A Multi-label Classification Algorithm Combining Feature Screening and Label Correlation

Xinying Chen*, Xupeng Liang, Weiguo Yi, Xudong Song, Di Wang, and Yina Zhang

***Abstract*—Multi-label classification is a hot topic in the field of data mining. It has important applications in text classification, image, video annotation, music emotion classification, and other fields. In the past, most papers only used label correlation or feature screening to improve the accuracy of the multi-label classification and paid one-sided attention to feature screening while ignoring the correlation between labels. Therefore, in this paper, a multi-classification algorithm (MIRD) combining feature screening and label correlation is proposed, which not only combines the correlation between labels and features and design thresholds to screen features, but also uses association rules to update the label set to realize the correlation between labels, making full use of the correlation for multi-label classification. Finally, the proposed algorithm is compared with other multi-label algorithms, and the results show that it can achieve better results from most data sets, which proves that the proposed algorithm is better than the comparison algorithm.**

***Index Terms*—multi-label-classification, mutual-information, feature-screening, association-rules, label-correlation**

I. Introduction

TRADITIONAL multi-classification is when an example has several categories to choose from, but in the end can only belong to a single category. Multi-labeling means that an example may have more than one category label. For example, a film may fall under both the history and love categories. There may be more than one keyword in an article. In multi-label classification, there is still some correlation between these labels, and the accuracy of classification can be improved by using the correlation between these labels. At the same time, an example is made up of multiple features, and the selection of significant features can improve the accuracy of the classification. Multi-label classification has important applications in existing production and research areas[1], such as searching for films based on certain label information, searching for articles based on multiple keywords, and so on.

Manuscript received November 20, 2022; revised October 16, 2023.

This work was supported by the Liaoning Provincial Science and Technology Department (No. 1655706734383).

Xinying Chen is an associate professor at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, China (corresponding author, e-mail: chenxy1979@163.com)

Xupeng Liang is a postgraduate student at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, China (e-mail: lxpfm1055@163.com)

Weiguo Yi is an associate professor at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, China (e-mail: jiekexun98@163.com)

Xudong Song is a professor at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, China (e-mail: xudongsong@126.com)

Di Wang is a postgraduate student at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, China (e-mail: 1060496698@qq.com)

Yina Zhang is a postgraduate student at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, China (e-mail: 1345089796@qq.com)

In multi-label classification issues, an example is often composed of multiple features, even represented by vectors of high-dimensional features. However, vectors of large dimensions will not only cause computational difficulties, but will also reduce the precision of multi-label classification due to the redundancy of features. As a result, how to improve the accuracy of multi-labeling based on reducing the number of feature vectors requires further research.

Currently, researchers have come up with many algorithms for feature selection. Xu Hongfeng et al[2]. proposed using mutual information to measure the relationship between characteristics and the correlation of labels and then select a subset of characteristics according to the ranking. Pereira et al[3]. improved the information gain method and used it in multi-label classification to obtain a subset of characteristics based on a certain proportion. On the basis of literature[3], Pereira et al[4]. postponed the feature selection of classification, that is, different features were selected according to different instances, and good results were achieved by combination with multi-label classification algorithm. Yu et al[5]. proposed that firstly, mutual information is used to select local optimal features of each category marker, and then a genetic algorithm is used to select global optimal features. Wang Zhengkai et al[6]. applied Fisher Score (FS) feature evaluation index to multi-label classification, and proposed to obtain a dense sample set by multiplying the radius coefficient from the extreme point of each class of sample under different labels to the centre point of the class of samples, calculating the FS score of features, and sorting out feature subsets.

In multi-label classification, there is always some correlation between label. For example, if a landscape picture is marked “sea”, it is probably marked “spray”. Using the “blue sky” tag, it is likely that the “white cloud” tag will appear. Thus, predicting possible labels by studying the correlation between labels can improve the accuracy of multi-label classification to some degree. Therefore, the correlation between labels has also been a hot topic in the research of multi-label classification recently. According to the different correlation modes between labels, Zhang et al[7]. divided the existing marker correlation between three types: first-order strategy, second-order strategy and third-order strategy. The first order strategy consists of converting the multi-label issue with several single-label issues. The Binary Relevance (BR) method[8] involves converting the multi-label problem into a binary classification problem and forming a separate classifier for each tag. If the label is included in the example, it is a positive sample, otherwise it is a negative sample and will ignore the correlation between the labels. The second order strategy accounts for the correlation between labels and uses the paired correlation between labels. However, in real life, the correlation between labels often goes beyond the

second order, so this method has certain limits. Although this strategy can make full use of the correlation between tags, the computational complexity is too high for practical application, and it cannot be processed when the amount of data is too large and the tags are too many. Huang et al[9]. proposed that the relationship between labels may exist in local data sets. Liu Junyu et al[10]. used association rules to mine the correlation between labels, predicted the occurrence probability of unknown labels according to known labels, and used clustering to divide the dataset into several classes, then obtained the association rule set of each class according to the set confidence, and then updated the label set of each class, which achieved good results.

