TABLE IV
THE COMPARISON OF THE TNBCC AND THE MRMR-CC METHODS

Datasets	mRMR-CC	TNBCC	mRMR-CC	TNBCC
	M-Acc		F-measure1	
Emotion	**0.7417 ± 0.0280**	0.7266 ± 0.0302	0.5112 ± 0.0401	**0.5207 ± 0.0560**
Scene	0.7782 ± 0.0090	**0.7859 ± 0.0149**	0.1725 ± 0.0574	**0.2629 ± 0.0018**
Yeast	0.6703 ± 0.0090	**0.7085 ± 0.0139**	0.3835 ± 0.0178	**0.4584 ± 0.0125**
Slashdot	0.9473 ± 0.0033	**0.9478 ± 0.0032**	**0.4283 ± 0.0446**	0.4269 ± 0.0456
Genbase	**0.9816 ± 0.0077**	0.9815 ± 0.0077	**0.7447 ± 0.0631**	0.7446 ± 0.0629
Medical	**0.8807 ± 0.0068**	0.8805 ± 0.0068	0.2812 ± 0.0043	**0.2813 ± 0.0043**
Enron	**0.7029 ± 0.0182**	0.7028 ± 0.0174	0.4230 ± 0.0139	**0.4269 ± 0.0131**
Langlog	0.4916 ± 0.0338	**0.4936 ± 0.0393**	0.1728 ± 0.0146	**0.1738 ± 0.0155**
Retures	**0.9494 ± 0.0018**	0.9494 ± 0.0018	-	-
	0/1 Loss		ML-Acc	
Emotion	**0.8264 ± 0.0383**	0.8669 ± 0.0337	**0.4778 ± 0.0498**	0.4642 ± 0.0464
Scene	0.9165 ± 0.0197	**0.8600 ± 0.0302**	0.1844 ± 0.0248	**0.2718 ± 0.0344**
Yeast	0.9905 ± 0.0047	**0.8953 ± 0.0161**	0.3473 ± 0.0136	**0.4578 ± 0.0151**
Slashdot	**0.2800 ± 0.0159**	0.2855 ± 0.0122	**0.1858 ± 0.0127**	0.1848 ± 0.0128
Genbase	**0.0620 ± 0.0211**	0.0621 ± 0.0211	**0.1200 ± 0.0200**	0.1199 ± 0.0100
Medical	**0.4885 ± 0.0235**	0.4885 ± 0.0235	**0.2899 ± 0.0002**	0.2899 ± 0.0002
Enron	**0.8628 ± 0.0245**	0.8630 ± 0.0245	0.3784 ± 0.0102	**0.3793 ± 0.0096**
Langlog	**0.8582 ± 0.0125**	0.8590 ± 0.0125	0.0813 ± 0.0075	**0.0817 ± 0.0075**
Retures	**0.2750 ± 0.0068**	0.2750 ± 0.0068	-	-
	F-measure2		Hamming Loss	
Emotion	**0.6164 ± 0.0541**	0.6072 ± 0.0564	**0.2583 ± 0.0280**	0.2734 ± 0.0302
Scene	0.2254 ± 0.0375	**0.3279 ± 0.0467**	0.2218 ± 0.0090	**0.2140 ± 0.0149**
Yeast	0.5060 ± 0.0149	**0.6114 ± 0.0121**	0.3502 ± 0.0091	**0.2915 ± 0.0139**
Slashdot	**0.1896 ± 0.0120**	0.1885 ± 0.0121	0.0526 ± 0.0032	**0.0524 ± 0.0030**
Genbase	**0.1127 ± 0.0230**	0.1126 ± 0.0231	**0.0101 ± 0.0021**	0.0102 ± 0.0031
Medical	**0.3265 ± 0.0025**	0.3265 ± 0.0025	**0.1193 ± 0.0068**	0.1290 ± 0.0068
Enron	0.4791 ± 0.0020	**0.4801 ± 0.0017**	**0.2971 ± 0.0182**	0.2971 ± 0.0177
Langlog	0.1315 ± 0.0116	**0.1319 ± 0.0117**	**0.5084 ± 0.0339**	0.5092 ± 0.0336
Retures	-	-	**0.0506 ± 0.0018**	0.0506 ± 0.0018

TABLE V
THE COMPARISON OF THE PATH-BCC AND THE MRMR-CC METHODS

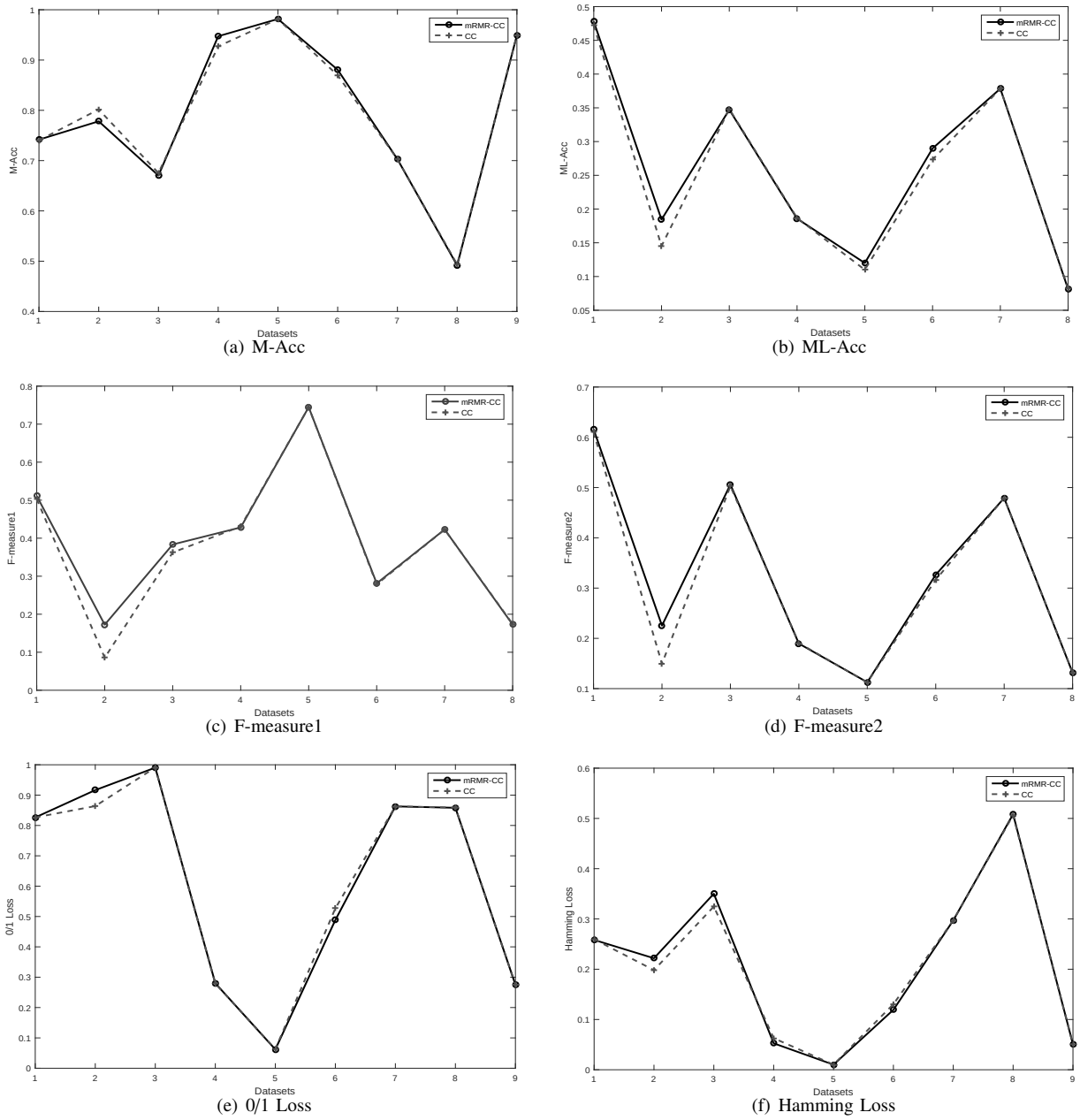
Datasets	mRMR-CC	Path-BCC	mRMR-CC	Path-BCC
	M-Acc		F-measure1	
Emotion	**0.7417 ± 0.0280**	0.7269 ± 0.0190	**0.5112 ± 0.0401**	0.5003 ± 0.0450
Scene	0.7782 ± 0.0090	**0.7843 ± 0.0152**	0.1725 ± 0.0574	**0.2618 ± 0.0481**
Yeast	0.6703 ± 0.0090	**0.7108 ± 0.0139**	0.3835 ± 0.0178	**0.4463 ± 0.0222**
Slashdot	0.9473 ± 0.0033	**0.9478 ± 0.0032**	**0.4283 ± 0.0446**	0.4269 ± 0.0456
Genbase	**0.9816 ± 0.0077**	0.9815 ± 0.0077	**0.7447 ± 0.0631**	0.7446 ± 0.0629
Medical	**0.8807 ± 0.0068**	0.8807 ± 0.0068	0.2812 ± 0.0043	**0.2813 ± 0.0043**
Enron	**0.7029 ± 0.0182**	0.7029 ± 0.0192	**0.4230 ± 0.0139**	0.4214 ± 0.0162
Langlog	0.4916 ± 0.0338	**0.4928 ± 0.0337**	0.1728 ± 0.0146	**0.1746 ± 0.0156**
Retures	**0.9494 ± 0.0018**	0.9475 ± 0.0047	-	-
	0/1 Loss		ML-Acc	
Emotion	**0.8264 ± 0.0383**	0.8702 ± 0.0360	**0.4778 ± 0.0498**	0.4592 ± 0.0397
Scene	0.9165 ± 0.0197	**0.8712 ± 0.0356**	0.1844 ± 0.0248	**0.2723 ± 0.0389**
Yeast	0.9905 ± 0.0047	**0.9073 ± 0.0355**	0.3473 ± 0.0136	**0.4491 ± 0.0223**
Slashdot	**0.2800 ± 0.0159**	0.2855 ± 0.0122	**0.1858 ± 0.0127**	0.1848 ± 0.0128
Genbase	**0.0620 ± 0.0211**	0.0621 ± 0.0211	**0.1200 ± 0.0200**	0.1199 ± 0.0100
Medical	**0.4885 ± 0.0235**	0.4885 ± 0.0235	**0.2899 ± 0.0002**	0.2899 ± 0.0002
Enron	**0.8628 ± 0.0245**	0.8635 ± 0.0290	**0.3784 ± 0.0102**	0.3781 ± 0.0117
Langlog	**0.8582 ± 0.0125**	0.8582 ± 0.0125	0.0813 ± 0.0075	**0.0820 ± 0.0079**
Retures	**0.2750 ± 0.0068**	0.2750 ± 0.0068	-	-
	F-measure2		Hamming Loss	
Emotion	**0.6164 ± 0.0541**	0.6017 ± 0.0505	**0.2583 ± 0.0280**	0.2732 ± 0.0190
Scene	0.2254 ± 0.0375	**0.3207 ± 0.0461**	0.2218 ± 0.0090	**0.2170 ± 0.0162**
Yeast	0.5060 ± 0.0149	**0.6030 ± 0.0180**	0.3502 ± 0.0091	**0.2920 ± 0.0163**
Slashdot	**0.1896 ± 0.0120**	0.1885 ± 0.0121	0.0526 ± 0.0032	**0.0524 ± 0.0030**
Genbase	**0.1127 ± 0.0230**	0.1126 ± 0.0231	**0.0101 ± 0.0021**	0.0102 ± 0.0031
Medical	**0.3265 ± 0.0025**	0.3265 ± 0.0025	**0.1193 ± 0.0068**	0.1291 ± 0.0068
Enron	0.4791 ± 0.0020	**0.4792 ± 0.0029**	**0.2971 ± 0.0182**	0.2971 ± 0.0192
Langlog	0.1315 ± 0.0116	**0.1322 ± 0.0121**	0.5084 ± 0.0339	**0.5072 ± 0.0337**
Retures	-	-	**0.0506 ± 0.0018**	0.0506 ± 0.0018