There are two deficiencies in the available literature. First, the above algorithm uses different rating indices to filter the characteristics, but does not determine the number of characteristics within the subset of characteristics. While Zhang Zhenhai et al[11]. used a threshold to determine the number of subsets of entities and obtained good results, they did not consider the correlation between the labels. Although mutual information may be used to measure the correlation between discrete characteristics and labels, it cannot directly address the correlation between continuing characteristics and labels. The second is to use association rules to update tags, although the correlation between tags can be used to achieve good results. However, when the marker density is too large, too many association rules will be generated, which will be very time-consuming, and the accuracy of multi-marker classification will decrease, because it is based on one marker to inferring another marker, so there will be a certain probability of error correction. It also fails to screen the feature set, which can lead to computational difficulties, when the feature is too large.

Based on the inadequacies of the above methods, this paper proposes a multi-label classification algorithm (MIRD) which combines the screening of characteristics and the correlation of markers. Firstly, use mutual information to measure characteristic and the correlation between tags, design a type of threshold into filter characteristics, feature high correlation between tag and feature subset, and using the method of region partition will be divided to several area continuous characteristics numerical space, transformed into discrete characteristic, make continuous mutual information can be calculated with the correlation between the tags. Secondly, based on the correlation between tags to mining association rules, in view of the above multi-label classification method based on association rules, in this paper, on the basis of the above methods, for each category of association rules is more, the use of the selected confidence-level high association rules, by reducing the number of association rules and reducing the calculation time, can achieve better results. This paper combines the correlation between features and labels. The correlations between tags are also considered. Experimental results show that the proposed algorithm may achieve better experimental results with the same dataset and parameters than other comparison algorithms.

In section 2, the concepts and formulas of multi-label classification, mutual information and association rules will be introduced. Section 3 provides the mathematical model and pseudocode of the algorithm proposed in this paper. Section 4 gives the experimental process, data sets, experimental results and experimental analysis of the algorithm. Section 5 summarizes and looks forward to this paper.

II. Related Knowledge

A. Multi-Label Classification

With the research on multi-label classification, there are many research methods and directions of multi-label classification. Currently, multi-label classification can be approximately divided into problem transformation type and algorithm adaptation type.

Problem transformation type: The multi-label classification is converted into a single label classification and then a single label classifier is used for classification. For example, the Label Powerset (LP) method [12] creates a label for the affected subset of labels in the label set. Such methods will relevant labels together, took into account the correlation between labels, but there are three questions, one is not predicted label combination does not appear, the second is simply to encode labels, did not make full use of the label, the relationship among the three is the converted may cause categories imbalances. Read et al[13]. improved LP method and proposed Pruned Problem Transformation (PPT) method. This method can predict the tag set that has not appeared in the training set, and at the same time, it will set the minimum category threshold to filter out the category data with less frequency. BR method [9] is a common problem conversion method. It converts the multi-label classification into a binary classification and assigns a classifier to each tag. That is, for tag l_j in the tag set, if an instance of the dataset contains the tag l_j, The L_j tag for this instance is assigned to 1, otherwise it is assigned to 0. Then, the classification result sets of more than one classifier are merged into the classification result of that instance. The classification method is relatively simple, but does not consider the correlation between labels, resulting in poor classification accuracy. The CC (Classifier Chain) method[14] compensates for the defect of the BR method which does not take into account the correlation between the tags. Whenever the training sample passes through a grader, the classification result is added to the classification characteristic. In order to improve the order in which the two CC classifiers are arranged, the ECC (Ensembles of Classifier Chains) method[14] randomly produces CC combinations of different label sequences. In order to reduce the impact of the order of CC.

Algorithm fitness: The single-label classifier is improved to adapt to multi-label classification. The common multi-label classifiers are ML-KNN algorithm[15], which is based on KNN algorithm. The basic idea is to identify the k nearest neighbours in the training set instances, and then obtain statistical information from these sets of instance markers. C4.5 decision tree algorithm[16], the basic idea is to extend the entropy of information about the single label issue to the multi-label issue; Based on AdaBoost method [17], two algorithms AdaBoost. MH and AdaBoost. MR are proposed, the former is to reduce Hamming-loss, the latter is to minimize Ranking loss. The random forest algorithm[23] uses the idea of ensemble learning to integrate multiple decision trees. For classification issues, output categories are identified based on the classification outcomes of most decision trees.

The multi-label problem is defined as follows: Let $D = \{d_1, d_2, d_3, \ldots, d_n\}$ is the dataset, $X = \{x_1, x_2, x_3, \ldots, x_m\}$ is one of the examples; $L = \{l_1, l_2, l_3, \ldots, l_k\}$ as the tag set, then set the example belongs belongs to tag set $Y = \{l_1, l_2, l_3, \ldots, l_u\}$, where $u \leqslant k$, $Y \subseteq L$; So the multi-label classification can be described as follows: for each example $X_i \in D$, there exists a tag set $Y_i \subseteq L$ such that each example corresponds to a tag set, where $1 < i < n$, and each example is represented by multiple features.

B. Mutual Information

Mutual information [18] is an information measure that measures the degree of correlation between random variables. When there exists a variable X and another random variable Y, the mutual information of these two random variables can be expressed as:

$$I(X;Y) = H(X) - H(X \mid Y) \tag{1}$$

Where $H(X)$ is expressed as the information entropy of a random variable X, $H(X|Y)$, X brought by information entropy.