(b) The next experiment compares the mRMR-CC and the TNBCC. The comparison is shown in Table IV based on six evaluation metrics. It is well known that the TNBCC method is a superior method against other methods in multi-label classifier. From the Table IV, we could summarize these results: (i) in ML-Acc and F-measure2, we obtain four wins; (ii) our method obtains five wins in M-Acc; (iii) the proposed mRMR-CC method obtains six wins in Hamming Loss; (iv) in 0/1 Loss, our method outperforms CC in seven data sets. Besides, in F-measure1, the performance of TNBCC is better than our method in six datasets.

According to these results, we could conclude that the mRMR-CC method is a competitive method against the TNBCC method. And the mRMR-CC method need not specially learn a dependence structure like TNBCC method. And the mRMR-CC method is used by a direct way. As a whole, the mRMR-CC method could lead to significative results.

TABLE VI
THE COMPARISON IN ALL ALGORITHMS

Datasets	mRMR-CC	CC	ECC	TNBCC	Path-BCC
	M-Acc				
Emotion	0.7417(1.0)	0.7403(2.0)	0.7298(3.0)	0.7266(5.0)	0.7269(4.0)
Scene	0.7782(5.0)	0.8018(1.0)	0.7966(2.0)	0.7859(3.0)	0.7843(4.0)
Yeast	0.6703(5.0)	0.6753(3.0)	0.6708(4.0)	0.7085(2.0)	0.7108(1.0)
Slashdot	0.9473(3.0)	0.9272(4.0)	0.9270(5.0)	0.9478(1.5)	0.9478(1.5)
Genbase	0.9816(1.0)	0.9814(5.0)	0.9815(3.0)	0.9815(3.0)	0.9815(3.0)
Medical	0.8807(1.5)	0.8696(4.5)	0.8696(4.5)	0.8805(3.0)	0.8807(1.5)
Enron	0.7029(2.5)	0.7029(2.5)	0.7029(2.5)	0.7028(5.0)	0.7029(2.5)
Langlog	0.4916(5.0)	0.4936(2.0)	0.4936(2.0)	0.4936(2.0)	0.4928(4.0)
Retures	0.9494(2.5)	0.9494(2.5)	0.9494(2.5)	0.9494(2.5)	0.9475(5.0)
Average Ranking	2.94	2.94	3.17	3.00	2.94
	ML-Acc				
Emotion	0.4778(2.0)	0.4724(3.0)	0.4813(1.0)	0.4642(4.0)	0.4592(5.0)
Scene	0.1844(3.0)	0.1452(4.0)	0.1330(5.0)	0.2718(2.0)	0.2723(1.0)
Yeast	0.3473(5.0)	0.3482(4.0)	0.3500(3.0)	0.4578(1.0)	0.4491(2.0)
Slashdot	0.1858(2.0)	0.1868(1.0)	0.1856(3.0)	0.1848(4.5)	0.1848(4.5)
Genbase	0.1200(1.0)	0.1100(5.0)	0.1105(4.0)	0.1199(2.5)	0.1199(2.5)
Medical	0.2899(2.0)	0.2732(4.5)	0.2732(4.5)	0.2899(2.0)	0.2899(2.0)
Enron	0.3784(3.0)	0.3784(3.0)	0.3784(3.0)	0.3793(1.0)	0.3781(5.0)
Langlog	0.0813(5.0)	0.0819(2.0)	0.0817(3.5)	0.0817(3.5)	0.0820(1.0)
Retures	-	-	-	-	-
Average Ranking	2.88	3.31	3.38	2.56	2.88
	Hamming Loss				
Emotion	0.2583(1.0)	0.2597(2.0)	0.2683(3.0)	0.2734(5.0)	0.2732(4.0)
Scene	0.2218(5.0)	0.1982(1.0)	0.2034(2.0)	0.2140(3.0)	0.2170(4.0)
Yeast	0.3502(5.0)	0.3247(3.0)	0.3292(4.0)	0.2915(1.0)	0.2920(2.0)
Slashdot	0.0526(3.0)	0.0628(5.0)	0.0527(4.0)	0.0524(1.5)	0.0524(1.5)
Genbase	0.0101(1.0)	0.0102(3.5)	0.0102(3.5)	0.0102(3.5)	0.0102(3.5)
Medical	0.1193(1.0)	0.1305(4.5)	0.1305(4.5)	0.1290(2.0)	0.1291(3.0)
Enron	0.2971(3.0)	0.2971(3.0)	0.2971(3.0)	0.2971(3.0)	0.2971(3.0)
Langlog	0.5084(4.0)	0.5064(1.5)	0.5064(1.5)	0.5092(5.0)	0.5072(3.0)
Retures	0.0506(3.0)	0.0506(3.0)	0.0506(3.0)	0.0506(3.0)	0.0506(3.0)
Average Ranking	2.89	2.94	3.17	3.00	3.00
	0/1 Loss				
Emotion	0.8264(1.0)	0.8265(2.0)	0.8281(3.0)	0.8669(4.0)	0.8702(5.0)
Scene	0.9165(5.0)	0.8638(2.0)	0.8714(4.0)	0.8600(1.0)	0.8712(3.0)
Yeast	0.9905(3.0)	0.9909(4.0)	0.9921(5.0)	0.8953(1.0)	0.9073(2.0)
Slashdot	0.2800(3.0)	0.2792(2.0)	0.2787(1.0)	0.2855(4.5)	0.2855(4.5)
Genbase	0.0620(1.0)	0.0629(5.0)	0.0628(4.0)	0.0621(2.5)	0.0621(2.5)
Medical	0.4885(2.0)	0.5269(5.0)	0.5264(4.0)	0.4885(2.0)	0.4885(2.0)
Enron	0.8628(2.0)	0.8628(2.0)	0.8628(2.0)	0.8630(4.0)	0.8635(5.0)
Langlog	0.8582(1.5)	0.8588(3.0)	0.8599(5.0)	0.8590(4.0)	0.8582(1.5)
Retures	0.2750(3.0)	0.2750(3.0)	0.2750(3.0)	0.2750(3.0)	0.2750(3.0)
Average Ranking	2.39	3.11	3.44	2.89	3.17
	F-measure1				
Emotion	0.5112(3.0)	0.5029(4.0)	0.5134(2.0)	0.5207(1.0)	0.5003(5.0)
Scene	0.1725(3.0)	0.0855(5.0)	0.1138(4.0)	0.2629(1.0)	0.2618(2.0)
Yeast	0.3835(4.0)	0.3624(5.0)	0.3886(3.0)	0.4584(1.0)	0.4463(2.0)
Slashdot	0.4283(2.0)	0.4309(1.0)	0.4268(5.0)	0.4269(3.5)	0.4269(3.5)
Genbase	0.7447(1.0)	0.7445(4.0)	0.7440(5.0)	0.7446(2.5)	0.7446(2.5)
Medical	0.2812(3.0)	0.2787(4.5)	0.2787(4.5)	0.2813(1.5)	0.2813(1.5)
Enron	0.4230(2.5)	0.4227(4.0)	0.4230(2.5)	0.4269(1.0)	0.4214(5.0)
Langlog	0.1728(5.0)	0.1740(2.5)	0.1740(2.5)	0.1738(4.0)	0.1746(1.0)
Retures	-	-	-	-	-
Average Ranking	2.94	3.75	3.56	1.94	2.81
	F-measure2				
Emotion	0.6164(2.0)	0.6105(3.0)	0.6194(1.0)	0.6072(4.0)	0.6017(5.0)
Scene	0.2254(3.0)	0.1495(4.0)	0.1364(5.0)	0.3279(1.0)	0.3207(2.0)
Yeast	0.5060(4.0)	0.5027(5.0)	0.5076(3.0)	0.6114(1.0)	0.6030(2.0)
Slashdot	0.1896(2.0)	0.1906(1.0)	0.1893(3.0)	0.1885(4.5)	0.1885(4.5)
Genbase	0.1127(1.0)	0.1124(5.0)	0.1125(4.0)	0.1126(2.5)	0.1126(2.5)
Medical	0.3265(2.0)	0.3165(4.5)	0.3165(4.5)	0.3265(2.0)	0.3265(2.0)
Enron	0.4791(4.0)	0.4791(4.0)	0.4791(4.0)	0.4801(1.0)	0.4792(2.0)
Langlog	0.1315(5.0)	0.1319(3.0)	0.1319(3.0)	0.1319(3.0)	0.1322(1.0)
Retures	-	-	-	-	-
Average Ranking	2.88	3.69	3.44	2.38	2.63