It is commonly understood that the random variable X is uncertain, and the degree of uncertainty can be expressed by $H(X)$. If X and Y have a certain relationship, and the random variable $Y$is determined, The information entropy of X $H(X)$ minus the information entropy of X when Y exists, because when Y is determined, the information entropy of X will change to some extent. From the perspective of probability theory, it can also be expressed as:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) log\left(\frac{P(x,y)}{p(x)p(y)}\right) \tag{2}$$

Where $P(x)$ is the probability of the existence of x, and $P(x,y)$ is the probability of the joint distribution of x and y.

C. Association Rule

Association rule is an implication in the shape of $X \rightarrow Y$. Its purpose is to mine the association relationship between the items in the data set. It is a common technique in the field of data mining.

The generation of association rules consists of two steps: generating frequent item sets and generating association rules with frequent item sets. Suppose the item set is $It = \{it1, it2, it3, \ldots, itn\}$, the transaction set is $Tr = \{tr_1, tr_2, tr_3, \ldots, tr_m\}$, tr_i of $it_j \in It$, if tr_i contains the number of items for n, It is called a n-item sets. The generation of frequent item sets is to calculate the occurrence probability of each item in the item set in the transaction set, and then filter the items with probability greater than the minimum support according to the minimum support. They combine them in pairs to recalculate the probability of occurrence, and filter them according to the minimal support until no new set of items can be formed. The association rule generation is to generate the non-empty subset of the frequent item set according to the generated frequent item set and calculate the probability that the elements in the non-empty subset appear in the same transaction set. For example, $A \rightarrow BC$ is the probability that BC appears when item A appears. Then filter according to the minimum confidence Minconf until a strong association rule that satisfies Minsupport and Minconf is found. The support degree is expressed as the probability of item set X, Y appearing in the total item set, and the formula is:

$$\begin{aligned} Support(X \rightarrow Y) &= \frac{P(X,Y)}{P(Tr)} \\ &= \frac{P(X \cup Y)}{P(Tr)} = \frac{Count(XUY)}{Count(Tr)} \end{aligned} \tag{3}$$

Where $Count(.)$ represents the number of occurrences of a particular item set in the transaction set, and $P(X \cup Y)$ represents the probability of the joint occurrence of X and Y.

Confidence represents the probability that Y is deduced from the association rule "$X \rightarrow Y$" given the occurrence of the prerequisite X. That is, in the item set containing X, what is the possibility of containing Y, The formula is:

$$\begin{aligned} Confidence(X \rightarrow Y) &= P(Y \mid X) \\ &= \frac{P(Y,X)}{P(X)} = \frac{P(Y \cup X)}{P(X)} \end{aligned} \tag{4}$$

III. Multi-Label Classification Algorithm, Combining Feature Screening and Label Correlation

A. Normalization of Feature Sets

In most datasets, between the different features of each dataset, the range of values is different. For example, some data set values range between single figures, but some feature values as in ten or even a hundred. It will not just affect operational efficiency, it will also affect the outcome of the experiment to some extent. Therefore, it is necessary to normalize the dataset. Because the defined marker value is 0 or 1, only the feature set is normalized. In this paper, the MaxAbsScaler function is used to normalize the feature set. The value range is between [-1,1].

B. Feature Screening Algorithm based on Mutual Information

In this paper, mutual information is used to calculate the correlation between characteristics and labels, and the calculated correlation between characteristics and labels is sorted after taking the mean value. And then the feature subset with a high correlation is screened according to a certain threshold. Since the features in most data sets are not discrete, and the mutual information gain is mainly used to calculate the discrete type, the continuous type eigenvalues need to be converted into discrete type eigenvalues. In this document, the uniform interval between regions was chosen according to the respective values themselves, and the spacing of each region was the same. The regions were divided according to the characteristics of the dataset and represented by discrete numbers, respectively. To make the correlation between the features and labels obtained from mutual information gain calculation within the same value range, which is convenient for future calculation, MaxAbsScaler function is used for normalization.

The threshold is used to screen out features with a high correlation between features and labels, which makes up for the previous problem that although the correlation value is calculated, the number of feature subsets cannot be determined[2]. In this article, the average correlation value between each characteristic and all labels is multiplied by the coefficient 0.7 to determine the size of the threshold. The calculation formula of threshold is as follows:

$$u = \frac{\sum_{i=0}^{n} \sum_{j=0}^{m} R[i][j]}{mn} * 0.7 \tag{5}$$

The specific steps of the feature screening algorithm based on mutual information are as follows:

Algorithm 1 Feature Screening Algorithm Based on Mutual Information

input: feature set $Features = \{f_1, f_2, f_3, ..., f_n\}$ label set $Labels = \{l_1, l_2, l_3, ..., l_m\}$
output: feature subset $Feature\_subs$

```
for i = 0 → n do
    for j = 0 → m do
        R[i][j] = H(f_i) − H(f_i|l_j)
    end for
end for
The R (mutual information matrix of features and labels) is normalized.
IG = ⊘
IG = (1/m) Σ_{j=0}^{m} R[i][j]
u = (1/n IG) * 0.7
Sort(IG) //Sort it from largest to smallest
Feature_subs = ⊘
for i = 0 → n do
    if IG_i > u then
        Feature_subs = Feature_subs ∪ Features[i]
    else
        break
    end if
end for
```

C. Label Correlation Algorithm using Association Rules

In the past, using association rules to solve multi-label classification problems was to decompose and mine the single tag association rule[17], and transform the multi-label problem into a single tag problem, which does not take advantage of the correlation between tags. In this article, Liu Junyu et al[10]. proposed the idea of exploiting the correlation between labels by association rules, and further enhancing the method. The method can make full use of the correlation between tags, and achieve better results.

The MIRD algorithm first clusters the feature sets in the dataset. In this paper, the density peak clustering algorithm (DPC) clustering algorithm[19] is used for clustering. The DPC algorithm can automatically find the cluster centre, and the cluster centre can be determined once for any data set. Meanwhile, the DPC algorithm is highly suitable for analyzing large-scale data clusters, and the cluster effect is better than the traditional k-means clustering algorithm. The reason why the dataset feature sets are clustered is because the relationship between labels exists in the local dataset[11]. For example, when judging the colors of national flags of different countries according to the set of features, the meanings of the colors of national flags of different countries are different. Due to their various languages, customs, religious beliefs and cultures with. It is necessary to classify the instances of similar features such as religious belief into a class by clustering algorithm, and then use association rule mining algorithm to mine association rules among tags. Another point is that mining association rules directly globally yields fewer association rules, while mining association rules locally yields more[10].

In this paper, AprioriPlus algorithm is used to mine association rules between tags. Compared to the Apriori algorithm, the algorithm can expedite the association rule extraction process. Firstly, the minimum support is set, and the tag set is used as the transaction set to mine the frequent item sets. Then, the minimum confidence is set to mine the strong association rules using the frequent item sets. If the association rule is in the form of $A \rightarrow BC$ confidence is 0.8, then when A=1, the probability of BC is 0.8. Therefore, the update rule is that when the condition A is marked as 1 and BC in the instance is marked as 0, the BC tag of the instance is assigned as 1. Among them, in order to avoid the high label density of some data sets, too many association rules are generated in the cluster, which causes high time complexity and affects the experimental results. Therefore, in this paper, the number of association rules greater than 700 in the cluster is sorted from the largest to the smallest according to the confidence degree, and the association rules with high confidence are screened according to a certain coefficient. The specific algorithm steps are as follows:

Algorithm 2 Tag Correlation Algorithm Based on Association Rules

input: feature set $Features = \{f_1, f_2, f_3, ..., f_n\}$ label set $Labels = \{l_1, l_2, l_3, ..., l_m\}$
output: The results after classification.

1: The feature subset is screened by algorithm 1.
2: The DPC clustering algorithm is used to cluster according to the feature set $Features$, including the training dataset and the test dataset.
3: The AprioriPlus algorithm is used to generate frequent itemsets within the generated clusters.
4: Association Rules are generated based on frequent itemsets within each cluster.
5: The more association rules in the cluster are selected according to the confidence degree.
6: Update the tag set within the cluster according to the association rules.
7: Generate the updated tag set $Label\_ups$.
8: Multi-label classification algorithms such as ML-KNN were used for classification.
9: Returns the classified result.

The MIRD algorithm first uses mutual information to calculate the correlation between features and labels, and uses the threshold to filter features and obtain subsets of features with high correlation. On this basis, DPC clustering algorithm is used to cluster according to features, and the AprioriPlus algorithm is used to mine association rules between tags in the cluster to update the tag set, which can effectively use the correlation between features and tags, and

achieve good results. Considering the high density of the label set, the calculation is complicated and the accuracy is reduced. In this paper, on the basis of the above methods, for each type of association rule with more association rules, the confidence degree is used to screen the association rules with high confidence. By reducing the number of association rules and the calculation time, it is possible to obtain better results.

IV. Experiment

To verify whether the proposed method can achieve the desired effect in the experiment, this study uses five evaluation indicators on six data sets for training and verification. The dataset from mulan[21] platform, mulan platform is an open source Java library, many multi-label datasets and classification algorithms are provided. In this paper, ML-KNN algorithm and Random Forest algorithm (RFC) multi-label classification algorithm are used for classification. The comparison algorithms are ML-KNN algorithm, RFC algorithm, ML-KNN algorithm with feature selection and RFC algorithm, ML-KNN algorithm with marker correlation (derived from the algorithm recommended by Liu Junyu et al[10]) and RFC algorithm. Five valuation indices are used in six data sets to compare them to the algorithm suggested in this paper.

A. Data Set

Flags, Birds, Emotions, Yeast, Genbase and Image are used as experimental data sets. Specific information is shown in Table 1 below.

TABLE I: Experimental Data Set

Name	Instances	Attributes	Labels	Type
Flags	194	19	7	video
Birds	645	260	19	audio
Emotions	593	72	6	music
Yeast	2417	103	14	biology
Genbase	662	1186	27	biology
Image	1000	294	5	images

B. Experiments Settings

The evaluation indexes used in this experiment are Hamming loss, One error, Coverage, Ranking loss and Average precision)[22]. A total of 5 evaluation indexes, these 5 evaluation indexes are also commonly used in multi-label classification evaluation indexes. The following is a brief introduction to the above five evaluation indicators: Hamming loss index evaluates the misclassification of labels; the One error index calculates that the marker with the highest probability in each sample marker set is in the part unmarked as 1. If it is, it is marked as 0 and if it is not marked as 1. The Coverage index calculates the mark set of each sample. The marked as 1 can be covered when the probability is arranged from the largest to the smallest, that is, the search depth required by the concept marks belonging to the sample. Ranking loss metric evaluates whether tag pairs are incorrectly ordered. The Average precision indicator indicates that in a sample, the marks ranked before the sample mark are still sample marks. Among them, the first four indicators, the smaller the value, the better, while the last one, the larger the value, the better.