(a) M-Acc (b) ML-Acc

(c) F-measure1 (d) F-measure2

(e) 0/1 Loss (f) Hamming Loss

Fig. 3. The results on the test data for CC and mRMR-CC method. From (a)-(f), these figures show the experimental results of six evaluation metrics in different data sets.

(c) The proposed mRMR-CC and the Path-BCC method are compared in the section. The Path-BCC method incorporates all previous classes in the path towards the root of the tree as additional attributes. Detailed experiment results are summarized in Table V. From the Table V, the mRMR-CC is also effective than the Path-BCC method. In terms of the results, in F-measure1 and F-measure2, we obtain four wins; in M-Acc, ML-Acc and Hamming Loss, our method has five wins, and in 0/1 Loss, our method outperforms the CC method in seven data sets. Though the Path-BCC uses all previous classes as additional attributes for each base classifier, it also exists redundant information between classes. Thus, the experiment shows that using redundant labels is not beneficial to multi-label classification. The result value 0 (Retures) is not given in these tables.

3) the whole comparison: Finally, on the one hand, we compare the proposed mRMR-CC method with other methods and conclude these comparisons results in Table VI. We only summarize the mean value of each evaluation metric. The Table VI shows the average ranking of each algorithm in all evaluation metrics and all datasets. In M-Acc, Hamming Loss and 0/1 Loss, the mRMR-CC is ranked first in all algorithms. In ML-Acc, the method is ranked second in all methods. In F-measure1 and F-measure2, our method is ranked third in all methods. On average, the mRMR-CC is better than these multi-label methods.