In this paper, ML-KNN and RFC algorithms are used as the basic algorithms for multi-label classification. After feature screening and tag set modification by association rules, these two algorithms are used for classification. In the ML-KNN algorithm, the value of the number of neighboring samples k is 10, and the value of the smoothing parameter s is 1. In the RFC algorithm, the maximum depth Max_depth takes the value 70. In this paper, DPC is used for clustering, and the number of clusters $centres$ is 5. When FP-growth algorithm is used to mine association rules between tags, the minimum support $Minsupport$ value is 0.1 and the minimum confidence $Minconf$ value is 0.75 to ensure the experimental effect. To ensure the accuracy of the experiment, this paper used 5-fold cross validation, each dataset was divided five times, and the results were taken as the mean of five experiments.

C. Experimental Results and Analysis

RD: indicates that the labeled correlation method based on association rules is adopted, and global association rules are adopted, and global clustering is adopted. Order to prevent overfitting caused by excessive association rules, if the number of association rules in each category exceeds 300, the first 0.75 association rules are selected in descending order of confidence. MI: indicates that mutual information is used to calculate the correlation value between features and labels, and then the threshold is used to determine the number of feature subsets. The threshold coefficient was set at 0.7. MIRD: It is a method that combines the detection of characteristics with the correlation of markers, that is, the method proposed in this paper. Where the best results are obtained for the median value of the table, these are high lighted in bold.

Table 3–8 shows the comparison between the algorithm proposed in this paper and the other three algorithms in the five evaluation indexes. This experiment is divided into experiments based on ML-KNN and RFC multi-label classification algorithms. On the whole, the algorithm proposed in this paper achieves better results than other algorithms, and proves that compared with the algorithm that only uses ML-KNN algorithm or only uses feature screening and only uses association rules to update the tag set, the algorithm that combines feature screening and tag correlation proposed in this paper has certain effectiveness.

As shown in the table, MIRD-RFC method performs well on six datasets and achieves better results in three or more indexes. The MIRD-ML-KNN method achieved better results on three or more evaluation indices on five datasets: Flags, Emotions, Yeast, Genbase and Image. In the Birds dataset, the MIRD-ML-KNN method achieved suboptimal results on the whole. From the experimental results, the algorithm based on RFC is generally better than the algorithm based on ML-KNN, especially on the four datasets of Birds, Emotions, Yeast and Image.

In order to further test the comparative performance of MIRD-ML-KNN algorithm, MIRD-RFC algorithm and their comparison algorithm, Friedman is included in this paper for

further test, and the performance of the algorithm is analysed at the significance level of α=0.1, as shown in Table 2.

TABLE II: The Corresponding Critical Values of T_F Under Each Evaluation Index

Evaluation Index	T_F	Critical Values
Hamming loss	26.369	
One error	10.685	
Coverage	39.602	1.896
Ranking loss	53.378	
Average precision	40.210	

Because the total number of algorithms k=8 and the total number of data sets N=6, the critical value is found to be 1.896. Where T_F's the calculation formula can be calculated as follows:

$$T_{x^2} = \frac{12N}{k\left(k+1\right)} = \left(\sum_{j=1}^{k}\gamma_j^2 - \frac{k\left(k+1\right)^2}{4}\right) \tag{6}$$

$$T_F = \frac{\left(N-1\right)T_{x^2}}{N\left(k-1\right)-T_{x^2}} \tag{7}$$

in this γ_j represents the average ordinal value of the j algorithm.

According to Friedman's conclusion, the T_F value of each index in Table 8 is greater than the critical value, so the hypothesis of no difference in the performance of all algorithms is rejected. To further compare the performance differences of each algorithm, Nemenyi follow-up test is used in this paper. The critical value CD can be calculated as follows:

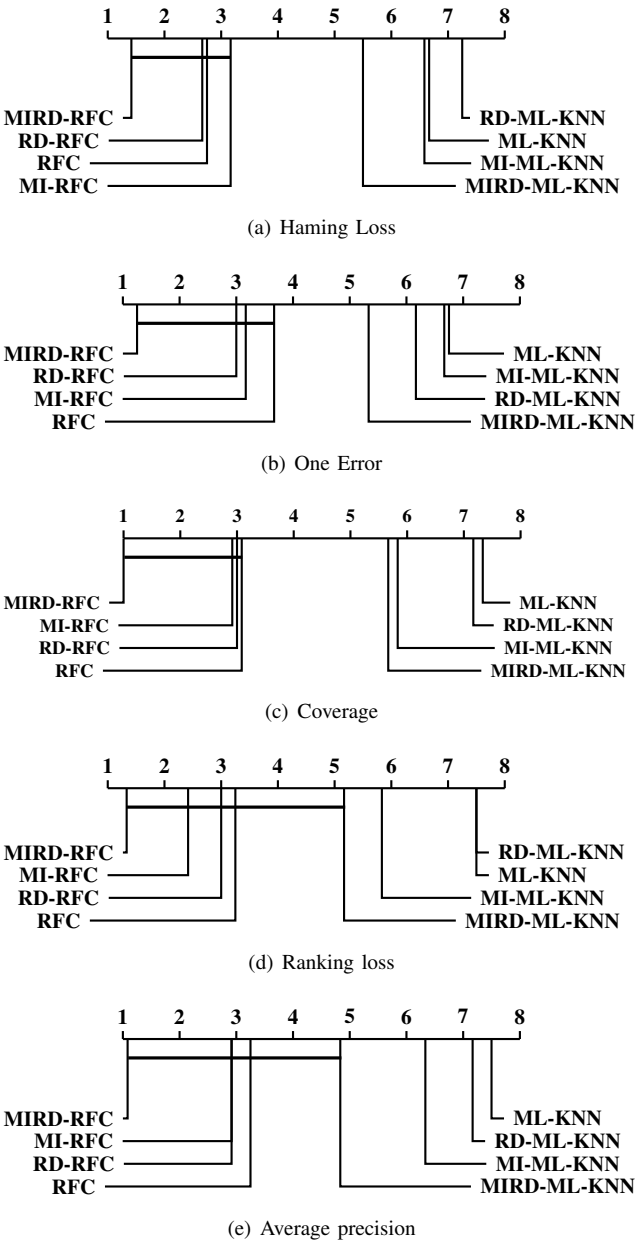
$$CD = q_\alpha\sqrt{\frac{k\left(k+1\right)}{6N}} \tag{8}$$

In particular, $q\alpha$ can be known as 2.780 when the significance level is α=0.1, and CD=3.932 can be calculated. According to the average order value of evaluation indexes on all data sets, it can be shown in Figure 1.

If the interval between the algorithms is less than the CD value, it means that there is no significant difference between the algorithms. The more the values from Figure (a) to Figure (e) are to the left, the better the effect. Compared with the improved algorithm based on ML-KNN, the improved algorithm based on RFC has a certain improvement in HL and CV indicators. In each algorithm, especially in the AV indicators, the algorithm based on MIRD has improved to a certain extent compared with the comparison algorithm. The internal difference between the algorithm based on ML-KNN and the algorithm based on RFC is not obvious. In a word, the MIRD-RFC algorithm is obviously superior to the MIRD-ML-KNN algorithm.

V. Conclusion

This paper proposes a combination of feature selection and marked correlation of multi-label classification algorithm (MIRD). In the process of research, to feature selection, to reduce the redundant features, improve multi-label classification results. This study uses mutual information to calculate the correlation value between labels and features as standard to filter features, in order to determine the number of feature subsets. In this paper, a method of calculation of the thresholds is proposed, which can improve the experimental results by reducing the features. Furthermore, this paper is also inspired by the idea of using association rules to realize and improve the correlation between tags. In summary, this paper combines the correlation between characteristics and labels with the correlation between labels. The experimental results show that the proposed algorithm can achieve better experimental results than using feature screening alone and using association rules to realize the correlation between tags and achieve multi-label classification. It also illustrates the efficacy of the proposed algorithm. In the next stage, the mainresearch direction will be how to better screen features and make a more accurate use of the correlation between tags, to improve the results and performance of multi-label classification.



(a) Haming Loss

(b) One Error

(c) Coverage

(d) Ranking loss

(e) Average precision

Fig. 1: Nemenyi Significance Test Effect of Each Algorithm Under Different Indexes

TABLE III: Classification Results of the Comparison Algorithm on Flags Dataset (mean ± standard deviation)

Method	Hamming Loss ↓	One Error ↓	Coverage ↓	Ranking Loss ↓	Average Precision ↑
ML-KNN	0.2763±0.0420	0.1958±0.0497	3.9444±0.2128	0.2390±0.0353	0.8066±0.0232
RD-ML-KNN	0.2858±0.0143	0.1753±0.0932	3.9650±0.1519	0.2421±0.0328	0.8061±0.0302
MI-ML-KNN	0.2768±0.0186	0.2009±0.0609	3.8352±0.1208	0.2343±0.0202	0.8019±0.0178
MIRD-ML-KNN	**0.2709±0.0197**	**0.1598±0.0555**	**3.7528±0.1454**	**0.2144±0.0259**	**0.8226±0.0213**
RFC	**0.2416±0.0238**	0.2220±0.0610	3.6607±0.2610	0.1988±0.0306	0.8204±0.0239
RD-RFC	0.2599±0.0138	0.1598±0.0379	3.7059±0.1246	0.2057±0.0193	0.8246±0.0117
MI-RFC	0.2489±0.0222	0.1756±0.0724	3.6031±0.1799	**0.1859±0.0276**	0.8329±0.0205
MIRD-RFC	0.2479±0.0258	**0.1443±0.0341**	**3.6030±0.1621**	0.1875±0.0310	**0.8383±0.0122**

TABLE IV: Classification Results of the Comparison Algorithm on Birds Dataset (mean ± standard deviation)