On the other hand, these results are shown in the Fig. 4 and Fig. 5. In Fig. 4 and Fig. 5, the lower the value of average ranking, the better. The average rankings of mRMR-CC and CC in six evaluation measures are shown in Fig. 4. In Fig. 4, we could find that the mRMR-CC method is all ranked first in the two methods. On the whole, the proposed mRMR-CC method can improve the classifier chains method and has competitive results in terms of predictive performance.

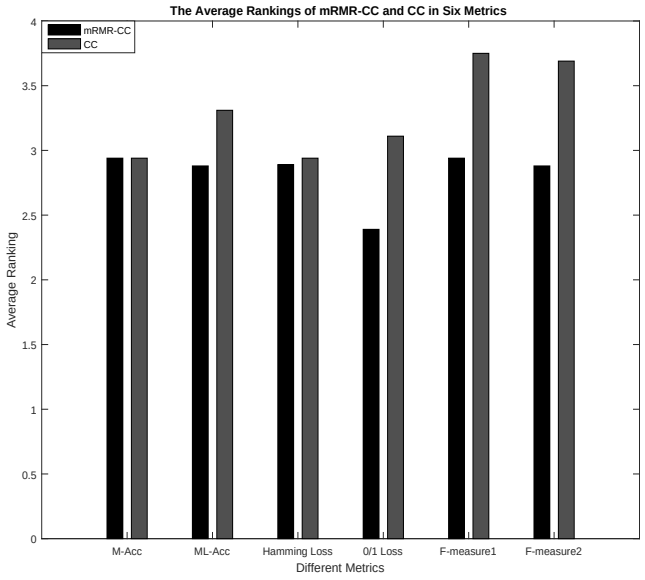
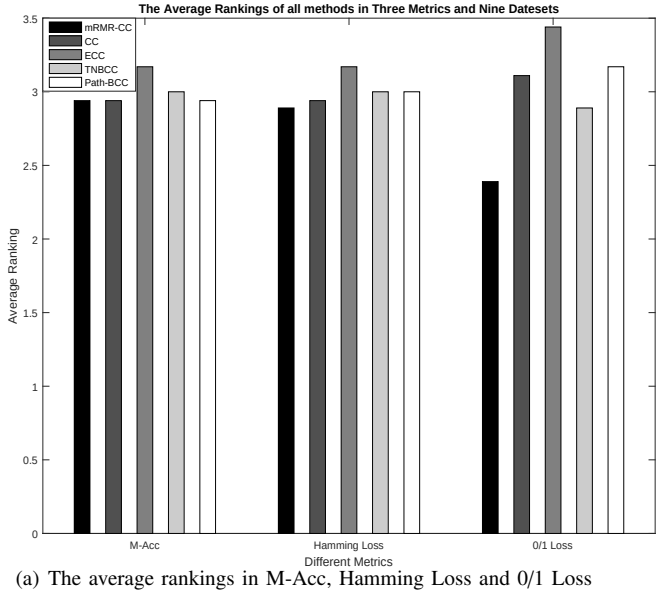
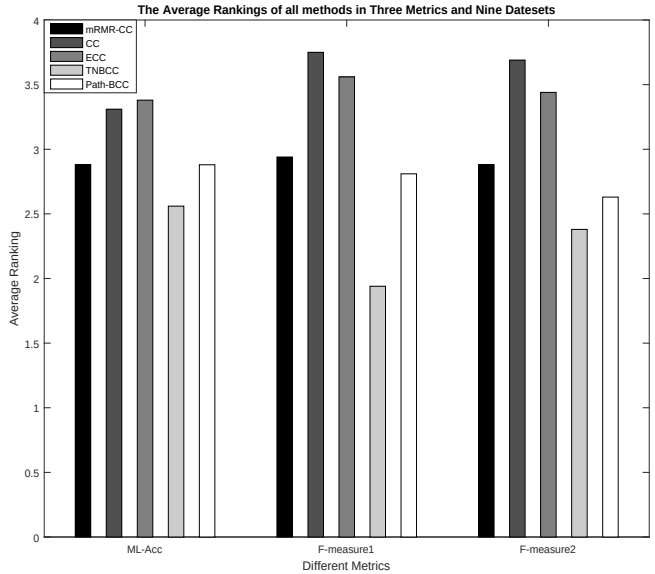


Fig. 4. The average rankings of mRMR-CC and CC in six metrics



(a) The average rankings in M-Acc, Hamming Loss and 0/1 Loss



(b) The average rankings in ML-Acc, F-measure1 and F-measure2

Fig. 5. The Average Rankings of all methods in Different Metrics.