Method	Hamming Loss ↓	One Error ↓	Coverage ↓	Ranking Loss ↓	Average Precision ↑
ML-KNN	**0.0477±0.0065**	0.8171±0.0405	3.1504±0.6219	0.1212±0.0229	0.2459±0.0376
RD-ML-KNN	0.0478±0.0071	**0.8093±0.0298**	3.1411±0.5712	0.1206±0.0206	0.2480±0.0367
MI-ML-KNN	0.0487±0.0071	0.8279±0.0460	**3.0961±0.3689**	**0.1194±0.0135**	0.2510±0.0498
MIRD-ML-KNN	0.0501±0.0066	0.8171±0.0386	3.1256±0.3937	0.1200±0.0144	**0.2522±0.0455**
RFC	0.0433±0.0051	0.6512±0.0468	2.2016±0.1774	0.0853±0.0109	0.3639±0.0557
RD-RFC	0.0437±0.0045	0.6574±0.0377	2.1798±0.4482	0.0826±0.0173	0.3662±0.0422
MI-RFC	0.0438±0.0052	0.6620±0.0527	2.2217±0.3368	0.0838±0.0157	0.3592±0.0555
MIRD-RFC	**0.0425±0.0057**	**0.6496±0.0389**	**2.1628±0.3785**	**0.0800±0.0149**	**0.3672±0.0503**

TABLE V: Classification Results of the Comparison Algorithm on Emotions Dataset (mean ± standard deviation)

Method	Hamming Loss ↓	One Error ↓	Coverage ↓	Ranking Loss ↓	Average Precision ↑
ML-KNN	0.1945±0.0199	0.4081±0.0358	2.0537±0.1714	0.2184±0.0283	0.7268±0.0220
RD-ML-KNN	0.1978±0.0202	0.4150±0.0519	2.0521±0.1801	0.2205±0.0280	0.7256±0.0249
MI-ML-KNN	0.1978±0.0234	0.3660±0.0709	1.9997±0.1296	0.2087±0.0231	0.7429±0.0255
MIRD-ML-KNN	**0.1933±0.0202**	**0.3508±0.0796**	**1.9557±0.1588**	**0.2022±0.0308**	**0.7529±0.0304**
RFC	0.1827±0.0114	0.2580±0.0263	1.7216±0.1355	0.1530±0.0218	0.8067±0.0204
RD-RFC	0.1804±0.0138	0.2665±0.0151	1.6914±0.1248	0.1526±0.0166	0.8062±0.0096
MI-RFC	0.1846±0.0107	0.2630±0.0314	1.6980±0.1560	0.1517±0.0232	0.8054±0.0170
MIRD-RFC	**0.1785±0.0102**	**0.2512±0.0225**	**1.6830±0.1166**	**0.1484±0.0197**	**0.8130±0.0137**

TABLE VI: Classification Results of the Comparison Algorithm on Yeast Dataset (mean ± standard deviation)

Method	Hamming Loss ↓	One Error ↓	Coverage ↓	Ranking Loss ↓	Average Precision ↑
ML-KNN	0.2028±0.0023	0.2627±0.0089	**7.1696±0.0813**	0.2137±0.0077	0.7092±0.0084
RD-ML-KNN	0.2026±0.0024	**0.2623±0.0087**	7.1858±0.1025	0.2141±0.0085	0.7100±0.0087
MI-ML-KNN	0.2019±0.0044	0.2698±0.0222	7.1784±0.1369	0.2115±0.0090	0.7102±0.0080
MIRD-ML-KNN	**0.2015±0.0050**	0.2664±0.0221	7.1887±0.1258	**0.2113±0.0086**	**0.7116±0.0081**
RFC	0.1941±0.0035	0.2338±0.0126	6.1444±0.0916	0.1683±0.0031	0.7585±0.0062
RD-RFC	0.1925±0.0022	**0.2296±0.0075**	6.1423±0.0745	**0.1673±0.0012**	**0.7614±0.0056**
MI-RFC	0.1941±0.0035	0.2338±0.0126	6.1444±0.0916	0.1683±0.0031	0.7585±0.0062
MIRD-RFC	**0.1922±0.0029**	0.2325±0.0053	**6.1241±0.0835**	0.1676±0.0027	**0.7614±0.0044**

TABLE VII: Classification Results of the Comparison Algorithm on Genbase Dataset (mean ± standard deviation)

Method	Hamming Loss ↓	One Error ↓	Coverage ↓	Ranking Loss ↓	Average Precision ↑
ML-KNN	0.0030±0.0016	0.0182±0.0157	0.9951±0.4510	0.0379±0.0179	0.9632±0.1540
RD-ML-KNN	0.0028±0.0016	0.0197±0.0157	0.9620±0.4766	0.0360±0.0183	0.9648±0.0160
MI-ML-KNN	0.0026±0.0015	**0.0121±0.0086**	0.8591±0.3856	0.0309±0.0127	0.9700±0.0108
MIRD-ML-KNN	**0.0025±0.0015**	0.0136±0.0099	**0.8199±0.3930**	**0.0294±0.0134**	**0.9711±0.0117**
RFC	0.0011±0.0012	0.0045±0.0068	0.4628±0.2720	0.0126±0.0120	0.9870±0.0124
RD-RFC	0.0012±0.0011	0.0045±0.0068	0.4764±0.2631	0.0134±0.0114	0.9863±0.0118
MI-RFC	**0.0010±0.0009**	**0.0030±0.0068**	0.4553±0.2469	0.0118±0.0094	0.9882±0.0100
MIRD-RFC	**0.0010±0.0009**	**0.0030±0.0068**	**0.4432±0.2534**	**0.0111±0.0099**	**0.9884±0.0101**

TABLE VIII: Classification Results of the Comparison Algorithm on Images Dataset (mean ± standard deviation)

Method	Hamming Loss ↓	One Error ↓	Coverage ↓	Ranking Loss ↓	Average Precision ↑
ML-KNN	0.1628±0.0091	0.4080±0.0266	1.0390±0.0540	0.1980±0.0066	0.7433±0.0144
RD-ML-KNN	0.1640±0.0100	0.4030±0.0202	1.0300±0.0436	0.1957±0.0060	0.7464±0.0107
MI-ML-KNN	0.1602±0.0100	0.3650±0.0345	0.9460±0.0615	0.1765±0.0068	0.7724±0.0167
MIRD-ML-KNN	**0.1600±0.0107**	**0.3550±0.0300**	**0.9430±0.0627**	**0.1753±0.0059**	**0.7753±0.0134**
RFC	0.1434±0.0148	0.2570±0.0452	0.7390±0.0946	0.1247±0.0185	0.8330±0.0276
RD-RFC	**0.1382±0.0107**	0.2630±0.0295	0.7640±0.0576	0.1307±0.0055	0.8271±0.0134
MI-RFC	0.1410±0.0099	0.2540±0.0360	0.7430±0.0391	0.1261±0.0076	0.8337±0.0164
MIRD-RFC	0.1388±0.0106	**0.2400±0.0218**	**0.7240±0.0640**	**0.1210±0.0060**	**0.8425±0.0114**

REFERENCES

[1] B. Wu, E. Zhong, A. Horner, and Q. Yang, "Music emotion recognition by multi-label multi-layer multi-instance multi-view learning," Proceedings of the 22nd ACM international conference on Multimedia, PP.117-126, 2014.

[2] Hongfeng Xu, Zhenqiang Sun, "Fast feature selection method based on mutual information in multi-label learning," Computer Applications, pp.2815-2821, 2019.

[3] R. B. Pereira, A. P. D. Carvalho, B. Zadrozny, and L. H. D. C. Merschmann, "Information gain feature selection for multi-label classification," Journal of Information and Data Management, pp.48-48, 2015.

[4] R. B. Pereira, A. Pereira, B. Zadrozny, and L. H. Merschmann, "A lazy feature selection method for multi-label classification," Intelligent Data Analysis, pp.21-34, 2021.

[5] Y. Yu, Y. Wang, "Feature selection for multi-label learning using mutual information and GA," International Conference on Rough Sets and Knowledge Technology. Springer, Cham, pp.454-463, 2014.

[6] Zhengkai Wang, Dongsheng Shen, Chenxi Wang, "Fast Multi-label Feature Selection algorithm based on Fisher Score text classification," Computer engineering, pp.113-124, 2022.

[7] M. L. Zhang, Z. H. Zhou, "A review on multi-label learning algorithms," IEEE transactions on knowledge and data engineering, pp.1819-1837, 2013.

[8] M. L. Zhang, Y. K. Li, X. Y. Liu, and X. Geng, "Binary relevance for multi-label learning: an overview," Frontiers of Computer Science, pp.191-202, 2018.

[9] S. J. Huang, Z. H. Zhou, "Multi-label learning by exploiting label correlations locally," Proceedings of the AAAI Conference on Artificial Intelligence, 2012.

[10] Junyu Liu, Xiuyi Jia, "A multi-label classification of association rule mining algorithm," Journal of software, pp.2865-2878, 2017.

[11] Zhenhai Zhang, Shining Li, Zhigang Li, Hao Chen, "A class of multi-label feature selection algorithm based on information entropy," Computer research and development, pp.1177-1184, 2013.

[12] J. Read, B. Pfahringer, G. Holmes, "Multi-label classification using ensembles of pruned sets," 2008 eighth IEEE international conference on data mining. IEEE, pp. 995-1000, 2018.

[13] J. Read, "A pruned problem transformation method for multi-label classification," Proc. 2008 New Zealand Computer Science Research Student Conference (NZCSRS 2008), 2008.

[14] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," Machine learning, pp.333-359, 2011.

[15] M. L. Zhang, Z. H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," Pattern recognition, pp.2038-2048, 2007.

[16] A. Rana, R. PandyR, "A review of popular decision tree algorithms in data mining," Asian Journal of Multidimensional Research, pp.230-237, 2021.

[17] R. E. Schapire, Y. Singer, "BoosTexter: A boosting-based system for text categorization," Machine learning, pp.135-168, 2000.

[18] Cover T M, "Elements of information theory," John Wiley & Sons, 1999.

[19] B. Li, H. Li, M. Wu, and P. Li, "Multi-label classification based on association rules with application to scene classification," 2008 The 9th International Conference for Young Computer Scientists. IEEE, pp.36-41, 2018.

[20] A. Rodriguez, A. Laio, "Clustering by fast search and find of density peaks," science, pp.1492-1496, 2014.

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

[22] M. L. Zhang, Z. H. Zhou, "A review on multi-label learning algorithms," IEEE transactions on knowledge and data engineering, pp.1819-1837, 2013.

[23] M. Schonlau, R. Y. Zou, "The random forest algorithm for statistical learning," The Stata Journal, pp.3-29, 2020.