And these results are described more clearly in Fig. 5. In Fig. 5 (a), we describe the average rankings of all methods in M-Acc, Hamming Loss and 0/1 Loss. We could find that the proposed mRMR-CC method is ranked first in all algorithms. The average rankings of all methods in ML-Acc, F-measure1 and F-measure2 are concluded in Fig. 5 (b). In ML-Acc, the method is ranked second in all algorithms, and in F-measure1 and F-measure2, it is ranked third in all methods. Thus, we could conclude that the mRMR-CC method leads to the promising improvement on the CC method. And the mRMR-CC method could consider all information between labels.

VI. Conclusion and future work

Although multi-label classification can be seen as a simple extension of the well-studied single-class classification, it comes with the challenge that labels generally display dependencies and redundancies amongst each other. In view of the classifier chains method, this paper proposes a dynamic process of selection labels based on the max-relevance and min-redundancy feature selection algorithm. To that end, the original input space is extended with the selected labels set for each classifier. Traditional classifier chains method only takes into account the dependencies between labels. The main goal of our method is to consider the redundancy between any pair of additional attributes based on the CC. In addition to, it could also measure the relevances between labels. The mRMR-CC method can be directly used to improve the classifier chains without another preprocessing process.

At present, these experiments results have shown that the mRMR-CC model is able to detect the labels redundancies by comparing the existing methods in MLC. This is due to the fact that the method uses all available information. Thus, in term of predictive performance, the method is available for CC method.

As future work, we will plan to further study how to deal with the noise problem of additional attributes and inhere attributes by a proper way. Moreover, since the label ordering has an important influence on the process of a classifier chain, we need to consider the problem in the next step. And we will find alternative method to improve the classifier chains.

Acknowledgment

The author thank the editors and the anonymous reviewers for helpful comments and suggestions. The research was supported by the National Natural Science Foundation of China (Grant No. 61573266).

References

[1] G. Tsoumakas and I. Katakis, "Multi-label classification: an overview," *International Journal of Data Warehousing and Mining*, vol. 3, no. 3, pp. 1-13, 2007.

[2] M. Boutell, J. Luo, X. Shen and C. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757-1771, 2004.

[3] S. Abe, "Fuzzy support vector machines for multi-label classification," *Pattern Recognition*, vol. 48, no. 6, pp. 2110-2117, 2015.

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

[5] K. Dembczyński, W. Waegeman, W. Cheng and E. Hüllermeier, "On label dependence and loss minimization in multi-label classification," *Machine Learning*, vol. 88, no. 1, pp. 5-45, 2012.
[6] J. Read, B. Pfahringer, G. Holmes and Eibe Frank, "Classifier chains for multi-label classification," *In ECML'09;20th European conference on maching learning 2009b*, pp. 254-269.
[7] J. Read, B. Pfahringer, G. Holmes and Eibe Frank, "Classifier chains for multi-label classification,"*Machine Learning*, vol. 85, no. 3, pp. 333-359, 2011.
[8] S. S. Ahmed, B. P. C. Rao and T. jayakumar, "Application of Multi-dimensional Chain classfiers to Eddy Current Images for Defect Characterization,"*Research Journal of Applied Sciences,Engineering and Technology*, vol. 24, no. 4, pp. 5544-5547, 2012.
[9] G. Tsoumakas and I. Vlahavas, "Random k-labelsets:an ensemble method for multilabel classification, "*Proceedings of the 18th European conference on Machine Learning 2007*, pp. 406-417.
[10] Rodriguez, J. D. Guez and J. A. lozano, "Multi-objective learning of multi-dimensional Bayesian classifiers,"*In:Proceedings of the Eighth internatioanla Conference on Hybrid Intelligent Systems 2008*, pp. 501-506.
[11] J. C. Zaragoza, L. E. Sucar and E. F. Morales, "A two-step method to learn multi-dimensiaonal Bayesian network classifiers based on mutual information measures,"*In:Proceedings of the 24th Internatioanal Florida Aritifivial Intelligence Research Society Conference(FLAIRS),AAAI Press 2011a*, pp. 644-649.
[12] L. Enrique Sucar, C. Bielza, E. F. Morales, P. Hemandez-Leal, J. H. Zaragoca and P. Larrañaga, "Multi-label classification with Bayesian network-based chain classifiers,"*Pattern Recognition Letters*, vol. 184, pp. 155-165, 2013.
[13] H. Peng, F. Long and C. Ding, "Feature Selection Based on Mutual Information:Criteria of Max-Dependency,Max-Relevance,and Min-Redundancy,"*IEEE Trans.Pattern Anal.Mach.Intell*, vol. 27, no. 8, pp. 1226-1238, 2005.
[14] Y. Zheng and C. K. Kwoh, "A feature subset selection method based on highdimensional mutual information,"*Entropy*, vol. 13, no. 4, pp. 860-901, 2011.
[15] P. A. Estevez, M. Tesmer, C. A. Perez and J. M. Zurada, "Normalized mutual information feature selection,"*IEEE Trans,Neural Netw*, vol. 20, no. 2, pp. 189-201, 2009.
[16] G. Doquire and M. Verleysen, "Mutual information-based on feature selection for multi-label classification,"*Neurocomputing*, vol. 122, pp. 148-155, 2013.
[17] J. Che and Y. Yang, "Stochastioc correlation coefficient ensembles for variable selection,"*Information and Control,Journal of Applied Statistics*, 2016.
[18] J. Che, "Optimal sub-models selection algorithm for combination forecasting model,"*Neurocomputing*, vol. 151, pp. 364-375, 2015.
[19] L. T. Vinh, N. D. Thang and Y. K. Lee, "An improvement maximum and minimum redundancy feature selection algorithm based on normalized mutual information,"*in: 2010 10th IEEE/IPSJ International Symposium on Applications and the Internet(SAINT)2010*, pp. 395-398.
[20] V. Sasikala and V. Lakshmi prabha, "Bee Swarm based Feature Selection for Fake and Real Fingerprint Classification using Neural Network Classifiers," IAENG International Journal of Computer Science, vol. 42, no. 4, pp 389-403, 2015.
[21] J. Kuriakose and P. Vinod, "Unknown Metamorphic Malware Detection: Modelling with Fewer Relevant Features and Robust Feature Selection Techniques," IAENG International Journal of Computer Science, vol. 42, no. 2, pp 139-151, 2015.
[22] C. S. Yang, L. Y. Chuang, C. H. Ke and C. H. Yang, "A Hybrid Feature Selection Method for Microarray Classification," IAENG International Journal of Computer Science, vol. 35, no. 3, pp 285-290, 2008.
[23] E. Es, R. Senge, J. Barranquero, et al, "Dependent binary relevance models for multi-label classification,"*Pattern Recognition*, vol. 47, no. 3, pp. 1494-1508, 2014.
[24] J. Read and J. Hollmn, "A Deep Interpretation of Classifier Chains,"*Advances in Intelligent Data Analysis 2014*, pp. 251-262.
[25] Y. Jiang, H. Lin, X. Wang and D. Lu, "A Technique for Improving the Performance of Naive Bayes Text Classification,"*Springer-Verlag Berlin Heidelberg 2011*, pp. 196-203.
[26] S. Moran, Y. He and K. Liu, "Choosing the Best Bayesian Classifier: An Empirical Study," IAENG International Journal of Computer Science, vol. 36, no. 4, pp 322-331, 2009.
[27] Mulan: A Java Library for Multi-label Learning. $< http://mulan.sourceforge.net/datasets.html >$, 2015.
[28] J. J. Zhang, M. Fang, J. Q. Wu and X. Li, "Robust label compression for multi-label classification, "*Knowledge-Based Symtems*, vol. 107, pp. 32-42, 2016.
[29] M. Hall, E. Frank, G. Holmes, et al, "The WEKA data mining software: an updata, "*ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10-18, 2009.

Ge Huang was born in Yuncheng, Shanxi Province, China in 1992. She received her B. S. degree in the Department of Mathematics from Taiyuan Normal University in 2015, and began work for a M. S. degree in the Department of Mathematics from in Xidian University in 2015. She is major in Probabilistic graphical, data analysis and its application.

Youlong Yang received his B. S. and M. S. in the Department of Mathematics from Shaanxi Normal University, Xian, China in 1990 and 1993 respectively, and Ph.D. in System Engineering from Northwester Polytechnical University, Xian, China in 2003. Since 2004, he has been with the faculty at Xidian University, Xi'an, China. His research interests include Machine learning, Statistical data analysis and Probabilistic graphical models.

Jing Bai was born in Lvliang, Shanxi Province, China in 1993. She received her B. S. degree from Shanxi Normal University of Mathematics and Computer Science in 2014 and began work for a M. S. degree in the Department of Mathematics at Xidian University in 2015. She is major in Probabilistic graphical, data analysis and its application.